Exercise 2: First-Order Queries

Database Theory

2025-04-22

Lukas Gerlach, Maximilian Marx, Markus Krötzsch

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

1. Who is the director of "The Imitation Game"?

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

1. Who is the director of "The Imitation Game"?

$$\exists y_A. \text{Films}(\texttt{"The Imitation Game"}, x_D, y_A)[x_D]$$

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

1. Who is the director of "The Imitation Game"?

$$\exists y_A.\ \mathrm{Films}(\texttt{"The Imitation Game"}, x_D, y_A)[x_D]$$

2. Which cinemas feature "The Imitation Game"?

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

1. Who is the director of "The Imitation Game"?

$$\exists y_A.\ \mathrm{Films}(\texttt{"The Imitation Game"}, x_D, y_A)[x_D]$$

2. Which cinemas feature "The Imitation Game"?

$$\exists y_T.\ \mathrm{Program}(x_C, \texttt{"The Imitation Game"}, y_T)[x_C]$$

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	**Director**	**Actor**
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	**Address**	**Phone**
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	**Title**	**Time**
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

3. What are the address and phone number of “Schauburg”?

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

3. What are the address and phone number of “Schauburg”?

$$\text{Venues}(\texttt{"Schauburg"}, x_A, x_P)[x_A, x_P]$$

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

3. What are the address and phone number of "Schauburg"?

$$\text{Venues}(\texttt{"Schauburg"}, x_A, x_P)[x_A, x_P]$$

4. *Boolean query:* Is a film directed by "Smith" playing in some cinema?

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

3. What are the address and phone number of "Schauburg"?

$$\text{Venues}(\texttt{"Schauburg"}, x_A, x_P)[x_A, x_P]$$

4. *Boolean query:* Is a film directed by "Smith" playing in some cinema?

$$\exists y_T, y_A, y_C, z_T.\ \text{Films}(y_T, \texttt{"Smith"}, y_A) \wedge \text{Program}(y_C, y_T, z_T)$$

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

5. List the pairs of persons such that the first directed the second in a film, and vice versa.

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

5. List the pairs of persons such that the first directed the second in a film, and vice versa.

$$\exists y_T, z_T.\ \text{Films}(y_T, x_D, x_A) \land \text{Films}(z_T, x_A, x_D)[x_D, x_A]$$

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	**Director**	**Actor**
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	**Address**	**Phone**
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	**Title**	**Time**
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

5. List the pairs of persons such that the first directed the second in a film, and vice versa.

$$\exists y_T, z_T.\ \text{Films}(y_T, x_D, x_A) \wedge \text{Films}(z_T, x_A, x_D)[x_D, x_A]$$

6. List the names of directors who have acted in a film they directed.

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	**Director**	**Actor**
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	**Address**	**Phone**
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	**Title**	**Time**
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

5. List the pairs of persons such that the first directed the second in a film, and vice versa.

$$\exists y_T, z_T. \text{Films}(y_T, x_D, x_A) \wedge \text{Films}(z_T, x_A, x_D)[x_D, x_A]$$

6. List the names of directors who have acted in a film they directed.

$$\exists y_T. \text{Films}(y_T, x_D, x_D)[x_D]$$

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

7. Always return $\{\text{Title} \mapsto \texttt{"Apocalypse Now"}, \text{Director} \mapsto \texttt{"Coppola"}\}$ as the answer.

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

7. Always return $\{\text{Title} \mapsto \texttt{"Apocalypse Now"}, \text{Director} \mapsto \texttt{"Coppola"}\}$ as the answer.

$$\big\{\text{DirectedBy}(\texttt{"Apocalypse Now"}, \texttt{"Coppola"})\big\}$$

Note: FO queries always use the unnamed perspective.

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

7. Always return $\{\text{Title} \mapsto \texttt{"Apocalypse Now"}, \text{Director} \mapsto \texttt{"Coppola"}\}$ as the answer.

$$\left\{\text{DirectedBy}(\texttt{"Apocalypse Now"}, \texttt{"Coppola"})\right\}$$

   Note: FO queries always use the unnamed perspective.

8. Find the actors cast in at least one film by "Smith".

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

7. Always return $\{\text{Title} \mapsto \texttt{"Apocalypse Now"}, \text{Director} \mapsto \texttt{"Coppola"}\}$ as the answer.

$$\left\{\text{DirectedBy}(\texttt{"Apocalypse Now"}, \texttt{"Coppola"})\right\}$$

   Note: FO queries always use the unnamed perspective.

8. Find the actors cast in at least one film by “Smith”.

$$\exists y_T.\ \text{Films}(y_T, \texttt{"Smith"}, x_A)[x_A]$$

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

9. Find the actors cast in every film by "Smith."

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

9. Find the actors cast in every film by "Smith."

$$\exists y_T, y_D. \big(\text{Films}(y_T, y_D, x_A) \wedge \forall z_T, z_A. (\text{Films}(z_T, \texttt{"Smith"}, z_A) \rightarrow \text{Films}(z_T, \texttt{"Smith"}, x_A))\big)[x_A]$$

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

9. Find the actors cast in every film by “Smith.”

$$\exists y_T, y_D. \big(\text{Films}(y_T, y_D, x_A) \wedge \forall z_T, z_A. (\text{Films}(z_T, \texttt{"Smith"}, z_A) \rightarrow \text{Films}(z_T, \texttt{"Smith"}, x_A))\big)[x_A]$$

10. Find the actors cast only in films by “Smith.”

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

9. Find the actors cast in every film by "Smith."

$$\exists y_T, y_D. \left(\text{Films}(y_T, y_D, x_A) \land \forall z_T, z_A. \left(\text{Films}(z_T, \texttt{"Smith"}, z_A) \to \text{Films}(z_T, \texttt{"Smith"}, x_A)\right)\right)[x_A]$$

10. Find the actors cast only in films by "Smith."

$$\exists y_T, y_D. \left(\text{Films}(y_T, y_D, x_A) \land \forall z_T. \exists z_D. \left(\text{Films}(z_T, z_D, x_A) \to z_D \approx \texttt{"Smith"}\right)\right)[x_A]$$

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

11. Find all pairs of actors who act together in at least one film.

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

11. Find all pairs of actors who act together in at least one film.

$$\exists y_T, y_D.\ \mathrm{Films}(y_T, y_D, x_A) \land \mathrm{Films}(y_T, y_D, x_{A'}) \land x_A \not\approx x_{A'}\,[x_A, x_{A'}]$$

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

11. Find all pairs of actors who act together in at least one film.

$$\exists y_T, y_D.\ \text{Films}(y_T, y_D, x_A) \land \text{Films}(y_T, y_D, x_{A'}) \land x_A \not\approx x_{A'}\, [x_A, x_{A'}]$$

12. Find all pairs of actors cast in exactly the same films.

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

11. Find all pairs of actors who act together in at least one film.

$$\exists y_T, y_D.\ \text{Films}(y_T, y_D, x_A) \wedge \text{Films}(y_T, y_D, x_{A'}) \wedge x_A \not\approx x_{A'}\,[x_A, x_{A'}]$$

12. Find all pairs of actors cast in exactly the same films.

$$\exists y_T, y_D.\ \Big(\text{Films}(y_T, y_D, x_A) \wedge \exists z_T, z_D.\ \text{Films}(z_T, z_D, x_{A'}) \wedge \forall w, v.\ \big(\text{Films}(w, v, x_A) \leftrightarrow \text{Films}(w, v, x_{A'})\big)\Big)[x_A, x_{A'}]$$

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

13. Find the directors such that every actor is cast in one of their films.

Exercise 1

Exercise. Express the queries from Exercise 1.1 as domain-independent FO-queries.

Films

Title	Director	Actor
The Imitation Game	Tyldum	Cumberbatch
The Imitation Game	Tyldum	Knightley
...	...	...
The Internet's Own Boy	Knappenberger	Swartz
The Internet's Own Boy	Knappenberger	Lessig
The Internet's Own Boy	Knappenberger	Berners-Lee
...	...	...
Dogma	Smith	Damon
Dogma	Smith	Affleck
Dogma	Smith	Morissette
Dogma	Smith	Smith

Venues

Cinema	Address	Phone
UFA	St. Petersburger Str. 24	4825825
Schauburg	Königsbrücker Str. 55	8032185
CinemaxX	Hüblerstr. 8	3158910
...	...	...

Program

Cinema	Title	Time
Schauburg	The Imitation Game	19:30
Schauburg	Dogma	20:45
UFA	The Imitation Game	22:45
CinemaxX	The Imitation Game	19:30

Solution.

13. Find the directors such that every actor is cast in one of their films.

$$\exists y_T, y_A. \Big(\text{Films}(y_T, x_D, y_A) \land \forall z_T, z_D, z_A. \Big(\text{Films}(z_T, z_D, z_A) \rightarrow \exists w_T.\text{Films}(w_T, x_D, z_A)\Big)\Big)[x_D]$$

Exercise 2

Exercise. Let $R[A, B]$ be a table. Express the following RA_{named} query as a DI_{unnamed} query:

$$q[A, B] = \big(\pi_A(R) \bowtie \pi_B(R)\big) - \big(R \bowtie (\delta_{B,A \to A,B}(R))\big)$$

Exercise 2

Exercise. Let $R[A, B]$ be a table. Express the following RA_{named} query as a DI_{unnamed} query:

$$q[A, B] = \left(\pi_A(R) \bowtie \pi_B(R)\right) - \left(R \bowtie (\delta_{B,A \to A,B}(R))\right)$$

Solution.

Exercise 2

Exercise. Let $R[A, B]$ be a table. Express the following RA_{named} query as a DI_{unnamed} query:

$$q[A, B] = \big(\pi_A(R) \bowtie \pi_B(R)\big) - \big(R \bowtie (\delta_{B,A \to A,B}(R))\big)$$

Solution.

Definition (Lecture 2, Slide 19/20, excerpt)

For an RA query $q[a_1, \ldots, a_n]$, let $\varphi_q[x_{a_1}, \ldots, x_{a_n}]$ be the DI query defined as follows:

- If $q = R$ with signature $R[A_1, \ldots, A_n]$, then $\varphi_q = R(x_{A_1}, \ldots, x_{A_n})[x_{A_1}, \ldots, x_{A_n}]$;
- if $q = \delta_{B_1,\ldots,B_n \to A_1,\ldots,A_n} q'$, then $\varphi_q = \exists y_{B_1}, \ldots, y_{B_n}.\ (x_{A_1} \approx y_{B_1}) \wedge \ldots \wedge (x_{A_n} \approx y_{B_n}) \wedge \varphi_{q'}[y_{A_1}, \ldots, y_{A_n}]$;

  Assumption: $A_1, \ldots, A_n$ in $\delta_{B_1,\ldots,B_n \to A_1,\ldots,A_n}$ are written in attribute order; $B_1, \ldots, B_n$ may be in arbitrary order.
- if $q = \pi_{A_1,\ldots,A_n}(q')$ for a subquery $q'[B_1, \ldots, B_m]$ with $\{B_1, \ldots, B_m\} = \{A_1, \ldots, A_n\} \cup \{C_1, \ldots, C_k\}$, then $\varphi_q = \exists x_{C_1}, \ldots, x_{C_k}.\ \varphi_{q'}$;
- if $q = q_1 \bowtie q_2$, then $\varphi_q = \varphi_{q_1} \wedge \varphi_{q_2}$; and
- if $q = q_1 - q_2$, then $\varphi_q = \varphi_{q_1} \wedge \neg\varphi_{q_2}$.

Exercise 2

Exercise. Let $R[A, B]$ be a table. Express the following RA_{named} query as a DI_{unnamed} query:

$$q[A, B] = \big(\pi_A(R) \bowtie \pi_B(R)\big) - \big(R \bowtie (\delta_{B,A \to A,B}(R))\big)$$

Solution.

Definition (Lecture 2, Slide 19/20, excerpt)

For an RA query $q[a_1, \ldots, a_n]$, let $\varphi_q[x_{a_1}, \ldots, x_{a_n}]$ be the DI query defined as follows:

- If $q = R$ with signature $R[A_1, \ldots, A_n]$, then $\varphi_q = R(x_{A_1}, \ldots, x_{A_n})[x_{A_1}, \ldots, x_{A_n}]$;
- if $q = \delta_{B_1,\ldots,B_n \to A_1,\ldots,A_n} q'$, then $\varphi_q = \exists y_{B_1}, \ldots, y_{B_n}.\ (x_{A_1} \approx y_{B_1}) \wedge \ldots \wedge (x_{A_n} \approx y_{B_n}) \wedge \varphi_{q'}[y_{A_1}, \ldots, y_{A_n}]$;
  Assumption: $A_1, \ldots, A_n$ in $\delta_{B_1,\ldots,B_n \to A_1,\ldots,A_n}$ are written in attribute order; $B_1, \ldots, B_n$ may be in arbitrary order.
- if $q = \pi_{A_1,\ldots,A_n}(q')$ for a subquery $q'[B_1, \ldots, B_m]$
  with $\{B_1, \ldots, B_m\} = \{A_1, \ldots, A_n\} \cup \{C_1, \ldots, C_k\}$, then $\varphi_q = \exists x_{C_1}, \ldots, x_{C_k}.\ \varphi_{q'}$;
- if $q = q_1 \bowtie q_2$, then $\varphi_q = \varphi_{q_1} \wedge \varphi_{q_2}$; and
- if $q = q_1 - q_2$, then $\varphi_q = \varphi_{q_1} \wedge \neg\varphi_{q_2}$.

$$\varphi_{\pi_A(R)}[x_A] = \exists y_B.\ R(x_A, y_B)[x_A]$$

Exercise 2

Exercise. Let $R[A,B]$ be a table. Express the following RA_{named} query as a DI_{unnamed} query:

$$q[A,B] = \big(\pi_A(R) \bowtie \pi_B(R)\big) - \big(R \bowtie (\delta_{B,A \to A,B}(R))\big)$$

Solution.

Definition (Lecture 2, Slide 19/20, excerpt)

For an RA query $q[a_1, \ldots, a_n]$, let $\varphi_q[x_{a_1}, \ldots, x_{a_n}]$ be the DI query defined as follows:

- If $q = R$ with signature $R[A_1, \ldots, A_n]$, then $\varphi_q = R(x_{A_1}, \ldots, x_{A_n})[x_{A_1}, \ldots, x_{A_n}]$;
- if $q = \delta_{B_1,\ldots,B_n \to A_1,\ldots,A_n} q'$, then $\varphi_q = \exists y_{B_1}, \ldots, y_{B_n}.\ (x_{A_1} \approx y_{B_1}) \wedge \ldots \wedge (x_{A_n} \approx y_{B_n}) \wedge \varphi_{q'}[y_{A_1}, \ldots, y_{A_n}]$;

  Assumption: $A_1, \ldots, A_n$ in $\delta_{B_1,\ldots,B_n \to A_1,\ldots,A_n}$ are written in attribute order; $B_1, \ldots, B_n$ may be in arbitrary order.
- if $q = \pi_{A_1,\ldots,A_n}(q')$ for a subquery $q'[B_1, \ldots, B_m]$ with $\{B_1, \ldots, B_m\} = \{A_1, \ldots, A_n\} \cup \{C_1, \ldots, C_k\}$, then $\varphi_q = \exists x_{C_1}, \ldots, x_{C_k}.\ \varphi_{q'}$;
- if $q = q_1 \bowtie q_2$, then $\varphi_q = \varphi_{q_1} \wedge \varphi_{q_2}$; and
- if $q = q_1 - q_2$, then $\varphi_q = \varphi_{q_1} \wedge \neg\varphi_{q_2}$.

$$\varphi_{\pi_A(R)}[x_A] = \exists y_B.\ R(x_A, y_B)[x_A]$$

$$\varphi_{\pi_B(R)}[x_B] = \exists y_A.\ R(y_A, x_B)[x_B]$$

Exercise 2

Exercise. Let $R[A,B]$ be a table. Express the following RA_{named} query as a DI_{unnamed} query:

$$q[A,B] = \big(\pi_A(R) \bowtie \pi_B(R)\big) - \big(R \bowtie (\delta_{B,A \to A,B}(R))\big)$$

Solution.

