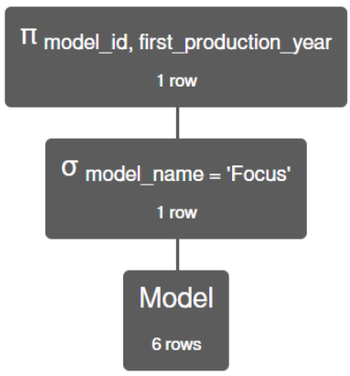
Project (∏)

Relational Algebra

π model_id, first_production_year (σ model_name = 'Focus' (Model))

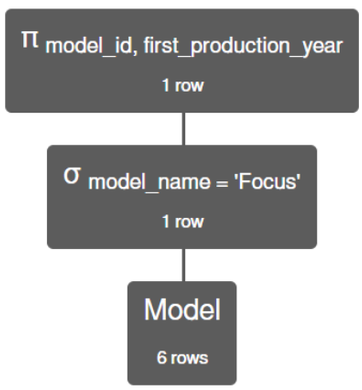


$$\pi_{\text{model\_id, first\_production\_year}} \left(\sigma_{\text{model\_name = 'Focus'}} \left(\text{Model} \right) \right)$$

Model.model_id	Model.first_production_year
4	'1986'

SQL Query

SELECT model_id, first_production_year FROM Model WHERE model_name='Focus';



$$\pi_{\text{model\_id, first\_production\_year}} \sigma_{\text{model\_name = 'Focus'}} \text{Model}$$

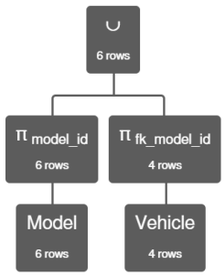
Model.model_id	Model.first_production_year
4	'1986'

Explanation: The Project operator will select columns to show, in this case the query will display the columns model_id and first_production_year in the selected table where select criteria is met.

Union (∪)

Relational Algebra

π model_id (Model) ∪ π fk_model_id (Vehicle)

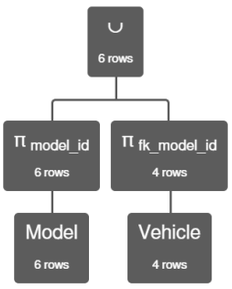


$$\pi_{model\_id}(Model) \cup \pi_{fk\_model\_id}(Vehicle)$$

Model.model_id
1
2
3
4
5
6

SQL Query

SELECT model_id FROM Model UNION SELECT fk_model_id FROM Vehicle;



$$\pi_{model\_id}\ Model \cup \pi_{fk\_model\_id}\ Vehicle$$

Model.model_id
1
2
3
4
5
6

Explanation: The Union operator will combine tables as long as the data types and columns are the same. In the example for tables Model and Vehicle both have a model_id that and a number data type.

Cartesian Product (X)

Relational Algebra

Make × Color

Make × Color

Make.make_id	Make.make_name	Make.country	Color.color_id	Color.name	Color.code
1	'Make1'	'Italy'	1	'Sky Blue'	'#1869c5'
1	'Make1'	'Italy'	2	'Bold Red'	'#b00007'
1	'Make1'	'Italy'	3	'Electic Green'	'#48D597'
1	'Make1'	'Italy'	4	'Smoke Grey'	'#333F48'
2	'Make2'	'Japan'	1	'Sky Blue'	'#1869c5'
2	'Make2'	'Japan'	2	'Bold Red'	'#b00007'
2	'Make2'	'Japan'	3	'Electic Green'	'#48D597'
2	'Make2'	'Japan'	4	'Smoke Grey'	'#333F48'
3	'Make3'	'USA'	1	'Sky Blue'	'#1869c5'
3	'Make3'	'USA'	2	'Bold Red'	'#b00007'

SQL Query

SELECT * FROM Make, Color;

Make × Color

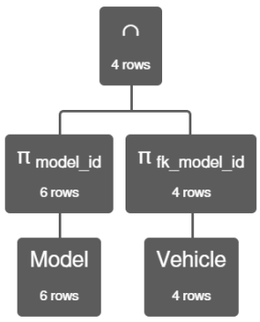
Make.make_id	Make.make_name	Make.country	Color.color_id	Color.name	Color.code
1	'Make1'	'Italy'	1	'Sky Blue'	'#1869c5'
1	'Make1'	'Italy'	2	'Bold Red'	'#b00007'
1	'Make1'	'Italy'	3	'Electic Green'	'#48D597'
1	'Make1'	'Italy'	4	'Smoke Grey'	'#333F48'
2	'Make2'	'Japan'	1	'Sky Blue'	'#1869c5'
2	'Make2'	'Japan'	2	'Bold Red'	'#b00007'
2	'Make2'	'Japan'	3	'Electic Green'	'#48D597'
2	'Make2'	'Japan'	4	'Smoke Grey'	'#333F48'
3	'Make3'	'USA'	1	'Sky Blue'	'#1869c5'
3	'Make3'	'USA'	2	'Bold Red'	'#b00007'

