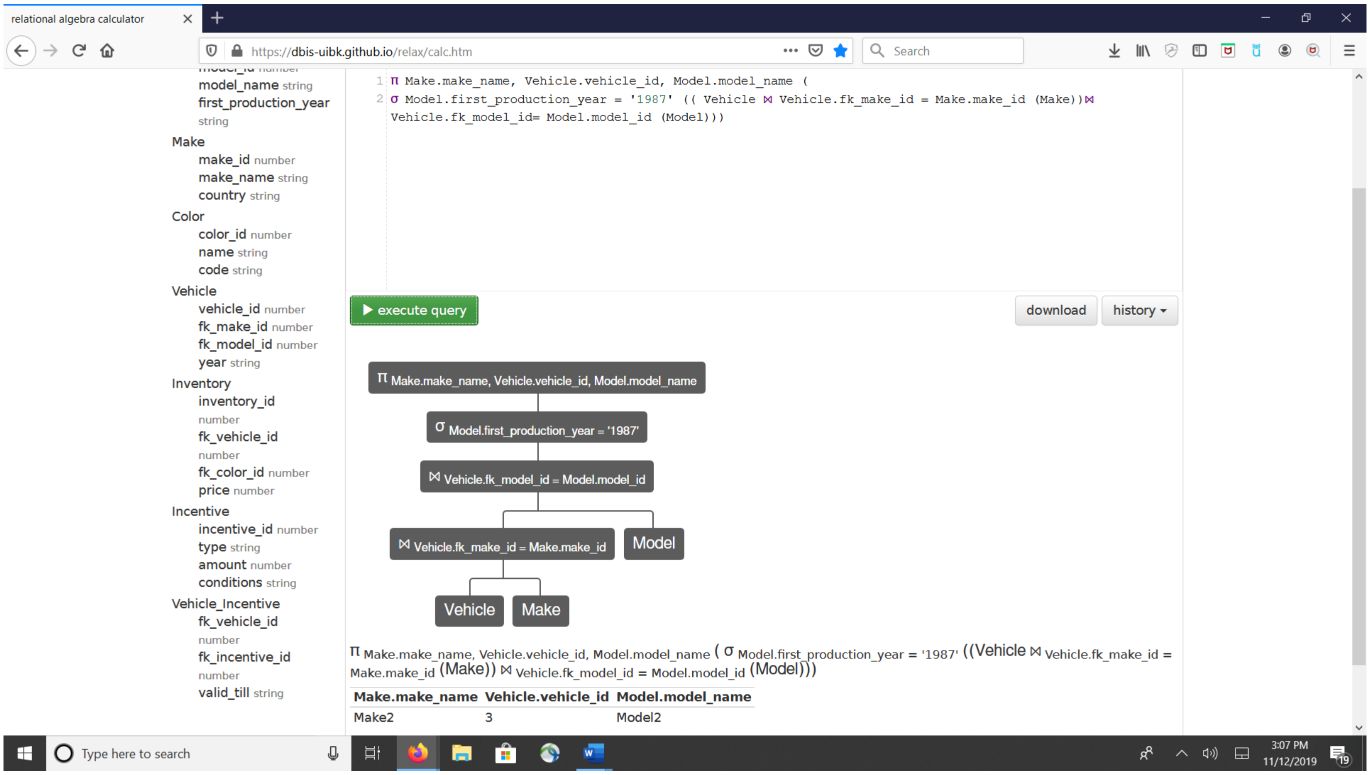
1. Select the make_name, vehicle_id, and model_name of all vehicles, regardless of whether they are on the lot or not, which have a first production year of 1987.

π Make.make_name, Vehicle.vehicle_id, Model.model_name(σ Model.first_production_year = 1987((Vehicle ⋈ Vehicle.fk_make_id = Make.make_id (Make))⋈ Vehicle.fk_model_id= Model.model_id (Model)))


relational algebra calculator
https://dbis-uibk.github.io/relax/calc.htm
Search
execute query
download
history
Π Make.make_name, Vehicle.vehicle_id, Model.model_name
σ Model.first_production_year = '1987'
⋈ Vehicle.fk_model_id = Model.model_id
⋈ Vehicle.fk_make_id = Make.make_id
Model
Vehicle
Make
Type here to search
3:07 PM
11/12/2019

model_name string
first_production_year string
Make
make_id number
make_name string
country string
Color
color_id number
name string
code string
Vehicle
vehicle_id number
fk_make_id number
fk_model_id number
year string
Inventory
inventory_id number
fk_vehicle_id number
fk_color_id number
price number
Incentive
incentive_id number
type string
amount number
conditions string
Vehicle_Incentive
fk_vehicle_id number
fk_incentive_id number
valid_till string