Definition (Lecture 2, Slide 19/20, excerpt)

For an RA query $q[a_1,\ldots,a_n]$, let $\varphi_q[x_{a_1},\ldots,x_{a_n}]$ be the DI query defined as follows:

- If $q = R$ with signature $R[A_1,\ldots,A_n]$, then $\varphi_q = R(x_{A_1},\ldots,x_{A_n})[x_{A_1},\ldots,x_{A_n}]$;
- if $q = \delta_{B_1,\ldots,B_n \to A_1,\ldots,A_n} q'$, then $\varphi_q = \exists y_{B_1},\ldots,y_{B_n}.\,(x_{A_1} \approx y_{B_1}) \wedge \ldots \wedge (x_{A_n} \approx y_{B_n}) \wedge \varphi_{q'}[y_{A_1},\ldots,y_{A_n}]$;

  Assumption: $A_1,\ldots,A_n$ in $\delta_{B_1,\ldots,B_n \to A_1,\ldots,A_n}$ are written in attribute order; $B_1,\ldots,B_n$ may be in arbitrary order.
- if $q = \pi_{A_1,\ldots,A_n}(q')$ for a subquery $q'[B_1,\ldots,B_m]$ with $\{B_1,\ldots,B_m\} = \{A_1,\ldots,A_n\} \cup \{C_1,\ldots,C_k\}$, then $\varphi_q = \exists x_{C_1},\ldots,x_{C_k}.\,\varphi_{q'}$;
- if $q = q_1 \bowtie q_2$, then $\varphi_q = \varphi_{q_1} \wedge \varphi_{q_2}$; and
- if $q = q_1 - q_2$, then $\varphi_q = \varphi_{q_1} \wedge \neg\varphi_{q_2}$.

$$\varphi_{\pi_A(R)}[x_A] = \exists y_B.\, R(x_A, y_B)[x_A]$$

$$\varphi_{\pi_B(R)}[x_B] = \exists y_A.\, R(y_A, x_B)[x_B]$$

$$\varphi_R[x_A, x_B] = R(x_A, x_B)[x_A, x_B]$$

Exercise 2

Exercise. Let $R[A,B]$ be a table. Express the following RA_{named} query as a DI_{unnamed} query:

$$q[A,B] = \big(\pi_A(R) \bowtie \pi_B(R)\big) - \big(R \bowtie (\delta_{B,A \to A,B}(R))\big)$$

Solution.

Definition (Lecture 2, Slide 19/20, excerpt)

For an RA query $q[a_1,\ldots,a_n]$, let $\varphi_q[x_{a_1},\ldots,x_{a_n}]$ be the DI query defined as follows:

- If $q = R$ with signature $R[A_1,\ldots,A_n]$, then $\varphi_q = R(x_{A_1},\ldots,x_{A_n})[x_{A_1},\ldots,x_{A_n}]$;
- if $q = \delta_{B_1,\ldots,B_n \to A_1,\ldots,A_n} q'$, then $\varphi_q = \exists y_{B_1},\ldots,y_{B_n}.\ (x_{A_1} \approx y_{B_1}) \land \ldots \land (x_{A_n} \approx y_{B_n}) \land \varphi_{q'}[y_{A_1},\ldots,y_{A_n}]$;
  Assumption: $A_1,\ldots,A_n$ in $\delta_{B_1,\ldots,B_n \to A_1,\ldots,A_n}$ are written in attribute order; $B_1,\ldots,B_n$ may be in arbitrary order.
- if $q = \pi_{A_1,\ldots,A_n}(q')$ for a subquery $q'[B_1,\ldots,B_m]$ with $\{B_1,\ldots,B_m\} = \{A_1,\ldots,A_n\} \cup \{C_1,\ldots,C_k\}$, then $\varphi_q = \exists x_{C_1},\ldots,x_{C_k}.\ \varphi_{q'}$;
- if $q = q_1 \bowtie q_2$, then $\varphi_q = \varphi_{q_1} \land \varphi_{q_2}$; and
- if $q = q_1 - q_2$, then $\varphi_q = \varphi_{q_1} \land \neg\varphi_{q_2}$.

$$\varphi_{\pi_A(R)}[x_A] = \exists y_B.\ R(x_A, y_B)[x_A]$$

$$\varphi_R[x_A, x_B] = R(x_A, x_B)[x_A, x_B]$$

$$\varphi_{\pi_B(R)}[x_B] = \exists y_A.\ R(y_A, x_B)[x_B]$$

$$\varphi_{\pi_A(R) \bowtie \pi_B(R)}[x_A, x_B] = \varphi_{\pi_A(R)} \land \varphi_{\pi_B(R)}[x_A, x_B]$$

Exercise 2

Exercise. Let $R[A, B]$ be a table. Express the following RA_{named} query as a DI_{unnamed} query:

$$q[A, B] = \big(\pi_A(R) \bowtie \pi_B(R)\big) - \big(R \bowtie (\delta_{B,A \to A,B}(R))\big)$$

Solution.

Definition (Lecture 2, Slide 19/20, excerpt)

For an RA query $q[a_1, \ldots, a_n]$, let $\varphi_q[x_{a_1}, \ldots, x_{a_n}]$ be the DI query defined as follows:

- If $q = R$ with signature $R[A_1, \ldots, A_n]$, then $\varphi_q = R(x_{A_1}, \ldots, x_{A_n})[x_{A_1}, \ldots, x_{A_n}]$;
- if $q = \delta_{B_1,\ldots,B_n \to A_1,\ldots,A_n} q'$, then $\varphi_q = \exists y_{B_1}, \ldots, y_{B_n}.\ (x_{A_1} \approx y_{B_1}) \wedge \ldots \wedge (x_{A_n} \approx y_{B_n}) \wedge \varphi_{q'}[y_{A_1}, \ldots, y_{A_n}]$;
  Assumption: $A_1, \ldots, A_n$ in $\delta_{B_1,\ldots,B_n \to A_1,\ldots,A_n}$ are written in attribute order; $B_1, \ldots, B_n$ may be in arbitrary order.
- if $q = \pi_{A_1,\ldots,A_n}(q')$ for a subquery $q'[B_1, \ldots, B_m]$ with $\{B_1, \ldots, B_m\} = \{A_1, \ldots, A_n\} \cup \{C_1, \ldots, C_k\}$, then $\varphi_q = \exists x_{C_1}, \ldots, x_{C_k}.\ \varphi_{q'}$;
- if $q = q_1 \bowtie q_2$, then $\varphi_q = \varphi_{q_1} \wedge \varphi_{q_2}$; and
- if $q = q_1 - q_2$, then $\varphi_q = \varphi_{q_1} \wedge \neg\varphi_{q_2}$.

$$\varphi_{\pi_A(R)}[x_A] = \exists y_B.\ R(x_A, y_B)[x_A]$$

$$\varphi_R[x_A, x_B] = R(x_A, x_B)[x_A, x_B]$$

$$\varphi_{(\delta_{B,A \to A,B}(R))}[x_A, x_B] = \exists y_B, y_A.\ (x_A \approx y_B) \wedge (x_B \approx y_A) \wedge R(y_A, y_B)[x_A, x_B]$$

$$\varphi_{\pi_B(R)}[x_B] = \exists y_A.\ R(y_A, x_B)[x_B]$$

$$\varphi_{\pi_A(R) \bowtie \pi_B(R)}[x_A, x_B] = \varphi_{\pi_A(R)} \wedge \varphi_{\pi_B(R)}[x_A, x_B]$$

Exercise 2

Exercise. Let $R[A, B]$ be a table. Express the following RA_{named} query as a DI_{unnamed} query:

$$q[A, B] = \big(\pi_A(R) \bowtie \pi_B(R)\big) - \big(R \bowtie (\delta_{B,A \to A,B}(R))\big)$$

Solution.

Definition (Lecture 2, Slide 19/20, excerpt)

For an RA query $q[a_1, \ldots, a_n]$, let $\varphi_q[x_{a_1}, \ldots, x_{a_n}]$ be the DI query defined as follows:

- If $q = R$ with signature $R[A_1, \ldots, A_n]$, then $\varphi_q = R(x_{A_1}, \ldots, x_{A_n})[x_{A_1}, \ldots, x_{A_n}]$;
- if $q = \delta_{B_1,\ldots,B_n \to A_1,\ldots,A_n} q'$, then $\varphi_q = \exists y_{B_1}, \ldots, y_{B_n}.\ (x_{A_1} \approx y_{B_1}) \wedge \ldots \wedge (x_{A_n} \approx y_{B_n}) \wedge \varphi_{q'}[y_{A_1}, \ldots, y_{A_n}]$;
  Assumption: $A_1, \ldots, A_n$ in $\delta_{B_1,\ldots,B_n \to A_1,\ldots,A_n}$ are written in attribute order; $B_1, \ldots, B_n$ may be in arbitrary order.
- if $q = \pi_{A_1,\ldots,A_n}(q')$ for a subquery $q'[B_1, \ldots, B_m]$ with $\{B_1, \ldots, B_m\} = \{A_1, \ldots, A_n\} \cup \{C_1, \ldots, C_k\}$, then $\varphi_q = \exists x_{C_1}, \ldots, x_{C_k}.\ \varphi_{q'}$;
- if $q = q_1 \bowtie q_2$, then $\varphi_q = \varphi_{q_1} \wedge \varphi_{q_2}$; and
- if $q = q_1 - q_2$, then $\varphi_q = \varphi_{q_1} \wedge \neg\varphi_{q_2}$.

$$\varphi_{\pi_A(R)}[x_A] = \exists y_B.\ R(x_A, y_B)[x_A]$$

$$\varphi_R[x_A, x_B] = R(x_A, x_B)[x_A, x_B]$$

$$\varphi_{(\delta_{B,A \to A,B}(R))}[x_A, x_B] = \exists y_B, y_A.\ (x_A \approx y_B) \wedge (x_B \approx y_A) \wedge R(y_A, y_B)[x_A, x_B]$$

$$\varphi_{\pi_B(R)}[x_B] = \exists y_A.\ R(y_A, x_B)[x_B]$$

$$\varphi_{\pi_A(R) \bowtie \pi_B(R)}[x_A, x_B] = \varphi_{\pi_A(R)} \wedge \varphi_{\pi_B(R)}[x_A, x_B]$$

$$\varphi_{R \bowtie (\delta_{B,A \to A,B}(R))}[x_A, x_B] = \varphi_R \wedge \varphi_{(\delta_{B,A \to A,B}(R))}[x_A, x_B]$$

Exercise 2

Exercise. Let $R[A, B]$ be a table. Express the following RA_{named} query as a DI_{unnamed} query:

$$q[A, B] = \big(\pi_A(R) \bowtie \pi_B(R)\big) - \big(R \bowtie (\delta_{B,A \to A,B}(R))\big)$$

Solution.

Definition (Lecture 2, Slide 19/20, excerpt)

For an RA query $q[a_1, \ldots, a_n]$, let $\varphi_q[x_{a_1}, \ldots, x_{a_n}]$ be the DI query defined as follows:

- If $q = R$ with signature $R[A_1, \ldots, A_n]$, then $\varphi_q = R(x_{A_1}, \ldots, x_{A_n})[x_{A_1}, \ldots, x_{A_n}]$;
- if $q = \delta_{B_1,\ldots,B_n \to A_1,\ldots,A_n} q'$, then $\varphi_q = \exists y_{B_1}, \ldots, y_{B_n}. \, (x_{A_1} \approx y_{B_1}) \wedge \ldots \wedge (x_{A_n} \approx y_{B_n}) \wedge \varphi_{q'}[y_{A_1}, \ldots, y_{A_n}]$;
  Assumption: $A_1, \ldots, A_n$ in $\delta_{B_1,\ldots,B_n \to A_1,\ldots,A_n}$ are written in attribute order; $B_1, \ldots, B_n$ may be in arbitrary order.
- if $q = \pi_{A_1,\ldots,A_n}(q')$ for a subquery $q'[B_1, \ldots, B_m]$ with $\{B_1, \ldots, B_m\} = \{A_1, \ldots, A_n\} \cup \{C_1, \ldots, C_k\}$, then $\varphi_q = \exists x_{C_1}, \ldots, x_{C_k}. \, \varphi_{q'}$;
- if $q = q_1 \bowtie q_2$, then $\varphi_q = \varphi_{q_1} \wedge \varphi_{q_2}$; and
- if $q = q_1 - q_2$, then $\varphi_q = \varphi_{q_1} \wedge \neg\varphi_{q_2}$.

$$\varphi_{\pi_A(R)}[x_A] = \exists y_B. \, R(x_A, y_B)[x_A]$$

$$\varphi_R[x_A, x_B] = R(x_A, x_B)[x_A, x_B]$$

$$\varphi_{(\delta_{B,A \to A,B}(R))}[x_A, x_B] = \exists y_B, y_A. \, (x_A \approx y_B) \wedge (x_B \approx y_A) \wedge R(y_A, y_B)[x_A, x_B]$$

$$\varphi_{q[A,B]}[x_A, x_B] = \varphi_{\pi_A(R)} \wedge \varphi_{\pi_B(R)} \wedge \neg\Big(\varphi_R \wedge \varphi_{(\delta_{B,A \to A,B}(R))}\Big)[x_A, x_B]$$

$$\varphi_{\pi_B(R)}[x_B] = \exists y_A. \, R(y_A, x_B)[x_B]$$

$$\varphi_{\pi_A(R) \bowtie \pi_B(R)}[x_A, x_B] = \varphi_{\pi_A(R)} \wedge \varphi_{\pi_B(R)}[x_A, x_B]$$

$$\varphi_{R \bowtie (\delta_{B,A \to A,B}(R))}[x_A, x_B] = \varphi_R \wedge \varphi_{(\delta_{B,A \to A,B}(R))}[x_A, x_B]$$

Exercise 3

Exercise. It was stated in the lecture (Lecture 2, slide 17) that query mappings under named perspective can be translated into query mappings under unnamed perspective. Specify this translation.

Exercise 3

Exercise. It was stated in the lecture (Lecture 2, slide 17) that query mappings under named perspective can be translated into query mappings under unnamed perspective. Specify this translation.

Solution.

Exercise 3

Exercise. It was stated in the lecture (Lecture 2, slide 17) that query mappings under named perspective can be translated into query mappings under unnamed perspective. Specify this translation.
Solution.

- Let $\mathbb{D}_n$ and $\mathbb{D}_u$ be the sets of all database instances over a named and an unnamed perspective, respectively, and let $\mathbb{T}_n$ and $\mathbb{T}_u$ be the sets of all database tables over a named and an unnamed perspective.

Exercise 3

Exercise. It was stated in the lecture (Lecture 2, slide 17) that query mappings under named perspective can be translated into query mappings under unnamed perspective. Specify this translation.

Solution.

- Let $\mathbb{D}_n$ and $\mathbb{D}_u$ be the sets of all database instances over a named and an unnamed perspective, respectively, and let $\mathbb{T}_n$ and $\mathbb{T}_u$ be the sets of all database tables over a named and an unnamed perspective.
- Consider a query mapping $M[q] : \mathbb{D}_n \rightarrow \mathbb{T}_n$.

Exercise 3

Exercise. It was stated in the lecture (Lecture 2, slide 17) that query mappings under named perspective can be translated into query mappings under unnamed perspective. Specify this translation.

Solution.