Explanation: The Cartesian Product operator will join columns to show the results of both tables by adding all columns from each, in this case the query will display all columns from Make and Color.

Intersect (∩)

Relational Algebra

π model_id Model ∩ π fk_model_id Vehicle

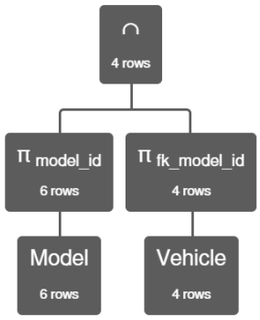


π model_id Model ∩ π fk_model_id Vehicle

Model.model_id
1
2
4
5

SQL Query

SELECT model_id FROM Model INTERSECT SELECT fk_model_id FROM Vehicle;



π model_id Model ∩ π fk_model_id Vehicle

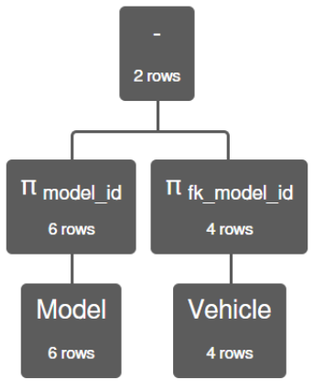
Model.model_id
1
2
4
5

Explanation: The Intersect operator combine columns but return only those from that match. In this case it will return all model_id that are in both Model and Vehicle (1, 2, 4, and 5).

Difference (-)

Relational Algebra

π model_id Model - π fk_model_id Vehicle

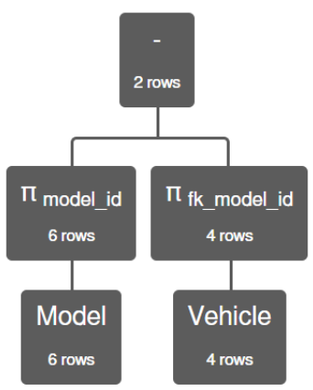


π_{model_id} Model - $\pi_{fk\_model\_id}$ Vehicle

Model.model_id
3
6

SQL Query

SELECT model_id FROM Model EXCEPT SELECT fk_model_id FROM Vehicle



π_{model_id} Model - $\pi_{fk\_model\_id}$ Vehicle

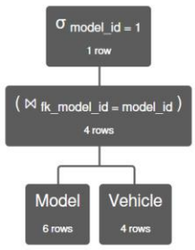
Model.model_id
3
6

Explanation: The Difference operator will return all that are in the first relation but are not on the second one. In this case it will return all model_id that are in Model but not in Vehicle (3 and 6).

Join (⋈, ⋈, ⟕, or ⟖)

Relational Algebra

π model_id Model - π fk_model_id Vehicle

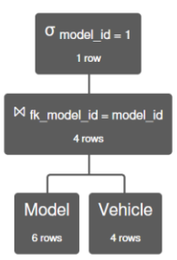


σ model_id = 1 (Model ⋈ fk_model_id = model_id Vehicle)

Model.model_id	Model.model_name	Model.first_production_year	Vehicle.vehicle_id	Vehicle.fk_make_id	Vehicle.fk_model_id	Vehicle.year
1	'Model1'	'1985'	1	1	1	'1985'

SQL Query

SELECT * FROM Model INNER JOIN Vehicle ON fk_model_id = model_id WHERE model_id = 1;



σ model_id = 1 (Model ⋈ fk_model_id = model_id Vehicle)

Model.model_id	Model.model_name	Model.first_production_year	Vehicle.vehicle_id	Vehicle.fk_make_id	Vehicle.fk_model_id	Vehicle.year
1	'Model1'	'1985'	1	1	1	'1985'

Explanation: The Join operator shows all column from two tables that have a common relation, in this case model_id and fk_model_id where model_id is equal to 1 meaning that fk_model_id has to be 1.