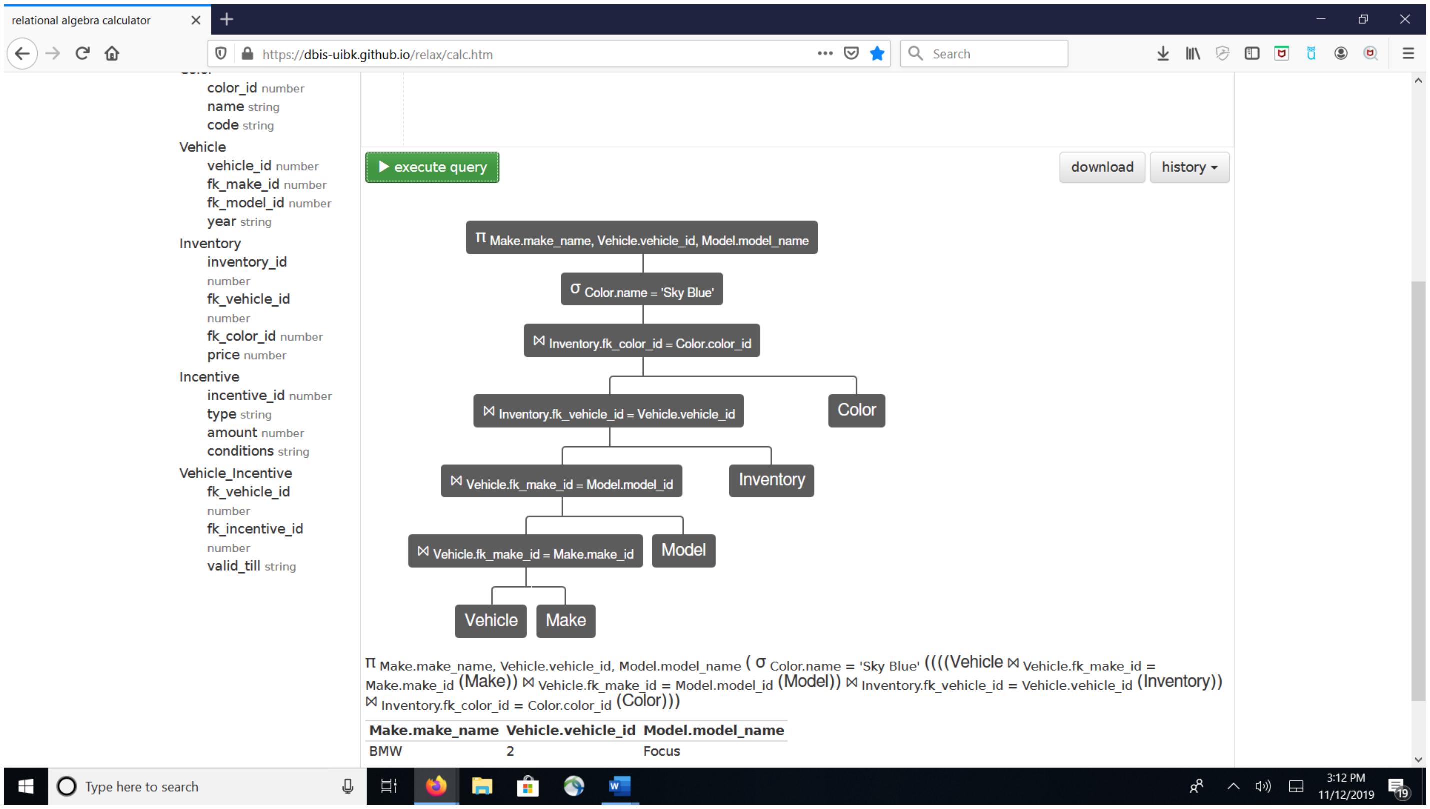
```
π Make.make_name, Vehicle.vehicle_id, Model.model_name (
σ Model.first_production_year = '1987' (( Vehicle ⋈ Vehicle.fk_make_id = Make.make_id (Make))⋈
Vehicle.fk_model_id= Model.model_id (Model)))
```

π Make.make_name, Vehicle.vehicle_id, Model.model_name (σ Model.first_production_year = '1987' ((Vehicle ⋈ Vehicle.fk_make_id = Make.make_id (Make)) ⋈ Vehicle.fk_model_id = Model.model_id (Model)))

Make.make_name	Vehicle.vehicle_id	Model.model_name
Make2	3	Model2

2. Select the make_name, vehicle_id, and model_name of all vehicles with the color name “Sky Blue” and which are on the lot.