- Let $\mathbb{D}_n$ and $\mathbb{D}_u$ be the sets of all database instances over a named and an unnamed perspective, respectively, and let $\mathbb{T}_n$ and $\mathbb{T}_u$ be the sets of all database tables over a named and an unnamed perspective.
- Consider a query mapping $M[q] : \mathbb{D}_n \to \mathbb{T}_n$.
- Define $\nu : \mathbb{T}_n \to \mathbb{T}_u$ as the function taking named database tables $R[A_1, \ldots, A_n]$ to unnamed database tables $R^{\mathcal{J}}$, such that attribute A_i is mapped to column i:

$$\nu(R^{\mathcal{I}}) = \left\{ \langle r(A_1), \ldots, r(A_n) \rangle \,\middle|\, (r : \{ A_1, \ldots, A_n \} \to \mathbf{dom}) \in R^{\mathcal{I}} \right\}$$

Exercise 3

Exercise. It was stated in the lecture (Lecture 2, slide 17) that query mappings under named perspective can be translated into query mappings under unnamed perspective. Specify this translation.

Solution.

- Let $\mathbb{D}_n$ and $\mathbb{D}_u$ be the sets of all database instances over a named and an unnamed perspective, respectively, and let $\mathbb{T}_n$ and $\mathbb{T}_u$ be the sets of all database tables over a named and an unnamed perspective.
- Consider a query mapping $M[q] : \mathbb{D}_n \to \mathbb{T}_n$.
- Define $\nu : \mathbb{T}_n \to \mathbb{T}_u$ as the function taking named database tables $R[A_1, \ldots, A_n]$ to unnamed database tables $R^{\mathcal{J}}$, such that attribute A_i is mapped to column i:

$$\nu(R^{\mathcal{I}}) = \left\{ \langle r(A_1), \ldots, r(A_n) \rangle \,\middle|\, (r : \{ A_1, \ldots, A_n \} \to \mathbf{dom}) \in R^{\mathcal{I}} \right\}$$

- Conversely, let $\mu : \mathbb{D}_u \to \mathbb{D}_n$ be the function taking unnamed database instances $\mathcal{J}$ to named database instances $\mathcal{I}$, by mapping each table $R^{\mathcal{J}}$ to a named table taking attribute A_i from column i:

$$\mu(\mathcal{J}) = \left\{ \left\{ \{ A_1 \mapsto a_1, \ldots, A_n \mapsto a_n \} \mid \langle a_1, \ldots, a_n \rangle \in R^{\mathcal{J}} \right\} \,\middle|\, R \in \mathcal{J} \right\}$$

Exercise 3

Exercise. It was stated in the lecture (Lecture 2, slide 17) that query mappings under named perspective can be translated into query mappings under unnamed perspective. Specify this translation.

Solution.

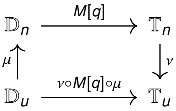
- Let $\mathbb{D}_n$ and $\mathbb{D}_u$ be the sets of all database instances over a named and an unnamed perspective, respectively, and let $\mathbb{T}_n$ and $\mathbb{T}_u$ be the sets of all database tables over a named and an unnamed perspective.
- Consider a query mapping $M[q] : \mathbb{D}_n \to \mathbb{T}_n$.
- Define $\nu : \mathbb{T}_n \to \mathbb{T}_u$ as the function taking named database tables $R[A_1, \ldots, A_n]$ to unnamed database tables $R^{\mathcal{J}}$, such that attribute A_i is mapped to column i:

$$\nu(R^{\mathcal{I}}) = \left\{ \langle r(A_1), \ldots, r(A_n) \rangle \,\middle|\, (r : \{ A_1, \ldots, A_n \} \to \mathbf{dom}) \in R^{\mathcal{I}} \right\}$$

- Conversely, let $\mu : \mathbb{D}_u \to \mathbb{D}_n$ be the function taking unnamed database instances $\mathcal{J}$ to named database instances $\mathcal{I}$, by mapping each table $R^{\mathcal{J}}$ to a named table taking attribute A_i from column i:

$$\mu(\mathcal{J}) = \left\{ \left\{ \{ A_1 \mapsto a_1, \ldots, A_n \mapsto a_n \} \mid \langle a_1, \ldots, a_n \rangle \in R^{\mathcal{J}} \right\} \,\middle|\, R \in \mathcal{J} \right\}$$

- Then $\nu \circ M[q] \circ \mu : \mathbb{D}_u \to \mathbb{T}_u$ is the required translation of $M[q] : \mathbb{D}_n \to \mathbb{T}_n$.



Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.
Solution.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.
Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q =$ R with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.
Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q =$ R with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \mathsf{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \text{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \mathrm{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \mathbf{adom}(q)$.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \mathrm{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \mathbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \text{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\textbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \textbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \textbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i = c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \mathrm{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \mathbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i = c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$. DI, since $c \in \mathbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \mathsf{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \mathbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i = c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$. DI, since $c \in \mathbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses. Equivalent, since q' and $\varphi_{q'}$ are equivalent and $x_{a_i} = c$ for all answers.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \mathrm{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \mathbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i = c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$. DI, since $c \in \mathbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses. Equivalent, since q' and $\varphi_{q'}$ are equivalent and $x_{a_i} = c$ for all answers.
- If $q = \sigma_{a_i = a_j}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx x_{a_j})$.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q =$ R with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\textbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \textbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \textbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i = c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$. DI, since $c \in \textbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses. Equivalent, since q' and $\varphi_{q'}$ are equivalent and $x_{a_i} = c$ for all answers.
- If $q = \sigma_{a_i = a_j}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx x_{a_j})$. Analogous.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \mathrm{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \mathbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i = c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$. DI, since $c \in \mathbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses. Equivalent, since q' and $\varphi_{q'}$ are equivalent and $x_{a_i} = c$ for all answers.
- If $q = \sigma_{a_i = a_j}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx x_{a_j})$. Analogous.
- If $q = \delta_{b_1, \ldots, b_n \to a_1, \ldots, a_n} q'$, then $\varphi_q = \exists y_{b_1}, \ldots, y_{b_n}. (x_{a_1} \approx y_{b_1}) \wedge \ldots \wedge (x_{a_n} \approx y_{b_n}) \wedge \varphi_{q'}[y_{B_1}, \ldots, y_{B_n}]$.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \mathrm{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \mathbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i = c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$. DI, since $c \in \mathbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses. Equivalent, since q' and $\varphi_{q'}$ are equivalent and $x_{a_i} = c$ for all answers.
- If $q = \sigma_{a_i = a_j}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx x_{a_j})$. Analogous.
- If $q = \delta_{b_1,\ldots,b_n \to a_1,\ldots,a_n} q'$, then $\varphi_q = \exists y_{b_1}, \ldots, y_{b_n}.\, (x_{a_1} \approx y_{b_1}) \wedge \ldots \wedge (x_{a_n} \approx y_{b_n}) \wedge \varphi_{q'}[y_{B_1}, \ldots, y_{B_n}]$. DI, since $\{y_{b_1}, \ldots, y_{b_n}\} = \{y_{B_1}, \ldots, y_{B_n}\}$, and $\varphi_{q'}$ is DI by induction. Thus, the values of $y_{b_1}, \ldots, y_{b_n}$ are DI, which restrict the values of $x_{a_1}, \ldots, x_{a_n}$.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q =$ R with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\textbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \textbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \textbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i = c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$. DI, since $c \in \textbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses. Equivalent, since q' and $\varphi_{q'}$ are equivalent and $x_{a_i} = c$ for all answers.
- If $q = \sigma_{a_i = a_j}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx x_{a_j})$. Analogous.
- If $q = \delta_{b_1, \ldots, b_n \to a_1, \ldots, a_n} q'$, then $\varphi_q = \exists y_{b_1}, \ldots, y_{b_n}. (x_{a_1} \approx y_{b_1}) \wedge \ldots \wedge (x_{a_n} \approx y_{b_n}) \wedge \varphi_{q'}[y_{B_1}, \ldots, y_{B_n}]$. DI, since $\{y_{b_1}, \ldots, y_{b_n}\} = \{y_{B_1}, \ldots, y_{B_n}\}$, and $\varphi_{q'}$ is DI by induction. Thus, the values of $y_{b_1}, \ldots, y_{b_n}$ are DI, which restrict the values of $x_{a_1}, \ldots, x_{a_n}$. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction, and $\mathcal{I} \models \varphi_q(c_{a_1}, \ldots, c_{a_n})$ iff $\mathcal{I} \models \varphi_{q'}(c_{B_1}, \ldots, c_{B_n})$.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \mathrm{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \mathbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i = c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$. DI, since $c \in \mathbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses. Equivalent, since q' and $\varphi_{q'}$ are equivalent and $x_{a_i} = c$ for all answers.
- If $q = \sigma_{a_i = a_j}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx x_{a_j})$. Analogous.
- If $q = \delta_{b_1,\ldots,b_n \to a_1,\ldots,a_n} q'$, then $\varphi_q = \exists y_{b_1}, \ldots, y_{b_n}. (x_{a_1} \approx y_{b_1}) \wedge \ldots \wedge (x_{a_n} \approx y_{b_n}) \wedge \varphi_{q'}[y_{B_1}, \ldots, y_{B_n}]$. DI, since $\{y_{b_1}, \ldots, y_{b_n}\} = \{y_{B_1}, \ldots, y_{B_n}\}$, and $\varphi_{q'}$ is DI by induction. Thus, the values of $y_{b_1}, \ldots, y_{b_n}$ are DI, which restrict the values of $x_{a_1}, \ldots, x_{a_n}$. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction, and $\mathcal{I} \models \varphi_q(c_{a_1}, \ldots, c_{a_n})$ iff $\mathcal{I} \models \varphi_{q'}(c_{B_1}, \ldots, c_{B_n})$.
- If $q = \pi_{a_1,\ldots,a_n}(q')$ for a subquery $q'[b_1, \ldots, b_m]$ with $\{b_1, \ldots, b_m\} = \{a_1, \ldots, a_n\} \cup \{c_1, \ldots, c_k\}$, then $\varphi_q = \exists x_{c_1}, \ldots, x_{c_k}. \varphi_{q'}$.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.
Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \text{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \mathbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i = c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$. DI, since $c \in \mathbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses. Equivalent, since q' and $\varphi_{q'}$ are equivalent and $x_{a_i} = c$ for all answers.
- If $q = \sigma_{a_i = a_j}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx x_{a_j})$. Analogous.
- If $q = \delta_{b_1, \ldots, b_n \to a_1, \ldots, a_n} q'$, then $\varphi_q = \exists y_{b_1}, \ldots, y_{b_n}. \, (x_{a_1} \approx y_{b_1}) \wedge \ldots \wedge (x_{a_n} \approx y_{b_n}) \wedge \varphi_{q'}[y_{B_1}, \ldots, y_{B_n}]$. DI, since $\{y_{b_1}, \ldots, y_{b_n}\} = \{y_{B_1}, \ldots, y_{B_n}\}$, and $\varphi_{q'}$ is DI by induction. Thus, the values of $y_{b_1}, \ldots, y_{b_n}$ are DI, which restrict the values of $x_{a_1}, \ldots, x_{a_n}$. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction, and $\mathcal{I} \models \varphi_q(c_{a_1}, \ldots, c_{a_n})$ iff $\mathcal{I} \models \varphi_{q'}(c_{B_1}, \ldots, c_{B_n})$.
- If $q = \pi_{a_1, \ldots, a_n}(q')$ for a subquery $q'[b_1, \ldots, b_m]$ with $\{b_1, \ldots, b_m\} = \{a_1, \ldots, a_n\} \cup \{c_1, \ldots, c_k\}$, then $\varphi_q = \exists x_{c_1}, \ldots, x_{c_k}. \, \varphi_{q'}$. DI, since all x_{c_i} occur in $\varphi_{q'}$, which is DI.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \mathrm{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \mathbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i=c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$. DI, since $c \in \mathbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses. Equivalent, since q' and $\varphi_{q'}$ are equivalent and $x_{a_i} = c$ for all answers.
- If $q = \sigma_{a_i=a_j}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx x_{a_j})$. Analogous.
- If $q = \delta_{b_1,\ldots,b_n \to a_1,\ldots,a_n} q'$, then $\varphi_q = \exists y_{b_1}, \ldots, y_{b_n}. (x_{a_1} \approx y_{b_1}) \wedge \ldots \wedge (x_{a_n} \approx y_{b_n}) \wedge \varphi_{q'}[y_{B_1}, \ldots, y_{B_n}]$. DI, since $\{y_{b_1}, \ldots, y_{b_n}\} = \{y_{B_1}, \ldots, y_{B_n}\}$, and $\varphi_{q'}$ is DI by induction. Thus, the values of $y_{b_1}, \ldots, y_{b_n}$ are DI, which restrict the values of $x_{a_1}, \ldots, x_{a_n}$. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction, and $\mathcal{I} \models \varphi_q(c_{a_1}, \ldots, c_{a_n})$ iff $\mathcal{I} \models \varphi_{q'}(c_{B_1}, \ldots, c_{B_n})$.
- If $q = \pi_{a_1,\ldots,a_n}(q')$ for a subquery $q'[b_1, \ldots, b_m]$ with $\{b_1, \ldots, b_m\} = \{a_1, \ldots, a_n\} \cup \{c_1, \ldots, c_k\}$, then $\varphi_q = \exists x_{c_1}, \ldots, x_{c_k}. \varphi_{q'}$. DI, since all x_{c_i} occur in $\varphi_{q'}$, which is DI. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction and $\{ a_1, \ldots, a_n \} = \{ b_1, \ldots, b_m \} \setminus \{ c_1, \ldots, c_k \}$.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \mathsf{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \mathbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i = c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$. DI, since $c \in \mathbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses. Equivalent, since q' and $\varphi_{q'}$ are equivalent and $x_{a_i} = c$ for all answers.
- If $q = \sigma_{a_i = a_j}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx x_{a_j})$. Analogous.
- If $q = \delta_{b_1, \ldots, b_n \to a_1, \ldots, a_n} q'$, then $\varphi_q = \exists y_{b_1}, \ldots, y_{b_n}. (x_{a_1} \approx y_{b_1}) \wedge \ldots \wedge (x_{a_n} \approx y_{b_n}) \wedge \varphi_{q'}[y_{B_1}, \ldots, y_{B_n}]$. DI, since $\{y_{b_1}, \ldots, y_{b_n}\} = \{y_{B_1}, \ldots, y_{B_n}\}$, and $\varphi_{q'}$ is DI by induction. Thus, the values of $y_{b_1}, \ldots, y_{b_n}$ are DI, which restrict the values of $x_{a_1}, \ldots, x_{a_n}$. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction, and $\mathcal{I} \models \varphi_q(c_{a_1}, \ldots, c_{a_n})$ iff $\mathcal{I} \models \varphi_{q'}(c_{B_1}, \ldots, c_{B_n})$.
- If $q = \pi_{a_1, \ldots, a_n}(q')$ for a subquery $q'[b_1, \ldots, b_m]$ with $\{b_1, \ldots, b_m\} = \{a_1, \ldots, a_n\} \cup \{c_1, \ldots, c_k\}$, then $\varphi_q = \exists x_{c_1}, \ldots, x_{c_k}. \varphi_{q'}$. DI, since all x_{c_i} occur in $\varphi_{q'}$, which is DI. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction and $\{ a_1, \ldots, a_n \} = \{ b_1, \ldots, b_m \} \setminus \{ c_1, \ldots, c_k \}$.
- If $q = q_1 \bowtie q_2$, then $\varphi_q = \varphi_{q_1} \wedge \varphi_{q_2}$.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q =$ R with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \mathbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i = c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$. DI, since $c \in \mathbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses. Equivalent, since q' and $\varphi_{q'}$ are equivalent and $x_{a_i} = c$ for all answers.
- If $q = \sigma_{a_i = a_j}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx x_{a_j})$. Analogous.
- If $q = \delta_{b_1,\ldots,b_n \to a_1,\ldots,a_n} q'$, then $\varphi_q = \exists y_{b_1}, \ldots, y_{b_n}. (x_{a_1} \approx y_{b_1}) \wedge \ldots \wedge (x_{a_n} \approx y_{b_n}) \wedge \varphi_{q'}[y_{B_1}, \ldots, y_{B_n}]$. DI, since $\{y_{b_1}, \ldots, y_{b_n}\} = \{y_{B_1}, \ldots, y_{B_n}\}$, and $\varphi_{q'}$ is DI by induction. Thus, the values of $y_{b_1}, \ldots, y_{b_n}$ are DI, which restrict the values of $x_{a_1}, \ldots, x_{a_n}$. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction, and $\mathcal{I} \models \varphi_q(c_{a_1}, \ldots, c_{a_n})$ iff $\mathcal{I} \models \varphi_{q'}(c_{B_1}, \ldots, c_{B_n})$.
- If $q = \pi_{a_1,\ldots,a_n}(q')$ for a subquery $q'[b_1, \ldots, b_m]$ with $\{b_1, \ldots, b_m\} = \{a_1, \ldots, a_n\} \cup \{c_1, \ldots, c_k\}$, then $\varphi_q = \exists x_{c_1}, \ldots, x_{c_k}. \varphi_{q'}$. DI, since all x_{c_i} occur in $\varphi_{q'}$, which is DI. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction and $\{ a_1, \ldots, a_n \} = \{ b_1, \ldots, b_m \} \setminus \{ c_1, \ldots, c_k \}$.
- If $q = q_1 \bowtie q_2$, then $\varphi_q = \varphi_{q_1} \wedge \varphi_{q_2}$. DI, since all variables occur in φ_{q_1} or φ_{q_2}, which are DI by induction.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \mathsf{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\textbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \textbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \textbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i=c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$. DI, since $c \in \textbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses. Equivalent, since q' and $\varphi_{q'}$ are equivalent and $x_{a_i} = c$ for all answers.
- If $q = \sigma_{a_i=a_j}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx x_{a_j})$. Analogous.
- If $q = \delta_{b_1,\ldots,b_n \to a_1,\ldots,a_n} q'$, then $\varphi_q = \exists y_{b_1}, \ldots, y_{b_n}. (x_{a_1} \approx y_{b_1}) \wedge \ldots \wedge (x_{a_n} \approx y_{b_n}) \wedge \varphi_{q'}[y_{B_1}, \ldots, y_{B_n}]$. DI, since $\{y_{b_1}, \ldots, y_{b_n}\} = \{y_{B_1}, \ldots, y_{B_n}\}$, and $\varphi_{q'}$ is DI by induction. Thus, the values of $y_{b_1}, \ldots, y_{b_n}$ are DI, which restrict the values of $x_{a_1}, \ldots, x_{a_n}$. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction, and $\mathcal{I} \models \varphi_q(c_{a_1}, \ldots, c_{a_n})$ iff $\mathcal{I} \models \varphi_{q'}(c_{B_1}, \ldots, c_{B_n})$.
- If $q = \pi_{a_1,\ldots,a_n}(q')$ for a subquery $q'[b_1, \ldots, b_m]$ with $\{b_1, \ldots, b_m\} = \{a_1, \ldots, a_n\} \cup \{c_1, \ldots, c_k\}$, then $\varphi_q = \exists x_{c_1}, \ldots, x_{c_k}. \varphi_{q'}$. DI, since all x_{c_i} occur in $\varphi_{q'}$, which is DI. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction and $\{ a_1, \ldots, a_n \} = \{ b_1, \ldots, b_m \} \setminus \{ c_1, \ldots, c_k \}$.
- If $q = q_1 \bowtie q_2$, then $\varphi_q = \varphi_{q_1} \wedge \varphi_{q_2}$. DI, since all variables occur in φ_{q_1} or φ_{q_2}, which are DI by induction. Equivalent, since any answer to q contains answers to q_1 and q_2.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q =$ R with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \mathbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i = c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$. DI, since $c \in \mathbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses. Equivalent, since q' and $\varphi_{q'}$ are equivalent and $x_{a_i} = c$ for all answers.
- If $q = \sigma_{a_i = a_j}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx x_{a_j})$. Analogous.
- If $q = \delta_{b_1, \ldots, b_n \to a_1, \ldots, a_n} q'$, then $\varphi_q = \exists y_{b_1}, \ldots, y_{b_n}. (x_{a_1} \approx y_{b_1}) \wedge \ldots \wedge (x_{a_n} \approx y_{b_n}) \wedge \varphi_{q'}[y_{B_1}, \ldots, y_{B_n}]$. DI, since $\{y_{b_1}, \ldots, y_{b_n}\} = \{y_{B_1}, \ldots, y_{B_n}\}$, and $\varphi_{q'}$ is DI by induction. Thus, the values of $y_{b_1}, \ldots, y_{b_n}$ are DI, which restrict the values of $x_{a_1}, \ldots, x_{a_n}$. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction, and $\mathcal{I} \models \varphi_q(c_{a_1}, \ldots, c_{a_n})$ iff $\mathcal{I} \models \varphi_{q'}(c_{B_1}, \ldots, c_{B_n})$.
- If $q = \pi_{a_1, \ldots, a_n}(q')$ for a subquery $q'[b_1, \ldots, b_m]$ with $\{b_1, \ldots, b_m\} = \{a_1, \ldots, a_n\} \cup \{c_1, \ldots, c_k\}$, then $\varphi_q = \exists x_{c_1}, \ldots, x_{c_k}. \varphi_{q'}$. DI, since all x_{c_i} occur in $\varphi_{q'}$, which is DI. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction and $\{ a_1, \ldots, a_n \} = \{ b_1, \ldots, b_m \} \setminus \{ c_1, \ldots, c_k \}$.
- If $q = q_1 \bowtie q_2$, then $\varphi_q = \varphi_{q_1} \wedge \varphi_{q_2}$. DI, since all variables occur in φ_{q_1} or φ_{q_2}, which are DI by induction. Equivalent, since any answer to q contains answers to q_1 and q_2.
- If $q = q_1 \cup q_2$, then $\varphi_q = \varphi_{q_1} \vee \varphi_{q_2}$.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.
Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \mathsf{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \mathbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i = c}(q')$, then $\varphi_q = \varphi_{q'} \land (x_{a_i} \approx c)$. DI, since $c \in \mathbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses. Equivalent, since q' and $\varphi_{q'}$ are equivalent and $x_{a_i} = c$ for all answers.
- If $q = \sigma_{a_i = a_j}(q')$, then $\varphi_q = \varphi_{q'} \land (x_{a_i} \approx x_{a_j})$. Analogous.
- If $q = \delta_{b_1, \ldots, b_n \to a_1, \ldots, a_n} q'$, then $\varphi_q = \exists y_{b_1}, \ldots, y_{b_n}. (x_{a_1} \approx y_{b_1}) \land \ldots \land (x_{a_n} \approx y_{b_n}) \land \varphi_{q'}[y_{B_1}, \ldots, y_{B_n}]$. DI, since $\{y_{b_1}, \ldots, y_{b_n}\} = \{y_{B_1}, \ldots, y_{B_n}\}$, and $\varphi_{q'}$ is DI by induction. Thus, the values of $y_{b_1}, \ldots, y_{b_n}$ are DI, which restrict the values of $x_{a_1}, \ldots, x_{a_n}$. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction, and $\mathcal{I} \models \varphi_q(c_{a_1}, \ldots, c_{a_n})$ iff $\mathcal{I} \models \varphi_{q'}(c_{B_1}, \ldots, c_{B_n})$.
- If $q = \pi_{a_1, \ldots, a_n}(q')$ for a subquery $q'[b_1, \ldots, b_m]$ with $\{b_1, \ldots, b_m\} = \{a_1, \ldots, a_n\} \cup \{c_1, \ldots, c_k\}$, then $\varphi_q = \exists x_{c_1}, \ldots, x_{c_k}. \varphi_{q'}$. DI, since all x_{c_i} occur in $\varphi_{q'}$, which is DI. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction and $\{ a_1, \ldots, a_n \} = \{ b_1, \ldots, b_m \} \setminus \{ c_1, \ldots, c_k \}$.
- If $q = q_1 \bowtie q_2$, then $\varphi_q = \varphi_{q_1} \land \varphi_{q_2}$. DI, since all variables occur in φ_{q_1} or φ_{q_2}, which are DI by induction. Equivalent, since any answer to q contains answers to q_1 and q_2.
- If $q = q_1 \cup q_2$, then $\varphi_q = \varphi_{q_1} \lor \varphi_{q_2}$. DI, since all variables occur in φ_{q_1}.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \mathsf{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\textbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \textbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \textbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i = c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$. DI, since $c \in \textbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses. Equivalent, since q' and $\varphi_{q'}$ are equivalent and $x_{a_i} = c$ for all answers.
- If $q = \sigma_{a_i = a_j}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx x_{a_j})$. Analogous.
- If $q = \delta_{b_1, \ldots, b_n \to a_1, \ldots, a_n} q'$, then $\varphi_q = \exists y_{b_1}, \ldots, y_{b_n}. (x_{a_1} \approx y_{b_1}) \wedge \ldots \wedge (x_{a_n} \approx y_{b_n}) \wedge \varphi_{q'}[y_{B_1}, \ldots, y_{B_n}]$. DI, since $\{y_{b_1}, \ldots, y_{b_n}\} = \{y_{B_1}, \ldots, y_{B_n}\}$, and $\varphi_{q'}$ is DI by induction. Thus, the values of $y_{b_1}, \ldots, y_{b_n}$ are DI, which restrict the values of $x_{a_1}, \ldots, x_{a_n}$. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction, and $\mathcal{I} \models \varphi_q(c_{a_1}, \ldots, c_{a_n})$ iff $\mathcal{I} \models \varphi_{q'}(c_{B_1}, \ldots, c_{B_n})$.
- If $q = \pi_{a_1, \ldots, a_n}(q')$ for a subquery $q'[b_1, \ldots, b_m]$ with $\{b_1, \ldots, b_m\} = \{a_1, \ldots, a_n\} \cup \{c_1, \ldots, c_k\}$, then $\varphi_q = \exists x_{c_1}, \ldots, x_{c_k}. \varphi_{q'}$. DI, since all x_{c_i} occur in $\varphi_{q'}$, which is DI. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction and $\{ a_1, \ldots, a_n \} = \{ b_1, \ldots, b_m \} \setminus \{ c_1, \ldots, c_k \}$.
- If $q = q_1 \bowtie q_2$, then $\varphi_q = \varphi_{q_1} \wedge \varphi_{q_2}$. DI, since all variables occur in φ_{q_1} or φ_{q_2}, which are DI by induction. Equivalent, since any answer to q contains answers to q_1 and q_2.
- If $q = q_1 \cup q_2$, then $\varphi_q = \varphi_{q_1} \vee \varphi_{q_2}$. DI, since all variables occur in φ_{q_1}. Clearly equivalent.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.
Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q = \mathsf{R}$ with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \mathbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i = c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$. DI, since $c \in \mathbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses. Equivalent, since q' and $\varphi_{q'}$ are equivalent and $x_{a_i} = c$ for all answers.
- If $q = \sigma_{a_i = a_j}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx x_{a_j})$. Analogous.
- If $q = \delta_{b_1,\ldots,b_n \to a_1,\ldots,a_n} q'$, then $\varphi_q = \exists y_{b_1}, \ldots, y_{b_n}. (x_{a_1} \approx y_{b_1}) \wedge \ldots \wedge (x_{a_n} \approx y_{b_n}) \wedge \varphi_{q'}[y_{B_1}, \ldots, y_{B_n}]$. DI, since $\{y_{b_1}, \ldots, y_{b_n}\} = \{y_{B_1}, \ldots, y_{B_n}\}$, and $\varphi_{q'}$ is DI by induction. Thus, the values of $y_{b_1}, \ldots, y_{b_n}$ are DI, which restrict the values of $x_{a_1}, \ldots, x_{a_n}$. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction, and $\mathcal{I} \models \varphi_q(c_{a_1}, \ldots, c_{a_n})$ iff $\mathcal{I} \models \varphi_{q'}(c_{B_1}, \ldots, c_{B_n})$.
- If $q = \pi_{a_1,\ldots,a_n}(q')$ for a subquery $q'[b_1, \ldots, b_m]$ with $\{b_1, \ldots, b_m\} = \{a_1, \ldots, a_n\} \cup \{c_1, \ldots, c_k\}$, then $\varphi_q = \exists x_{c_1}, \ldots, x_{c_k}. \varphi_{q'}$. DI, since all x_{c_i} occur in $\varphi_{q'}$, which is DI. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction and $\{ a_1, \ldots, a_n \} = \{ b_1, \ldots, b_m \} \setminus \{ c_1, \ldots, c_k \}$.
- If $q = q_1 \bowtie q_2$, then $\varphi_q = \varphi_{q_1} \wedge \varphi_{q_2}$. DI, since all variables occur in φ_{q_1} or φ_{q_2}, which are DI by induction. Equivalent, since any answer to q contains answers to q_1 and q_2.
- If $q = q_1 \cup q_2$, then $\varphi_q = \varphi_{q_1} \vee \varphi_{q_2}$. DI, since all variables occur in φ_{q_1}. Clearly equivalent.
- If $q = q_1 - q_2$, then $\varphi_q = \varphi_{q_1} \wedge \neg\varphi_{q_2}$.

Exercise 4

Exercise. Complete the proof that $RA_{\text{named}} \sqsubseteq DI_{\text{unnamed}}$ by showing that the results of the transformation are (a) *domain independent* and (b) equivalent to the input query. In each case, show that the claimed property holds true for each case of the recursive construction under the assumption (induction hypothesis) that it has been established for all subqueries. Use the mappings from the previous exercise to compare named and unnamed results.

Solution. We show domain independence and equivalence by induction on the structure of the RA query q.

- If $q =$ R with signature $R[a_1, \ldots, a_n]$, then $\varphi_q = R(x_{a_1}, \ldots, x_{a_n})$. DI, since the values of x_{a_i} belong to $\mathbf{adom}(\{ R^{\mathcal{I}} \}) \subseteq \mathbf{adom}(\mathcal{I})$. Equivalent, since $\mathcal{I} \models R(c_1, \ldots, c_n)$ iff $\{ a_1 \mapsto c_1, \ldots, a_n \mapsto c_n \} \in M[q](\mathcal{I})$.
- If $q = \{\{a_1 \mapsto c\}\}$, then $\varphi_q = (x_{a_1} \approx c)$. DI, since $c \in \mathbf{adom}(q)$. Equivalent, since $\{ \langle c \rangle \}$ is the only result.
- If $q = \sigma_{a_i = c}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx c)$. DI, since $c \in \mathbf{adom}(q)$, and x_{a_i} occurs in $\varphi_{q'}$, which is DI by the induction hypotheses. Equivalent, since q' and $\varphi_{q'}$ are equivalent and $x_{a_i} = c$ for all answers.
- If $q = \sigma_{a_i = a_j}(q')$, then $\varphi_q = \varphi_{q'} \wedge (x_{a_i} \approx x_{a_j})$. Analogous.
- If $q = \delta_{b_1, \ldots, b_n \to a_1, \ldots, a_n} q'$, then $\varphi_q = \exists y_{b_1}, \ldots, y_{b_n}. (x_{a_1} \approx y_{b_1}) \wedge \ldots \wedge (x_{a_n} \approx y_{b_n}) \wedge \varphi_{q'}[y_{B_1}, \ldots, y_{B_n}]$. DI, since $\{y_{b_1}, \ldots, y_{b_n}\} = \{y_{B_1}, \ldots, y_{B_n}\}$, and $\varphi_{q'}$ is DI by induction. Thus, the values of $y_{b_1}, \ldots, y_{b_n}$ are DI, which restrict the values of $x_{a_1}, \ldots, x_{a_n}$. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction, and $\mathcal{I} \models \varphi_q(c_{a_1}, \ldots, c_{a_n})$ iff $\mathcal{I} \models \varphi_{q'}(c_{B_1}, \ldots, c_{B_n})$.
- If $q = \pi_{a_1, \ldots, a_n}(q')$ for a subquery $q'[b_1, \ldots, b_m]$ with $\{b_1, \ldots, b_m\} = \{a_1, \ldots, a_n\} \cup \{c_1, \ldots, c_k\}$, then $\varphi_q = \exists x_{c_1}, \ldots, x_{c_k}. \varphi_{q'}$. DI, since all x_{c_i} occur in $\varphi_{q'}$, which is DI. Equivalent, since q' and $\varphi_{q'}$ are equivalent by induction and $\{ a_1, \ldots, a_n \} = \{ b_1, \ldots, b_m \} \setminus \{ c_1, \ldots, c_k \}$.
- If $q = q_1 \bowtie q_2$, then $\varphi_q = \varphi_{q_1} \wedge \varphi_{q_2}$. DI, since all variables occur in φ_{q_1} or φ_{q_2}, which are DI by induction. Equivalent, since any answer to q contains answers to q_1 and q_2.
- If $q = q_1 \cup q_2$, then $\varphi_q = \varphi_{q_1} \vee \varphi_{q_2}$. DI, since all variables occur in φ_{q_1}. Clearly equivalent.
- If $q = q_1 - q_2$, then $\varphi_q = \varphi_{q_1} \wedge \neg \varphi_{q_2}$. Analogous.