π Make.make_name, Vehicle.vehicle_id, Model.model_name (σ Color.name="Sky Blue" ((((Vehicle ⋈ Vehicle.fk_make_id = Make.make_id(Make))⋈ Vehicle.fk_make_id = Model.model_id(Model))⋈ Inventory.fk_vehicle_id = Vehicle.vehicle_id (Inventory))⋈ Inventory.fk_color_id= Color.color_id(Color)))


relational algebra calculator
https://dbis-uibk.github.io/relax/calc.htm
Search
execute query
download
history
Type here to search
3:12 PM
11/12/2019

color_id number
name string
code string

Vehicle
vehicle_id number
fk_make_id number
fk_model_id number
year string

Inventory
inventory_id number
fk_vehicle_id number
fk_color_id number
price number

Incentive
incentive_id number
type string
amount number
conditions string

Vehicle_Incentive
fk_vehicle_id number
fk_incentive_id number
valid_till string

π Make.make_name, Vehicle.vehicle_id, Model.model_name
σ Color.name = 'Sky Blue'
⋈ Inventory.fk_color_id = Color.color_id
⋈ Inventory.fk_vehicle_id = Vehicle.vehicle_id
Color
⋈ Vehicle.fk_make_id = Model.model_id
Inventory
⋈ Vehicle.fk_make_id = Make.make_id
Model
Vehicle
Make

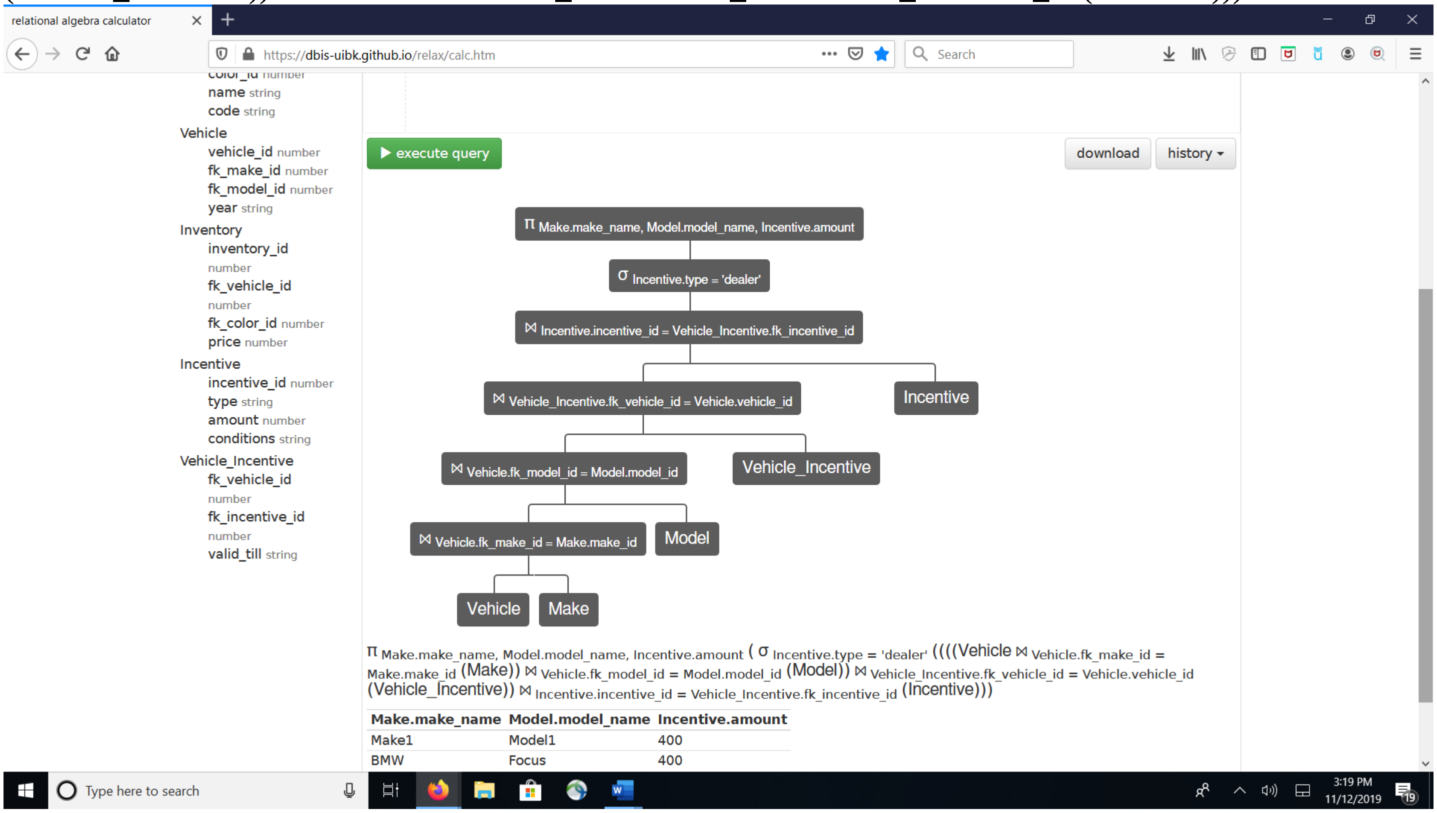
$\pi_{\text{Make.make\_name, Vehicle.vehicle\_id, Model.model\_name}} (\sigma_{\text{Color.name = 'Sky Blue'}} ((((\text{Vehicle} \bowtie_{\text{Vehicle.fk\_make\_id = Make.make\_id}} (\text{Make})) \bowtie_{\text{Vehicle.fk\_make\_id = Model.model\_id}} (\text{Model})) \bowtie_{\text{Inventory.fk\_vehicle\_id = Vehicle.vehicle\_id}} (\text{Inventory})) \bowtie_{\text{Inventory.fk\_color\_id = Color.color\_id}} (\text{Color})))$