Exercise 5

Exercise. Consider a binary predicate R and the AD_{unnamed} query

$$\varphi[x, y] = \neg(R(x, y) \wedge R(y, x)).$$

Use the construction from the lecture to express it as an RA_{named} query.

Exercise 5

Exercise. Consider a binary predicate R and the AD_{unnamed} query

$$\varphi[x, y] = \neg(R(x, y) \wedge R(y, x)).$$

Use the construction from the lecture to express it as an RA_{named} query.
Solution.

Exercise 5

Exercise. Consider a binary predicate R and the AD_{unnamed} query

$$\varphi[x, y] = \neg(R(x, y) \wedge R(y, x)).$$

Use the construction from the lecture to express it as an RA_{named} query.
Solution.

Definition (Lecture 2, Slide 22/23, excerpt)

Consider an AD query $q = \varphi[x_1, \ldots, x_n]$. For every attribute name a, there is an RA expression $E_{a,\textbf{adom}}$ with $E_{a,\textbf{adom}}(\mathcal{I}) = \{\{a \mapsto c\} | c \in \textbf{adom}(\mathcal{I}, q)\}$. For every variable x, we use a fresh, distinct attribute name a_x.

- If $\varphi = R(t_1, \ldots, t_m)$ with signature $R[a_1, \ldots, a_m]$, variables $x_1 = t_{v_1}, \ldots, x_n = t_{v_n}$ and constants $c_1 = t_{w_1}, \ldots, c_k = t_{w_k}$, then $E_\varphi = \delta_{a_{v_1} \ldots a_{v_n} \to a_{x_1} \ldots a_{x_n}}(\sigma_{a_{w_1}=c_1}(\ldots \sigma_{a_{w_k}=c_k}(R) \ldots))$;
- if $\varphi = \neg\psi$, then $E_\varphi = (E_{a_{x_1},\textbf{adom}} \bowtie \ldots \bowtie E_{a_{x_n},\textbf{adom}}) - E_\psi$; and
- if $\varphi = \varphi_1 \wedge \varphi_2$, then $E_\varphi = E_{\varphi_1} \bowtie E_{\varphi_2}$.

Exercise 5

Exercise. Consider a binary predicate R and the AD_{unnamed} query

$$\varphi[x, y] = \neg(R(x, y) \wedge R(y, x)).$$

Use the construction from the lecture to express it as an RA_{named} query.
Solution.

Definition (Lecture 2, Slide 22/23, excerpt)

Consider an AD query $q = \varphi[x_1, \ldots, x_n]$. For every attribute name a, there is an RA expression $E_{a,\textbf{adom}}$ with $E_{a,\textbf{adom}}(\mathcal{I}) = \{\{a \mapsto c\} | c \in \textbf{adom}(\mathcal{I}, q)\}$. For every variable x, we use a fresh, distinct attribute name a_x.

- If $\varphi = R(t_1, \ldots, t_m)$ with signature $R[a_1, \ldots, a_m]$, variables $x_1 = t_{v_1}, \ldots, x_n = t_{v_n}$ and constants $c_1 = t_{w_1}, \ldots, c_k = t_{w_k}$, then $E_\varphi = \delta_{a_{v_1} \ldots a_{v_n} \to a_{x_1} \ldots a_{x_n}}(\sigma_{a_{w_1} = c_1}(\ldots \sigma_{a_{w_k} = c_k}(R) \ldots))$;
- if $\varphi = \neg\psi$, then $E_\varphi = (E_{a_{x_1},\textbf{adom}} \bowtie \ldots \bowtie E_{a_{x_n},\textbf{adom}}) - E_\psi$; and
- if $\varphi = \varphi_1 \wedge \varphi_2$, then $E_\varphi = E_{\varphi_1} \bowtie E_{\varphi_2}$.

$$\varphi[x, y] = \neg(R(x, y) \wedge R(y, x))$$

Exercise 5

Exercise. Consider a binary predicate R and the AD_{unnamed} query

$$\varphi[x, y] = \neg(R(x, y) \wedge R(y, x)).$$

Use the construction from the lecture to express it as an RA_{named} query.
Solution.

Definition (Lecture 2, Slide 22/23, excerpt)

Consider an AD query $q = \varphi[x_1, \ldots, x_n]$. For every attribute name a, there is an RA expression $E_{a,\mathbf{adom}}$ with $E_{a,\mathbf{adom}}(\mathcal{I}) = \{\{a \mapsto c\} | c \in \mathbf{adom}(\mathcal{I}, q)\}$. For every variable x, we use a fresh, distinct attribute name a_x.

- If $\varphi = R(t_1, \ldots, t_m)$ with signature $R[a_1, \ldots, a_m]$, variables $x_1 = t_{v_1}, \ldots, x_n = t_{v_n}$ and constants $c_1 = t_{w_1}, \ldots, c_k = t_{w_k}$, then $E_\varphi = \delta_{a_{v_1} \ldots a_{v_n} \to a_{x_1} \ldots a_{x_n}}(\sigma_{a_{w_1} = c_1}(\ldots \sigma_{a_{w_k} = c_k}(R) \ldots))$;
- if $\varphi = \neg\psi$, then $E_\varphi = (E_{a_{x_1},\mathbf{adom}} \bowtie \ldots \bowtie E_{a_{x_n},\mathbf{adom}}) - E_\psi$; and
- if $\varphi = \varphi_1 \wedge \varphi_2$, then $E_\varphi = E_{\varphi_1} \bowtie E_{\varphi_2}$.

$$\begin{aligned} \varphi[x, y] &= \neg(R(x, y) \wedge R(y, x)) \\ &= (E_{a_x,\mathbf{adom}} \bowtie E_{a_y,\mathbf{adom}}) - E_{R(x,y) \wedge R(y,x)} \end{aligned}$$

Exercise 5

Exercise. Consider a binary predicate R and the AD_{unnamed} query

$$\varphi[x, y] = \neg(R(x, y) \wedge R(y, x)).$$

Use the construction from the lecture to express it as an RA_{named} query.
Solution.

Definition (Lecture 2, Slide 22/23, excerpt)

Consider an AD query $q = \varphi[x_1, \ldots, x_n]$. For every attribute name a, there is an RA expression $E_{a,\mathbf{adom}}$ with $E_{a,\mathbf{adom}}(\mathcal{I}) = \{\{a \mapsto c\} | c \in \mathbf{adom}(\mathcal{I}, q)\}$. For every variable x, we use a fresh, distinct attribute name a_x.

- If $\varphi = R(t_1, \ldots, t_m)$ with signature $R[a_1, \ldots, a_m]$, variables $x_1 = t_{v_1}, \ldots, x_n = t_{v_n}$ and constants $c_1 = t_{w_1}, \ldots, c_k = t_{w_k}$, then $E_\varphi = \delta_{a_{v_1} \ldots a_{v_n} \to a_{x_1} \ldots a_{x_n}}(\sigma_{a_{w_1}=c_1}(\ldots \sigma_{a_{w_k}=c_k}(R) \ldots))$;
- if $\varphi = \neg\psi$, then $E_\varphi = (E_{a_{x_1},\mathbf{adom}} \bowtie \ldots \bowtie E_{a_{x_n},\mathbf{adom}}) - E_\psi$; and
- if $\varphi = \varphi_1 \wedge \varphi_2$, then $E_\varphi = E_{\varphi_1} \bowtie E_{\varphi_2}$.

$$\begin{aligned}
\varphi[x, y] &= \neg(R(x, y) \wedge R(y, x)) \\
&= (E_{a_x,\mathbf{adom}} \bowtie E_{a_y,\mathbf{adom}}) - E_{R(x,y) \wedge R(y,x)} \\
&= (E_{a_x,\mathbf{adom}} \bowtie E_{a_y,\mathbf{adom}}) - \big(E_{R(x,y)} \bowtie E_{R(y,x)}\big)
\end{aligned}$$

Exercise 5

Exercise. Consider a binary predicate R and the AD_{unnamed} query

$$\varphi[x,y] = \neg(R(x,y) \wedge R(y,x)).$$

Use the construction from the lecture to express it as an RA_{named} query.
Solution.

Definition (Lecture 2, Slide 22/23, excerpt)

Consider an AD query $q = \varphi[x_1,\ldots,x_n]$. For every attribute name a, there is an RA expression $E_{a,\textbf{adom}}$ with $E_{a,\textbf{adom}}(\mathcal{I}) = \{\{a \mapsto c\} | c \in \textbf{adom}(\mathcal{I},q)\}$. For every variable x, we use a fresh, distinct attribute name a_x.

- If $\varphi = R(t_1,\ldots,t_m)$ with signature $R[a_1,\ldots,a_m]$, variables $x_1 = t_{v_1},\ldots,x_n = t_{v_n}$ and constants $c_1 = t_{w_1},\ldots,c_k = t_{w_k}$, then $E_\varphi = \delta_{a_{v_1}\ldots a_{v_n}\to a_{x_1}\ldots a_{x_n}}(\sigma_{a_{w_1}=c_1}(\ldots\sigma_{a_{w_k}=c_k}(R)\ldots))$;
- if $\varphi = \neg\psi$, then $E_\varphi = (E_{a_{x_1},\textbf{adom}} \bowtie \ldots \bowtie E_{a_{x_n},\textbf{adom}}) - E_\psi$; and
- if $\varphi = \varphi_1 \wedge \varphi_2$, then $E_\varphi = E_{\varphi_1} \bowtie E_{\varphi_2}$.

$$\begin{aligned}
\varphi[x,y] &= \neg(R(x,y) \wedge R(y,x)) \\
&= (E_{a_x,\textbf{adom}} \bowtie E_{a_y,\textbf{adom}}) - E_{R(x,y)\wedge R(y,x)} \\
&= (E_{a_x,\textbf{adom}} \bowtie E_{a_y,\textbf{adom}}) - \left(E_{R(x,y)} \bowtie E_{R(y,x)}\right) \\
&= (E_{a_x,\textbf{adom}} \bowtie E_{a_y,\textbf{adom}}) - \left(\delta_{a_1,a_2\to a_x,a_y}(R) \bowtie \delta_{a_1,a_2\to a_y,a_x}(R)\right)
\end{aligned}$$

Exercise 6

Exercise. Complete the constructions for the proof of $AD \sqsubseteq RA$ given in the lecture.

1. Define the relational algebra expression $E_{a,\mathbf{adom}}$, such that $E_{a,\mathbf{adom}}(\mathcal{I}) = \{\{a \mapsto c\} \mid c \in \mathbf{adom}(\mathcal{I}, q)\}$ (assume that the query and the database schema are known).
2. Define the expressions E_φ for $\varphi = \varphi_1 \vee \varphi_2$ and $\varphi = \forall y.\psi$ in terms of expressions that have already been defined in the lecture.
3. Give a direct definition for the expression E_φ for $\varphi = \varphi_1 \vee \varphi_2 \equiv \neg(\neg\varphi_1 \wedge \neg\varphi_2)$.

Exercise 6

Exercise. Complete the constructions for the proof of $AD \sqsubseteq RA$ given in the lecture.

1. Define the relational algebra expression $E_{a,\textbf{adom}}$, such that $E_{a,\textbf{adom}}(\mathcal{I}) = \{\{a \mapsto c\} \mid c \in \textbf{adom}(\mathcal{I}, q)\}$ (assume that the query and the database schema are known).
2. Define the expressions E_φ for $\varphi = \varphi_1 \vee \varphi_2$ and $\varphi = \forall y.\psi$ in terms of expressions that have already been defined in the lecture.
3. Give a direct definition for the expression E_φ for $\varphi = \varphi_1 \vee \varphi_2 \equiv \neg(\neg\varphi_1 \wedge \neg\varphi_2)$.

Solution.

Exercise 6

Exercise. Complete the constructions for the proof of $AD \sqsubseteq RA$ given in the lecture.

1. Define the relational algebra expression $E_{a,\mathbf{adom}}$, such that $E_{a,\mathbf{adom}}(\mathcal{I}) = \{\{a \mapsto c\} \mid c \in \mathbf{adom}(\mathcal{I}, q)\}$ (assume that the query and the database schema are known).
2. Define the expressions E_φ for $\varphi = \varphi_1 \vee \varphi_2$ and $\varphi = \forall y.\psi$ in terms of expressions that have already been defined in the lecture.
3. Give a direct definition for the expression E_φ for $\varphi = \varphi_1 \vee \varphi_2 \equiv \neg(\neg\varphi_1 \wedge \neg\varphi_2)$.

Solution.

1. Assume that the database schema consists of tables $R_1, \ldots, R_\ell$ with table schemata $R_i[a^i_1, \ldots a^i_{|R_i|}]$. Let q be the query and define

$$E_{a,\mathbf{adom}} = \left(\bigcup_{i=1}^{\ell} \bigcup_{j=1}^{|R_i|} \delta_{a^i_j \to a}\big(\pi_{a^i_j}(R_i)\big) \right) \cup \Big\{ \{ a \mapsto c \} \,\Big|\, c \in \mathbf{adom}(q) \Big\}.$$

Exercise 6

Exercise. Complete the constructions for the proof of $AD \sqsubseteq RA$ given in the lecture.

1. Define the relational algebra expression $E_{a,\mathbf{adom}}$, such that $E_{a,\mathbf{adom}}(\mathcal{I}) = \{\{a \mapsto c\} \mid c \in \mathbf{adom}(\mathcal{I}, q)\}$ (assume that the query and the database schema are known).
2. Define the expressions E_φ for $\varphi = \varphi_1 \vee \varphi_2$ and $\varphi = \forall y.\psi$ in terms of expressions that have already been defined in the lecture.
3. Give a direct definition for the expression E_φ for $\varphi = \varphi_1 \vee \varphi_2 \equiv \neg(\neg\varphi_1 \wedge \neg\varphi_2)$.

Solution.

1. Assume that the database schema consists of tables $R_1, \ldots, R_\ell$ with table schemata $R_i[a^i_1, \ldots a^i_{|R_i|}]$. Let q be the query and define

$$E_{a,\mathbf{adom}} = \left(\bigcup_{i=1}^{\ell} \bigcup_{j=1}^{|R_i|} \delta_{a^i_j \to a}\Big(\pi_{a^i_j}(R_i)\Big)\right) \cup \Big\{ \{ a \mapsto c \} \Big| c \in \mathbf{adom}(q) \Big\}.$$

2.

$$E_{\varphi_1 \vee \varphi_2} = E_{\neg(\neg\varphi_1 \wedge \neg\varphi_2)}$$

Exercise 6

Exercise. Complete the constructions for the proof of $AD \sqsubseteq RA$ given in the lecture.

1. Define the relational algebra expression $E_{a,\,\mathbf{adom}}$, such that $E_{a,\,\mathbf{adom}}(\mathcal{I}) = \{\{a \mapsto c\} \mid c \in \mathbf{adom}(\mathcal{I}, q)\}$ (assume that the query and the database schema are known).
2. Define the expressions E_φ for $\varphi = \varphi_1 \vee \varphi_2$ and $\varphi = \forall y.\psi$ in terms of expressions that have already been defined in the lecture.
3. Give a direct definition for the expression E_φ for $\varphi = \varphi_1 \vee \varphi_2 \equiv \neg(\neg\varphi_1 \wedge \neg\varphi_2)$.

Solution.

1. Assume that the database schema consists of tables $R_1, \ldots, R_\ell$ with table schemata $R_i[a^i_1, \ldots a^i_{|R_i|}]$. Let q be the query and define

$$E_{a,\,\mathbf{adom}} = \left(\bigcup_{i=1}^{\ell} \bigcup_{j=1}^{|R_i|} \delta_{a^i_j \to a}\left(\pi_{a^i_j}(R_i)\right) \right) \cup \left\{ \{ a \mapsto c \} \,\middle|\, c \in \mathbf{adom}(q) \right\}.$$

2.

$$E_{\varphi_1 \vee \varphi_2} = E_{\neg(\neg\varphi_1 \wedge \neg\varphi_2)} \qquad\qquad E_{\forall y.\,\psi} = E_{\neg\exists y.\,\neg\psi}$$

Exercise 6

Exercise. Complete the constructions for the proof of $AD \sqsubseteq RA$ given in the lecture.

1. Define the relational algebra expression $E_{a,\mathbf{adom}}$, such that $E_{a,\mathbf{adom}}(\mathcal{I}) = \{\{a \mapsto c\} \mid c \in \mathbf{adom}(\mathcal{I}, q)\}$ (assume that the query and the database schema are known).
2. Define the expressions E_φ for $\varphi = \varphi_1 \vee \varphi_2$ and $\varphi = \forall y.\psi$ in terms of expressions that have already been defined in the lecture.
3. Give a direct definition for the expression E_φ for $\varphi = \varphi_1 \vee \varphi_2 \equiv \neg(\neg\varphi_1 \wedge \neg\varphi_2)$.

Solution.

1. Assume that the database schema consists of tables $R_1, \ldots, R_\ell$ with table schemata $R_i[a^i_1, \ldots a^i_{|R_i|}]$. Let q be the query and define

$$E_{a,\mathbf{adom}} = \left(\bigcup_{i=1}^{\ell} \bigcup_{j=1}^{|R_i|} \delta_{a^i_j \to a}\Big(\pi_{a^i_j}(R_i)\Big) \right) \cup \Big\{ \{ a \mapsto c \} \,\Big|\, c \in \mathbf{adom}(q) \Big\}.$$

2.

$$E_{\varphi_1 \vee \varphi_2} = E_{\neg(\neg\varphi_1 \wedge \neg\varphi_2)} \qquad E_{\forall y.\,\psi} = E_{\neg\exists y.\,\neg\psi}$$

3. Assume φ has free variables $x_1, \ldots, x_n$ with $\{x_1, \ldots, x_n\} = \{y_1, \ldots, y_\ell\} \cup \{z_1, \ldots, z_k\}$, where $y_1, \ldots, y_\ell$ are the free variables of φ_1 and $z_1, \ldots, z_k$ are the free variables of φ_2.

$E_{\varphi_1 \vee \varphi_2} =$

Exercise 6

Exercise. Complete the constructions for the proof of $AD \sqsubseteq RA$ given in the lecture.

1. Define the relational algebra expression $E_{a,\mathbf{adom}}$, such that $E_{a,\mathbf{adom}}(\mathcal{I}) = \{\{a \mapsto c\} \mid c \in \mathbf{adom}(\mathcal{I}, q)\}$ (assume that the query and the database schema are known).
2. Define the expressions E_φ for $\varphi = \varphi_1 \vee \varphi_2$ and $\varphi = \forall y.\psi$ in terms of expressions that have already been defined in the lecture.
3. Give a direct definition for the expression E_φ for $\varphi = \varphi_1 \vee \varphi_2 \equiv \neg(\neg\varphi_1 \wedge \neg\varphi_2)$.

Solution.

1. Assume that the database schema consists of tables $R_1, \ldots, R_\ell$ with table schemata $R_i[a^i_1, \ldots a^i_{|R_i|}]$. Let q be the query and define

$$E_{a,\mathbf{adom}} = \left(\bigcup_{i=1}^{\ell} \bigcup_{j=1}^{|R_i|} \delta_{a^i_j \to a}\left(\pi_{a^i_j}(R_i)\right) \right) \cup \left\{ \{ a \mapsto c \} \,\middle|\, c \in \mathbf{adom}(q) \right\}.$$

2.

$$E_{\varphi_1 \vee \varphi_2} = E_{\neg(\neg\varphi_1 \wedge \neg\varphi_2)} \qquad\qquad E_{\forall y.\, \psi} = E_{\neg\exists y.\, \neg\psi}$$

3. Assume φ has free variables $x_1, \ldots, x_n$ with $\{ x_1, \ldots, x_n \} = \{ y_1, \ldots, y_\ell \} \cup \{ z_1, \ldots, z_k \}$, where $y_1, \ldots, y_\ell$ are the free variables of φ_1 and $z_1, \ldots, z_k$ are the free variables of φ_2.

$$E_{\varphi_1 \vee \varphi_2} = E_{a_{x_1},\mathbf{adom}} \bowtie \cdots \bowtie E_{a_{x_n},\mathbf{adom}} - E_{\neg\varphi_1 \wedge \neg\varphi_2}$$

Exercise 6

Exercise. Complete the constructions for the proof of $AD \sqsubseteq RA$ given in the lecture.

1. Define the relational algebra expression $E_{a,\mathbf{adom}}$, such that $E_{a,\mathbf{adom}}(\mathcal{I}) = \{\{a \mapsto c\} \mid c \in \mathbf{adom}(\mathcal{I}, q)\}$ (assume that the query and the database schema are known).
2. Define the expressions E_φ for $\varphi = \varphi_1 \vee \varphi_2$ and $\varphi = \forall y.\psi$ in terms of expressions that have already been defined in the lecture.
3. Give a direct definition for the expression E_φ for $\varphi = \varphi_1 \vee \varphi_2 \equiv \neg(\neg\varphi_1 \wedge \neg\varphi_2)$.

Solution.

1. Assume that the database schema consists of tables $R_1, \ldots, R_\ell$ with table schemata $R_i[a^i_1, \ldots a^i_{|R_i|}]$. Let q be the query and define

$$E_{a,\mathbf{adom}} = \left(\bigcup_{i=1}^{\ell} \bigcup_{j=1}^{|R_i|} \delta_{a^i_j \to a}\big(\pi_{a^i_j}(R_i)\big) \right) \cup \big\{ \{ a \mapsto c \} \,\big|\, c \in \mathbf{adom}(q) \big\}.$$

2.

$$E_{\varphi_1 \vee \varphi_2} = E_{\neg(\neg\varphi_1 \wedge \neg\varphi_2)} \qquad\qquad E_{\forall y.\,\psi} = E_{\neg\exists y.\,\neg\psi}$$

3. Assume φ has free variables $x_1, \ldots, x_n$ with $\{ x_1, \ldots, x_n \} = \{ y_1, \ldots, y_\ell \} \cup \{ z_1, \ldots, z_k \}$, where $y_1, \ldots, y_\ell$ are the free variables of φ_1 and $z_1, \ldots, z_k$ are the free variables of φ_2.

$$\begin{aligned} E_{\varphi_1 \vee \varphi_2} &= E_{a_{x_1},\mathbf{adom}} \bowtie \cdots \bowtie E_{a_{x_n},\mathbf{adom}} - E_{\neg\varphi_1 \wedge \neg\varphi_2} \\ &= E_{a_{x_1},\mathbf{adom}} \bowtie \cdots \bowtie E_{a_{x_n},\mathbf{adom}} - (E_{\neg\varphi_1} \bowtie E_{\neg\varphi_2}) \end{aligned}$$

Exercise 6

Exercise. Complete the constructions for the proof of $AD \sqsubseteq RA$ given in the lecture.

1. Define the relational algebra expression $E_{a,\mathbf{adom}}$, such that $E_{a,\mathbf{adom}}(\mathcal{I}) = \{\{a \mapsto c\} \mid c \in \mathbf{adom}(\mathcal{I}, q)\}$ (assume that the query and the database schema are known).
2. Define the expressions E_φ for $\varphi = \varphi_1 \vee \varphi_2$ and $\varphi = \forall y.\psi$ in terms of expressions that have already been defined in the lecture.
3. Give a direct definition for the expression E_φ for $\varphi = \varphi_1 \vee \varphi_2 \equiv \neg(\neg\varphi_1 \wedge \neg\varphi_2)$.

Solution.

1. Assume that the database schema consists of tables $R_1, \ldots, R_\ell$ with table schemata $R_i[a^i_1, \ldots a^i_{|R_i|}]$. Let q be the query and define

$$E_{a,\mathbf{adom}} = \left(\bigcup_{i=1}^{\ell} \bigcup_{j=1}^{|R_i|} \delta_{a^i_j \to a}\left(\pi_{a^i_j}(R_i)\right) \right) \cup \left\{ \{ a \mapsto c \} \,\middle|\, c \in \mathbf{adom}(q) \right\}.$$

2.

$$E_{\varphi_1 \vee \varphi_2} = E_{\neg(\neg\varphi_1 \wedge \neg\varphi_2)} \qquad E_{\forall y.\,\psi} = E_{\neg\exists y.\,\neg\psi}$$

3. Assume φ has free variables $x_1, \ldots, x_n$ with $\{ x_1, \ldots, x_n \} = \{ y_1, \ldots, y_\ell \} \cup \{ z_1, \ldots, z_k \}$, where $y_1, \ldots, y_\ell$ are the free variables of φ_1 and $z_1, \ldots, z_k$ are the free variables of φ_2.

$$\begin{aligned}
E_{\varphi_1 \vee \varphi_2} &= E_{a_{x_1},\mathbf{adom}} \bowtie \cdots \bowtie E_{a_{x_n},\mathbf{adom}} - E_{\neg\varphi_1 \wedge \neg\varphi_2} \\
&= E_{a_{x_1},\mathbf{adom}} \bowtie \cdots \bowtie E_{a_{x_n},\mathbf{adom}} - (E_{\neg\varphi_1} \bowtie E_{\neg\varphi_2}) \\
&= E_{a_{x_1},\mathbf{adom}} \bowtie \cdots \bowtie E_{a_{x_n},\mathbf{adom}} - ((E_{a_{y_1},\mathbf{adom}} \bowtie \cdots \bowtie E_{a_{y_\ell},\mathbf{adom}} - E_{\varphi_1}) \bowtie (E_{a_{z_1},\mathbf{adom}} \bowtie \cdots \bowtie E_{a_{z_k},\mathbf{adom}} - E_{\varphi_2}))
\end{aligned}$$

Exercise 7

Exercise. Use the function rr from the lecture to compute the set of range-restricted variables for the following queries:

1. $\varphi_1 = \exists y_{\mathrm{SID}}, y_{\mathrm{Stop}}, y_{\mathrm{To}}.\, \big(\mathrm{Stops}(y_{\mathrm{SID}}, y_{\mathrm{Stop}}, \texttt{"true"}) \wedge \mathrm{Connect}(y_{\mathrm{SID}}, y_{\mathrm{To}}, x_{\mathrm{Line}})\big)[x_{\mathrm{Line}}]$
2. $\varphi_2 = \neg\mathrm{Lines}(x, \texttt{"bus"})[x]$
3. $\varphi_3 = \big(\mathrm{Connect}(x_1, \texttt{"42"}, \texttt{"85"}) \vee \mathrm{Connect}(\texttt{"57"}, x_2, \texttt{"85"})\big)[x_1, x_2]$
4. $\varphi_4 = \forall y.\, p(x, y)[x]$
5. $\varphi_5 = \exists x.\, (((p(x) \to q(c)) \to p(x)) \to p(x))$

Which of these queries is a safe-range query? Which of the queries is domain independent?

Exercise 7

Exercise. Use the function rr from the lecture to compute the set of range-restricted variables for the following queries:

1. $\varphi_1 = \exists y_{\text{SID}}, y_{\text{Stop}}, y_{\text{To}}. \big(\text{Stops}(y_{\text{SID}}, y_{\text{Stop}}, \texttt{"true"}) \land \text{Connect}(y_{\text{SID}}, y_{\text{To}}, x_{\text{Line}})\big)[x_{\text{Line}}]$
2. $\varphi_2 = \neg\text{Lines}(x, \texttt{"bus"})[x]$
3. $\varphi_3 = \big(\text{Connect}(x_1, \texttt{"42"}, \texttt{"85"}) \lor \text{Connect}(\texttt{"57"}, x_2, \texttt{"85"})\big)[x_1, x_2]$
4. $\varphi_4 = \forall y.\ p(x, y)[x]$
5. $\varphi_5 = \exists x.\ (((p(x) \to q(c)) \to p(x)) \to p(x))$

Which of these queries is a safe-range query? Which of the queries is domain independent?

Solution.

Exercise 7

Exercise. Use the function rr from the lecture to compute the set of range-restricted variables for the following queries:

1. $\varphi_1 = \exists y_{\text{SID}}, y_{\text{Stop}}, y_{\text{To}}. \big(\text{Stops}(y_{\text{SID}}, y_{\text{Stop}}, \texttt{"true"}) \land \text{Connect}(y_{\text{SID}}, y_{\text{To}}, x_{\text{Line}})\big)[x_{\text{Line}}]$
2. $\varphi_2 = \neg\text{Lines}(x, \texttt{"bus"})[x]$
3. $\varphi_3 = \big(\text{Connect}(x_1, \texttt{"42"}, \texttt{"85"}) \lor \text{Connect}(\texttt{"57"}, x_2, \texttt{"85"})\big)[x_1, x_2]$
4. $\varphi_4 = \forall y.\ p(x, y)[x]$
5. $\varphi_5 = \exists x.\ (((p(x) \to q(c)) \to p(x)) \to p(x))$

Which of these queries is a safe-range query? Which of the queries is domain independent?

Solution.