Make.make_name	Vehicle.vehicle_id	Model.model_name
BMW	2	Focus

3. Select the make_name, model_name and incentive amount for all vehicles on the lot with an incentive type “dealer”.

π Make.make_name, Model.model_name, Incentive.amount (σ Incentive.type = 'dealer' ((((Vehicle ⋈ Vehicle.fk_make_id = Make.make_id (Make)) ⋈ Vehicle.fk_model_id = Model.model_id (Model)) ⋈ Vehicle_Incentive.fk_vehicle_id = Vehicle.vehicle_id

(Vehicle_Incentive))⋈ Incentive.incentive_id = Vehicle_Incentive.fk_incentive_id (Incentive)))


relational algebra calculator
https://dbis-uibk.github.io/relax/calc.htm
Search
execute query
download
history
π Make.make_name, Model.model_name, Incentive.amount
σ Incentive.type = 'dealer'
⋈ Incentive.incentive_id = Vehicle_Incentive.fk_incentive_id
⋈ Vehicle_Incentive.fk_vehicle_id = Vehicle.vehicle_id
Incentive
⋈ Vehicle.fk_model_id = Model.model_id
Vehicle_Incentive
⋈ Vehicle.fk_make_id = Make.make_id
Model
Vehicle
Make
Type here to search
3:19 PM
11/12/2019

```
    color_id number
    name string
    code string
Vehicle
    vehicle_id number
    fk_make_id number
    fk_model_id number
    year string
Inventory
    inventory_id
    number
    fk_vehicle_id
    number
    fk_color_id number
    price number
Incentive
    incentive_id number
    type string
    amount number
    conditions string
Vehicle_Incentive
    fk_vehicle_id
    number
    fk_incentive_id
    number
    valid_till string
```

π Make.make_name, Model.model_name, Incentive.amount (σ Incentive.type = 'dealer' ((((Vehicle ⋈ Vehicle.fk_make_id = Make.make_id (Make)) ⋈ Vehicle.fk_model_id = Model.model_id (Model)) ⋈ Vehicle_Incentive.fk_vehicle_id = Vehicle.vehicle_id (Vehicle_Incentive)) ⋈ Incentive.incentive_id = Vehicle_Incentive.fk_incentive_id (Incentive)))