Definition (Lecture 2, Slide 26)

The set $\text{rr}(\varphi)$ of range-restricted variables of φ in Safe-Range Normal Form is defined recursively:

$$\text{rr}(R(t_1, \ldots, t_n)) = \{x \mid x \text{ is a variable among the } t_1, \ldots, t_n\}$$

$$\text{rr}(x \approx a) = \{x\}$$

$$\text{rr}(\varphi_1 \land \varphi_2) = \begin{cases} \text{rr}(\varphi_1) \cup \{x, y\} & \text{if } \varphi_2 = (x \approx y) \text{ and } \{x, y\} \cap \text{rr}(\varphi_1) \neq \emptyset \\ \text{rr}(\varphi_1) \cup \text{rr}(\varphi_2) & \text{otherwise} \end{cases}$$

$$\text{rr}(x \approx y) = \emptyset$$

$$\text{rr}(\exists y.\ \psi) = \begin{cases} \text{rr}(\psi) \setminus \{y\} & \text{if } y \in \text{rr}(\psi) \\ \textbf{throw new } \text{NotSafeException}() & \text{if } y \notin \text{rr}(\psi) \end{cases}$$

$$\text{rr}(\varphi_1 \lor \varphi_2) = \text{rr}(\varphi_1) \cap \text{rr}(\varphi_2)$$

$$\text{rr}(\neg\psi) = \emptyset \quad \text{if } \text{rr}(\psi) \text{ is defined}$$

Exercise 7

Exercise. Use the function rr from the lecture to compute the set of range-restricted variables for the following queries:

1. $\varphi_1 = \exists y_{\text{SID}}, y_{\text{Stop}}, y_{\text{To}}. \big(\text{Stops}(y_{\text{SID}}, y_{\text{Stop}}, \texttt{"true"}) \land \text{Connect}(y_{\text{SID}}, y_{\text{To}}, x_{\text{Line}})\big)[x_{\text{Line}}]$
2. $\varphi_2 = \neg\text{Lines}(x, \texttt{"bus"})[x]$
3. $\varphi_3 = \big(\text{Connect}(x_1, \texttt{"42"}, \texttt{"85"}) \lor \text{Connect}(\texttt{"57"}, x_2, \texttt{"85"})\big)[x_1, x_2]$
4. $\varphi_4 = \forall y.\ p(x, y)[x]$
5. $\varphi_5 = \exists x.\ (((p(x) \to q(c)) \to p(x)) \to p(x))$

Which of these queries is a safe-range query? Which of the queries is domain independent?

Solution.

$$\text{rr}(\varphi_1) = \{\, x_{\text{Line}} \,\}$$

Definition (Lecture 2, Slide 26)

The set $\text{rr}(\varphi)$ of range-restricted variables of φ in Safe-Range Normal Form is defined recursively:

$$\text{rr}(R(t_1, \ldots, t_n)) = \{x \mid x \text{ is a variable among the } t_1, \ldots, t_n\}$$

$$\text{rr}(\varphi_1 \land \varphi_2) = \begin{cases} \text{rr}(\varphi_1) \cup \{x, y\} & \text{if } \varphi_2 = (x \approx y) \text{ and } \{x, y\} \cap \text{rr}(\varphi_1) \neq \emptyset \\ \text{rr}(\varphi_1) \cup \text{rr}(\varphi_2) & \text{otherwise} \end{cases}$$

$$\text{rr}(\exists y.\ \psi) = \begin{cases} \text{rr}(\psi) \setminus \{y\} & \text{if } y \in \text{rr}(\psi) \\ \textbf{throw new } \text{NotSafeException}() & \text{if } y \notin \text{rr}(\psi) \end{cases}$$

$$\text{rr}(x \approx a) = \{x\}$$

$$\text{rr}(x \approx y) = \emptyset$$

$$\text{rr}(\varphi_1 \lor \varphi_2) = \text{rr}(\varphi_1) \cap \text{rr}(\varphi_2)$$

$$\text{rr}(\neg\psi) = \emptyset \quad \text{if } \text{rr}(\psi) \text{ is defined}$$

Exercise 7

Exercise. Use the function rr from the lecture to compute the set of range-restricted variables for the following queries:

1. $\varphi_1 = \exists y_{\text{SID}}, y_{\text{Stop}}, y_{\text{To}}. \big(\text{Stops}(y_{\text{SID}}, y_{\text{Stop}}, \texttt{"true"}) \wedge \text{Connect}(y_{\text{SID}}, y_{\text{To}}, x_{\text{Line}})\big)[x_{\text{Line}}]$
2. $\varphi_2 = \neg\text{Lines}(x, \texttt{"bus"})[x]$
3. $\varphi_3 = \big(\text{Connect}(x_1, \texttt{"42"}, \texttt{"85"}) \vee \text{Connect}(\texttt{"57"}, x_2, \texttt{"85"})\big)[x_1, x_2]$
4. $\varphi_4 = \forall y.\ p(x, y)[x]$
5. $\varphi_5 = \exists x.\ (((p(x) \to q(c)) \to p(x)) \to p(x))$

Which of these queries is a safe-range query? Which of the queries is domain independent?

Solution.

$$\text{rr}(\varphi_1) = \{\, x_{\text{Line}} \,\} \qquad \text{rr}(\varphi_2) = \emptyset$$

Definition (Lecture 2, Slide 26)

The set $\text{rr}(\varphi)$ of range-restricted variables of φ in Safe-Range Normal Form is defined recursively:

$$\text{rr}(R(t_1, \ldots, t_n)) = \{x \mid x \text{ is a variable among the } t_1, \ldots, t_n\}$$

$$\text{rr}(x \approx a) = \{x\}$$

$$\text{rr}(\varphi_1 \wedge \varphi_2) = \begin{cases} \text{rr}(\varphi_1) \cup \{x, y\} & \text{if } \varphi_2 = (x \approx y) \text{ and } \{x, y\} \cap \text{rr}(\varphi_1) \neq \emptyset \\ \text{rr}(\varphi_1) \cup \text{rr}(\varphi_2) & \text{otherwise} \end{cases}$$

$$\text{rr}(x \approx y) = \emptyset$$

$$\text{rr}(\exists y.\ \psi) = \begin{cases} \text{rr}(\psi) \setminus \{y\} & \text{if } y \in \text{rr}(\psi) \\ \textbf{throw new } \text{NotSafeException}() & \text{if } y \notin \text{rr}(\psi) \end{cases}$$

$$\text{rr}(\varphi_1 \vee \varphi_2) = \text{rr}(\varphi_1) \cap \text{rr}(\varphi_2)$$

$$\text{rr}(\neg\psi) = \emptyset \quad \text{if } \text{rr}(\psi) \text{ is defined}$$

Exercise 7

Exercise. Use the function rr from the lecture to compute the set of range-restricted variables for the following queries:

1. $\varphi_1 = \exists y_{\text{SID}}, y_{\text{Stop}}, y_{\text{To}}. \big(\text{Stops}(y_{\text{SID}}, y_{\text{Stop}}, \texttt{"true"}) \wedge \text{Connect}(y_{\text{SID}}, y_{\text{To}}, x_{\text{Line}})\big)[x_{\text{Line}}]$
2. $\varphi_2 = \neg\text{Lines}(x, \texttt{"bus"})[x]$
3. $\varphi_3 = \big(\text{Connect}(x_1, \texttt{"42"}, \texttt{"85"}) \vee \text{Connect}(\texttt{"57"}, x_2, \texttt{"85"})\big)[x_1, x_2]$
4. $\varphi_4 = \forall y.\ p(x, y)[x]$
5. $\varphi_5 = \exists x.\ (((p(x) \to q(c)) \to p(x)) \to p(x))$

Which of these queries is a safe-range query? Which of the queries is domain independent?

Solution.

$$\text{rr}(\varphi_1) = \{\, x_{\text{Line}} \,\} \qquad \text{rr}(\varphi_2) = \emptyset \qquad \text{rr}(\varphi_3) = \emptyset$$

Definition (Lecture 2, Slide 26)

The set $\text{rr}(\varphi)$ of range-restricted variables of φ in Safe-Range Normal Form is defined recursively:

$$\text{rr}(R(t_1, \ldots, t_n)) = \{x \mid x \text{ is a variable among the } t_1, \ldots, t_n\}$$

$$\text{rr}(x \approx a) = \{x\}$$

$$\text{rr}(\varphi_1 \wedge \varphi_2) = \begin{cases} \text{rr}(\varphi_1) \cup \{x, y\} & \text{if } \varphi_2 = (x \approx y) \text{ and } \{x, y\} \cap \text{rr}(\varphi_1) \neq \emptyset \\ \text{rr}(\varphi_1) \cup \text{rr}(\varphi_2) & \text{otherwise} \end{cases}$$

$$\text{rr}(x \approx y) = \emptyset$$

$$\text{rr}(\exists y.\ \psi) = \begin{cases} \text{rr}(\psi) \setminus \{y\} & \text{if } y \in \text{rr}(\psi) \\ \textbf{throw new } \text{NotSafeException}() & \text{if } y \notin \text{rr}(\psi) \end{cases}$$

$$\text{rr}(\varphi_1 \vee \varphi_2) = \text{rr}(\varphi_1) \cap \text{rr}(\varphi_2)$$

$$\text{rr}(\neg\psi) = \emptyset \quad \text{if } \text{rr}(\psi) \text{ is defined}$$

Exercise 7

Exercise. Use the function rr from the lecture to compute the set of range-restricted variables for the following queries:

1. $\varphi_1 = \exists y_{\text{SID}}, y_{\text{Stop}}, y_{\text{To}}. \big(\text{Stops}(y_{\text{SID}}, y_{\text{Stop}}, \texttt{"true"}) \wedge \text{Connect}(y_{\text{SID}}, y_{\text{To}}, x_{\text{Line}})\big)[x_{\text{Line}}]$
2. $\varphi_2 = \neg\text{Lines}(x, \texttt{"bus"})[x]$
3. $\varphi_3 = \big(\text{Connect}(x_1, \texttt{"42"}, \texttt{"85"}) \vee \text{Connect}(\texttt{"57"}, x_2, \texttt{"85"})\big)[x_1, x_2]$
4. $\varphi_4 = \forall y.\ p(x, y)[x] = \neg\exists y.\ \neg p(x, y)[x]$
5. $\varphi_5 = \exists x.\ (((p(x) \to q(c)) \to p(x)) \to p(x))$

Which of these queries is a safe-range query? Which of the queries is domain independent?

Solution.

$$\text{rr}(\varphi_1) = \{\, x_{\text{Line}} \,\} \qquad \text{rr}(\varphi_2) = \emptyset \qquad \text{rr}(\varphi_3) = \emptyset$$

Definition (Lecture 2, Slide 26)

The set $\text{rr}(\varphi)$ of range-restricted variables of φ in Safe-Range Normal Form is defined recursively:

$$\text{rr}(R(t_1, \ldots, t_n)) = \{x \mid x \text{ is a variable among the } t_1, \ldots, t_n\}$$

$$\text{rr}(\varphi_1 \wedge \varphi_2) = \begin{cases} \text{rr}(\varphi_1) \cup \{x, y\} & \text{if } \varphi_2 = (x \approx y) \text{ and } \{x, y\} \cap \text{rr}(\varphi_1) \neq \emptyset \\ \text{rr}(\varphi_1) \cup \text{rr}(\varphi_2) & \text{otherwise} \end{cases}$$

$$\text{rr}(\exists y.\ \psi) = \begin{cases} \text{rr}(\psi) \setminus \{y\} & \text{if } y \in \text{rr}(\psi) \\ \textbf{throw new } \text{NotSafeException}() & \text{if } y \notin \text{rr}(\psi) \end{cases}$$

$$\text{rr}(x \approx a) = \{x\}$$

$$\text{rr}(x \approx y) = \emptyset$$

$$\text{rr}(\varphi_1 \vee \varphi_2) = \text{rr}(\varphi_1) \cap \text{rr}(\varphi_2)$$

$$\text{rr}(\neg\psi) = \emptyset \quad \text{if } \text{rr}(\psi) \text{ is defined}$$

Exercise 7

Exercise. Use the function rr from the lecture to compute the set of range-restricted variables for the following queries:

1. $\varphi_1 = \exists y_{\text{SID}}, y_{\text{Stop}}, y_{\text{To}}. \big(\text{Stops}(y_{\text{SID}}, y_{\text{Stop}}, \texttt{"true"}) \wedge \text{Connect}(y_{\text{SID}}, y_{\text{To}}, x_{\text{Line}})\big)[x_{\text{Line}}]$
2. $\varphi_2 = \neg\text{Lines}(x, \texttt{"bus"})[x]$
3. $\varphi_3 = \big(\text{Connect}(x_1, \texttt{"42"}, \texttt{"85"}) \vee \text{Connect}(\texttt{"57"}, x_2, \texttt{"85"})\big)[x_1, x_2]$
4. $\varphi_4 = \forall y.\ p(x, y)[x] = \neg\exists y.\ \neg p(x, y)[x]$
5. $\varphi_5 = \exists x.\ (((p(x) \rightarrow q(c)) \rightarrow p(x)) \rightarrow p(x))$

Which of these queries is a safe-range query? Which of the queries is domain independent?

Solution.

$\text{rr}(\varphi_1) = \{\, x_{\text{Line}} \,\}$ $\quad$ $\text{rr}(\varphi_2) = \emptyset$ $\quad$ $\text{rr}(\varphi_3) = \emptyset$ $\quad$ $\text{rr}(SNRF(\varphi_4)) =$ **Exception**

Definition (Lecture 2, Slide 26)

The set $\text{rr}(\varphi)$ of range-restricted variables of φ in Safe-Range Normal Form is defined recursively:

$$\text{rr}(R(t_1, \ldots, t_n)) = \{x \mid x \text{ is a variable among the } t_1, \ldots, t_n\}$$

$$\text{rr}(x \approx a) = \{x\}$$

$$\text{rr}(\varphi_1 \wedge \varphi_2) = \begin{cases} \text{rr}(\varphi_1) \cup \{x, y\} & \text{if } \varphi_2 = (x \approx y) \text{ and } \{x, y\} \cap \text{rr}(\varphi_1) \neq \emptyset \\ \text{rr}(\varphi_1) \cup \text{rr}(\varphi_2) & \text{otherwise} \end{cases}$$

$$\text{rr}(x \approx y) = \emptyset$$

$$\text{rr}(\exists y.\ \psi) = \begin{cases} \text{rr}(\psi) \setminus \{y\} & \text{if } y \in \text{rr}(\psi) \\ \textbf{throw new } \text{NotSafeException()} & \text{if } y \notin \text{rr}(\psi) \end{cases}$$

$$\text{rr}(\varphi_1 \vee \varphi_2) = \text{rr}(\varphi_1) \cap \text{rr}(\varphi_2)$$

$$\text{rr}(\neg\psi) = \emptyset \quad \text{if } \text{rr}(\psi) \text{ is defined}$$

Exercise 7

Exercise. Use the function rr from the lecture to compute the set of range-restricted variables for the following queries:

1. $\varphi_1 = \exists y_{\text{SID}}, y_{\text{Stop}}, y_{\text{To}}. \big(\text{Stops}(y_{\text{SID}}, y_{\text{Stop}}, \texttt{"true"}) \land \text{Connect}(y_{\text{SID}}, y_{\text{To}}, x_{\text{Line}})\big)[x_{\text{Line}}]$
2. $\varphi_2 = \neg\text{Lines}(x, \texttt{"bus"})[x]$
3. $\varphi_3 = \big(\text{Connect}(x_1, \texttt{"42"}, \texttt{"85"}) \lor \text{Connect}(\texttt{"57"}, x_2, \texttt{"85"})\big)[x_1, x_2]$
4. $\varphi_4 = \forall y.\, p(x, y)[x] = \neg\exists y.\, \neg p(x, y)[x]$
5. $\varphi_5 = \exists x.\, (((p(x) \to q(c)) \to p(x)) \to p(x)) = \exists x.\, (((\neg p(x) \lor q(c)) \land \neg p(x)) \lor p(x))$

Which of these queries is a safe-range query? Which of the queries is domain independent?

Solution.

$\text{rr}(\varphi_1) = \{\, x_{\text{Line}} \,\}$ $\qquad$ $\text{rr}(\varphi_2) = \emptyset$ $\qquad$ $\text{rr}(\varphi_3) = \emptyset$ $\qquad$ $\text{rr}(SNRF(\varphi_4)) =$ **Exception**

Definition (Lecture 2, Slide 26)

The set $\text{rr}(\varphi)$ of range-restricted variables of φ in Safe-Range Normal Form is defined recursively:

$$\text{rr}(R(t_1, \ldots, t_n)) = \{x \mid x \text{ is a variable among the } t_1, \ldots, t_n\}$$

$$\text{rr}(\varphi_1 \land \varphi_2) = \begin{cases} \text{rr}(\varphi_1) \cup \{x, y\} & \text{if } \varphi_2 = (x \approx y) \text{ and } \{x, y\} \cap \text{rr}(\varphi_1) \neq \emptyset \\ \text{rr}(\varphi_1) \cup \text{rr}(\varphi_2) & \text{otherwise} \end{cases}$$

$$\text{rr}(\exists y.\, \psi) = \begin{cases} \text{rr}(\psi) \setminus \{y\} & \text{if } y \in \text{rr}(\psi) \\ \textbf{throw new } \text{NotSafeException()} & \text{if } y \notin \text{rr}(\psi) \end{cases}$$

$$\text{rr}(x \approx a) = \{x\}$$

$$\text{rr}(x \approx y) = \emptyset$$

$$\text{rr}(\varphi_1 \lor \varphi_2) = \text{rr}(\varphi_1) \cap \text{rr}(\varphi_2)$$

$$\text{rr}(\neg\psi) = \emptyset \quad \text{if } \text{rr}(\psi) \text{ is defined}$$

Exercise 7

Exercise. Use the function rr from the lecture to compute the set of range-restricted variables for the following queries:

1. $\varphi_1 = \exists y_{\text{SID}}, y_{\text{Stop}}, y_{\text{To}}. \big(\text{Stops}(y_{\text{SID}}, y_{\text{Stop}}, \texttt{"true"}) \wedge \text{Connect}(y_{\text{SID}}, y_{\text{To}}, x_{\text{Line}})\big)[x_{\text{Line}}]$
2. $\varphi_2 = \neg\text{Lines}(x, \texttt{"bus"})[x]$
3. $\varphi_3 = \big(\text{Connect}(x_1, \texttt{"42"}, \texttt{"85"}) \vee \text{Connect}(\texttt{"57"}, x_2, \texttt{"85"})\big)[x_1, x_2]$
4. $\varphi_4 = \forall y.\, p(x, y)[x] = \neg\exists y.\, \neg p(x, y)[x]$
5. $\varphi_5 = \exists x.\, (((p(x) \rightarrow q(c)) \rightarrow p(x)) \rightarrow p(x)) = \exists x.\, (((\neg p(x) \vee q(c)) \wedge \neg p(x)) \vee p(x))$

Which of these queries is a safe-range query? Which of the queries is domain independent?

Solution.

$\text{rr}(\varphi_1) = \{\, x_{\text{Line}} \,\}$ $\quad$ $\text{rr}(\varphi_2) = \emptyset$ $\quad$ $\text{rr}(\varphi_3) = \emptyset$ $\quad$ $\text{rr}(SNRF(\varphi_4)) = \textbf{Exception}$ $\quad$ $\text{rr}(SNRF(\varphi_5)) = \textbf{Exception}$

Definition (Lecture 2, Slide 26)

The set $\text{rr}(\varphi)$ of range-restricted variables of φ in Safe-Range Normal Form is defined recursively:

$$\text{rr}(R(t_1, \ldots, t_n)) = \{x \mid x \text{ is a variable among the } t_1, \ldots, t_n\}$$

$$\text{rr}(\varphi_1 \wedge \varphi_2) = \begin{cases} \text{rr}(\varphi_1) \cup \{x, y\} & \text{if } \varphi_2 = (x \approx y) \text{ and } \{x, y\} \cap \text{rr}(\varphi_1) \neq \emptyset \\ \text{rr}(\varphi_1) \cup \text{rr}(\varphi_2) & \text{otherwise} \end{cases}$$

$$\text{rr}(\exists y.\, \psi) = \begin{cases} \text{rr}(\psi) \setminus \{y\} & \text{if } y \in \text{rr}(\psi) \\ \textbf{throw new } \text{NotSafeException}() & \text{if } y \notin \text{rr}(\psi) \end{cases}$$

$$\text{rr}(x \approx a) = \{x\}$$

$$\text{rr}(x \approx y) = \emptyset$$

$$\text{rr}(\varphi_1 \vee \varphi_2) = \text{rr}(\varphi_1) \cap \text{rr}(\varphi_2)$$

$$\text{rr}(\neg\psi) = \emptyset \quad \text{if } \text{rr}(\psi) \text{ is defined}$$

Exercise 7

Exercise. Use the function rr from the lecture to compute the set of range-restricted variables for the following queries:

1. $\varphi_1 = \exists y_{\mathrm{SID}}, y_{\mathrm{Stop}}, y_{\mathrm{To}}.\ \big(\mathrm{Stops}(y_{\mathrm{SID}}, y_{\mathrm{Stop}}, \texttt{"true"}) \wedge \mathrm{Connect}(y_{\mathrm{SID}}, y_{\mathrm{To}}, x_{\mathrm{Line}})\big)[x_{\mathrm{Line}}]$
2. $\varphi_2 = \neg\mathrm{Lines}(x, \texttt{"bus"})[x]$
3. $\varphi_3 = \big(\mathrm{Connect}(x_1, \texttt{"42"}, \texttt{"85"}) \vee \mathrm{Connect}(\texttt{"57"}, x_2, \texttt{"85"})\big)[x_1, x_2]$
4. $\varphi_4 = \forall y.\ p(x, y)[x] = \neg\exists y.\ \neg p(x, y)[x]$
5. $\varphi_5 = \exists x.\ (((p(x) \to q(c)) \to p(x)) \to p(x)) = \exists x.\ (((\neg p(x) \vee q(c)) \wedge \neg p(x)) \vee p(x))$

Which of these queries is a safe-range query? Which of the queries is domain independent?

Solution.

$\mathrm{rr}(\varphi_1) = \{\, x_{\mathrm{Line}} \,\}$ $\quad \mathrm{rr}(\varphi_2) = \emptyset$ $\quad \mathrm{rr}(\varphi_3) = \emptyset$ $\quad \mathrm{rr}(\mathit{SNRF}(\varphi_4)) =$ **Exception** $\quad \mathrm{rr}(\mathit{SNRF}(\varphi_5)) =$ **Exception**

Definition (Lecture 2, Slide 27)

An FO query $q = \varphi[x_1, \ldots, x_n]$ is a safe-range query if $\mathrm{rr}(\mathrm{SRNF}(\varphi)) = \{\, x_1, \ldots, x_n \,\}$.

Exercise 7

Exercise. Use the function rr from the lecture to compute the set of range-restricted variables for the following queries:

1. $\varphi_1 = \exists y_{\mathrm{SID}}, y_{\mathrm{Stop}}, y_{\mathrm{To}}. \big(\mathrm{Stops}(y_{\mathrm{SID}}, y_{\mathrm{Stop}}, \texttt{"true"}) \wedge \mathrm{Connect}(y_{\mathrm{SID}}, y_{\mathrm{To}}, x_{\mathrm{Line}})\big)[x_{\mathrm{Line}}]$
2. $\varphi_2 = \neg\mathrm{Lines}(x, \texttt{"bus"})[x]$
3. $\varphi_3 = \big(\mathrm{Connect}(x_1, \texttt{"42"}, \texttt{"85"}) \vee \mathrm{Connect}(\texttt{"57"}, x_2, \texttt{"85"})\big)[x_1, x_2]$
4. $\varphi_4 = \forall y.\, p(x, y)[x] = \neg\exists y.\, \neg p(x, y)[x]$
5. $\varphi_5 = \exists x.\, (((p(x) \to q(c)) \to p(x)) \to p(x)) = \exists x.\, (((\neg p(x) \vee q(c)) \wedge \neg p(x)) \vee p(x))$

Which of these queries is a safe-range query? Which of the queries is domain independent?

Solution.

$\mathrm{rr}(\varphi_1) = \{\, x_{\mathrm{Line}} \,\}$ SR, DI

$\mathrm{rr}(\varphi_2) = \emptyset$

$\mathrm{rr}(\varphi_3) = \emptyset$

$\mathrm{rr}(\mathit{SNRF}(\varphi_4)) =$ **Exception**

$\mathrm{rr}(\mathit{SNRF}(\varphi_5)) =$ **Exception**

Definition (Lecture 2, Slide 27)

An FO query $q = \varphi[x_1, \ldots, x_n]$ is a safe-range query if $\mathrm{rr}(\mathrm{SRNF}(\varphi)) = \{\, x_1, \ldots, x_n \,\}$.

Exercise 7

Exercise. Use the function rr from the lecture to compute the set of range-restricted variables for the following queries:

1. $\varphi_1 = \exists y_{\mathrm{SID}}, y_{\mathrm{Stop}}, y_{\mathrm{To}}. \big(\mathrm{Stops}(y_{\mathrm{SID}}, y_{\mathrm{Stop}}, \texttt{"true"}) \land \mathrm{Connect}(y_{\mathrm{SID}}, y_{\mathrm{To}}, x_{\mathrm{Line}})\big)[x_{\mathrm{Line}}]$
2. $\varphi_2 = \neg\mathrm{Lines}(x, \texttt{"bus"})[x]$
3. $\varphi_3 = \big(\mathrm{Connect}(x_1, \texttt{"42"}, \texttt{"85"}) \lor \mathrm{Connect}(\texttt{"57"}, x_2, \texttt{"85"})\big)[x_1, x_2]$
4. $\varphi_4 = \forall y.\, p(x, y)[x] = \neg\exists y.\, \neg p(x, y)[x]$
5. $\varphi_5 = \exists x.\, (((p(x) \to q(c)) \to p(x)) \to p(x)) = \exists x.\, (((\neg p(x) \lor q(c)) \land \neg p(x)) \lor p(x))$

Which of these queries is a safe-range query? Which of the queries is domain independent?

Solution.

$\mathrm{rr}(\varphi_1) = \{ x_{\mathrm{Line}} \}$	$\mathrm{rr}(\varphi_2) = \emptyset$	$\mathrm{rr}(\varphi_3) = \emptyset$	$\mathrm{rr}(SNRF(\varphi_4)) =$ **Exception**	$\mathrm{rr}(SNRF(\varphi_5)) =$ **Exception**
SR, DI	not SR, not DI			

Definition (Lecture 2, Slide 27)

An FO query $q = \varphi[x_1, \ldots, x_n]$ is a safe-range query if $\mathrm{rr}(\mathrm{SRNF}(\varphi)) = \{ x_1, \ldots, x_n \}$.

Exercise 7

Exercise. Use the function rr from the lecture to compute the set of range-restricted variables for the following queries:

1. $\varphi_1 = \exists y_{\text{SID}}, y_{\text{Stop}}, y_{\text{To}}. \big(\text{Stops}(y_{\text{SID}}, y_{\text{Stop}}, \texttt{"true"}) \land \text{Connect}(y_{\text{SID}}, y_{\text{To}}, x_{\text{Line}})\big)[x_{\text{Line}}]$
2. $\varphi_2 = \neg\text{Lines}(x, \texttt{"bus"})[x]$
3. $\varphi_3 = \big(\text{Connect}(x_1, \texttt{"42"}, \texttt{"85"}) \lor \text{Connect}(\texttt{"57"}, x_2, \texttt{"85"})\big)[x_1, x_2]$
4. $\varphi_4 = \forall y.\ p(x, y)[x] = \neg\exists y.\ \neg p(x, y)[x]$
5. $\varphi_5 = \exists x.\ (((p(x) \to q(c)) \to p(x)) \to p(x)) = \exists x.\ (((\neg p(x) \lor q(c)) \land \neg p(x)) \lor p(x))$

Which of these queries is a safe-range query? Which of the queries is domain independent?

Solution.

$\text{rr}(\varphi_1) = \{ x_{\text{Line}} \}$	$\text{rr}(\varphi_2) = \emptyset$	$\text{rr}(\varphi_3) = \emptyset$	$\text{rr}(\mathit{SNRF}(\varphi_4)) =$ **Exception**	$\text{rr}(\mathit{SNRF}(\varphi_5)) =$ **Exception**
SR, DI	not SR, not DI	not SR, not DI		

Definition (Lecture 2, Slide 27)

An FO query $q = \varphi[x_1, \ldots, x_n]$ is a safe-range query if $\text{rr}(\text{SRNF}(\varphi)) = \{ x_1, \ldots, x_n \}$.

Exercise 7

Exercise. Use the function rr from the lecture to compute the set of range-restricted variables for the following queries:

1. $\varphi_1 = \exists y_{\text{SID}}, y_{\text{Stop}}, y_{\text{To}}. \big(\text{Stops}(y_{\text{SID}}, y_{\text{Stop}}, \texttt{"true"}) \land \text{Connect}(y_{\text{SID}}, y_{\text{To}}, x_{\text{Line}})\big)[x_{\text{Line}}]$
2. $\varphi_2 = \neg\text{Lines}(x, \texttt{"bus"})[x]$
3. $\varphi_3 = \big(\text{Connect}(x_1, \texttt{"42"}, \texttt{"85"}) \lor \text{Connect}(\texttt{"57"}, x_2, \texttt{"85"})\big)[x_1, x_2]$
4. $\varphi_4 = \forall y.\ p(x, y)[x] = \neg\exists y.\ \neg p(x, y)[x]$
5. $\varphi_5 = \exists x.\ (((p(x) \rightarrow q(c)) \rightarrow p(x)) \rightarrow p(x)) = \exists x.\ (((\neg p(x) \lor q(c)) \land \neg p(x)) \lor p(x))$

Which of these queries is a safe-range query? Which of the queries is domain independent?

Solution.

$\text{rr}(\varphi_1) = \{ x_{\text{Line}} \}$	$\text{rr}(\varphi_2) = \emptyset$	$\text{rr}(\varphi_3) = \emptyset$	$\text{rr}(SNRF(\varphi_4)) =$ **Exception**	$\text{rr}(SNRF(\varphi_5)) =$ **Exception**
SR, DI	not SR, not DI	not SR, not DI	not SR, not DI	

Definition (Lecture 2, Slide 27)

An FO query $q = \varphi[x_1, \ldots, x_n]$ is a safe-range query if $\text{rr}(\text{SRNF}(\varphi)) = \{ x_1, \ldots, x_n \}$.

Exercise 7

Exercise. Use the function rr from the lecture to compute the set of range-restricted variables for the following queries:

1. $\varphi_1 = \exists y_{\text{SID}}, y_{\text{Stop}}, y_{\text{To}}. \big(\text{Stops}(y_{\text{SID}}, y_{\text{Stop}}, \texttt{"true"}) \wedge \text{Connect}(y_{\text{SID}}, y_{\text{To}}, x_{\text{Line}})\big)[x_{\text{Line}}]$
2. $\varphi_2 = \neg\text{Lines}(x, \texttt{"bus"})[x]$
3. $\varphi_3 = \big(\text{Connect}(x_1, \texttt{"42"}, \texttt{"85"}) \vee \text{Connect}(\texttt{"57"}, x_2, \texttt{"85"})\big)[x_1, x_2]$
4. $\varphi_4 = \forall y.\ p(x, y)[x] = \neg\exists y.\ \neg p(x, y)[x]$
5. $\varphi_5 = \exists x.\ (((p(x) \rightarrow q(c)) \rightarrow p(x)) \rightarrow p(x)) = \exists x.\ (((\neg p(x) \vee q(c)) \wedge \neg p(x)) \vee p(x))$

Which of these queries is a safe-range query? Which of the queries is domain independent?

Solution.

$\text{rr}(\varphi_1) = \{ x_{\text{Line}} \}$	$\text{rr}(\varphi_2) = \emptyset$	$\text{rr}(\varphi_3) = \emptyset$	$\text{rr}(SNRF(\varphi_4)) =$ **Exception**	$\text{rr}(SNRF(\varphi_5)) =$ **Exception**
SR, DI	not SR, not DI	not SR, not DI	not SR, not DI	not SR, DI

Definition (Lecture 2, Slide 27)

An FO query $q = \varphi[x_1, \dots, x_n]$ is a safe-range query if $\text{rr}(\text{SRNF}(\varphi)) = \{ x_1, \dots, x_n \}$.