Make.make_name	Model.model_name	Incentive.amount
Make1	Model1	400
BMW	Focus	400

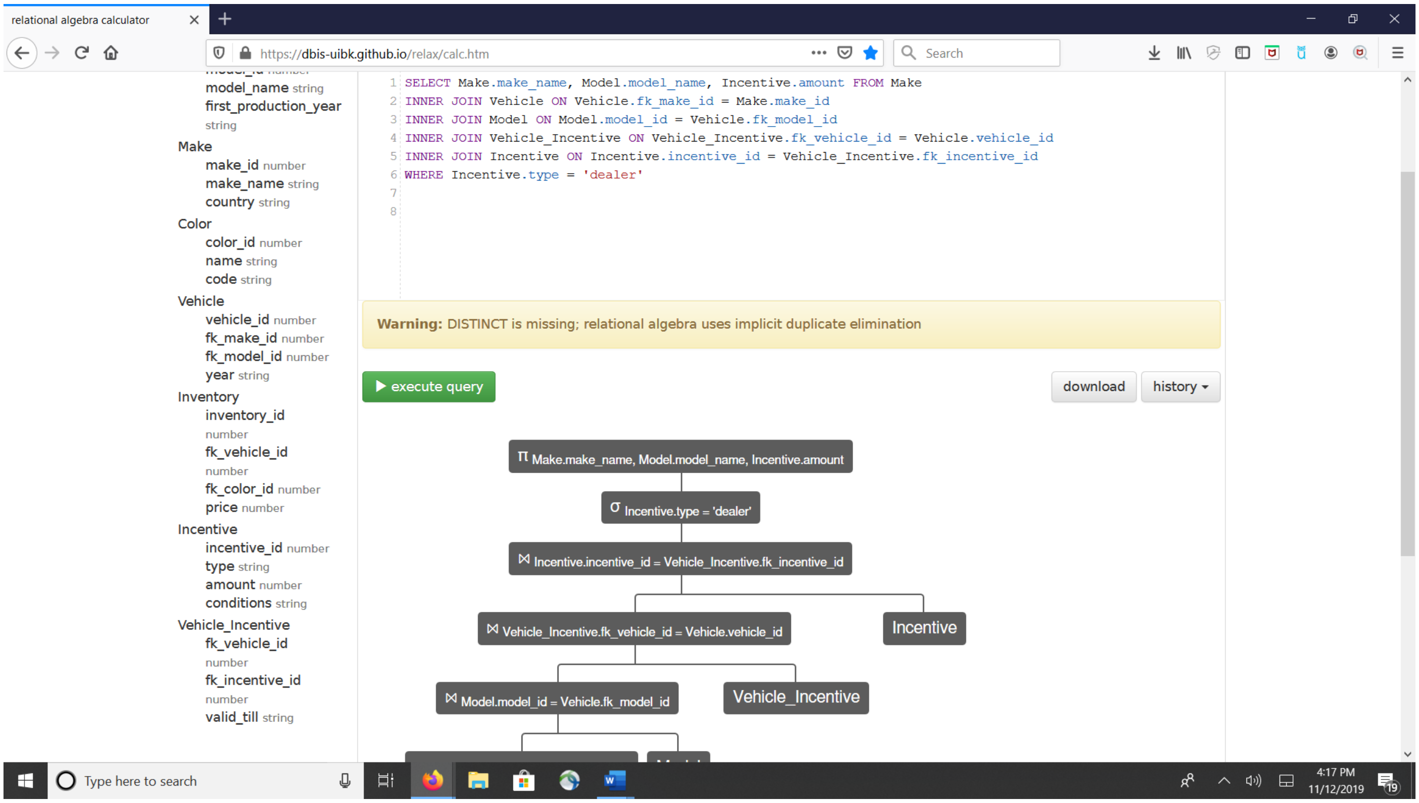
4. Convert the following query to relational algebra
   SELECT Player.id, Team.name, City.name FROM Player
   INNER JOIN Team ON Player.team_id = Team.id
   INNER JOIN City ON Team.city_id = City.id
   WHERE Player.score = 100;

π Player.id, Team.name, City.name (

σ Player.score = '100' ((Team ⋈ Team.id = Player.team_id (Player))

⋈ Team.city_id = City.id (City)))

5. For problem 3 above, convert your relational algebra query into a SQL query


relational algebra calculator
https://dbis-uibk.github.io/relax/calc.htm
Search
execute query
download
history
π Make.make_name, Model.model_name, Incentive.amount
σ Incentive.type = 'dealer'
⋈ Incentive.incentive_id = Vehicle_Incentive.fk_incentive_id
⋈ Vehicle_Incentive.fk_vehicle_id = Vehicle.vehicle_id
Incentive
⋈ Model.model_id = Vehicle.fk_model_id
Vehicle_Incentive
Type here to search
4:17 PM
11/12/2019

model_id number
model_name string
first_production_year string

Make
- make_id number
- make_name string
- country string

Color
- color_id number
- name string
- code string

Vehicle
- vehicle_id number
- fk_make_id number
- fk_model_id number
- year string

Inventory
- inventory_id number
- fk_vehicle_id number
- fk_color_id number
- price number

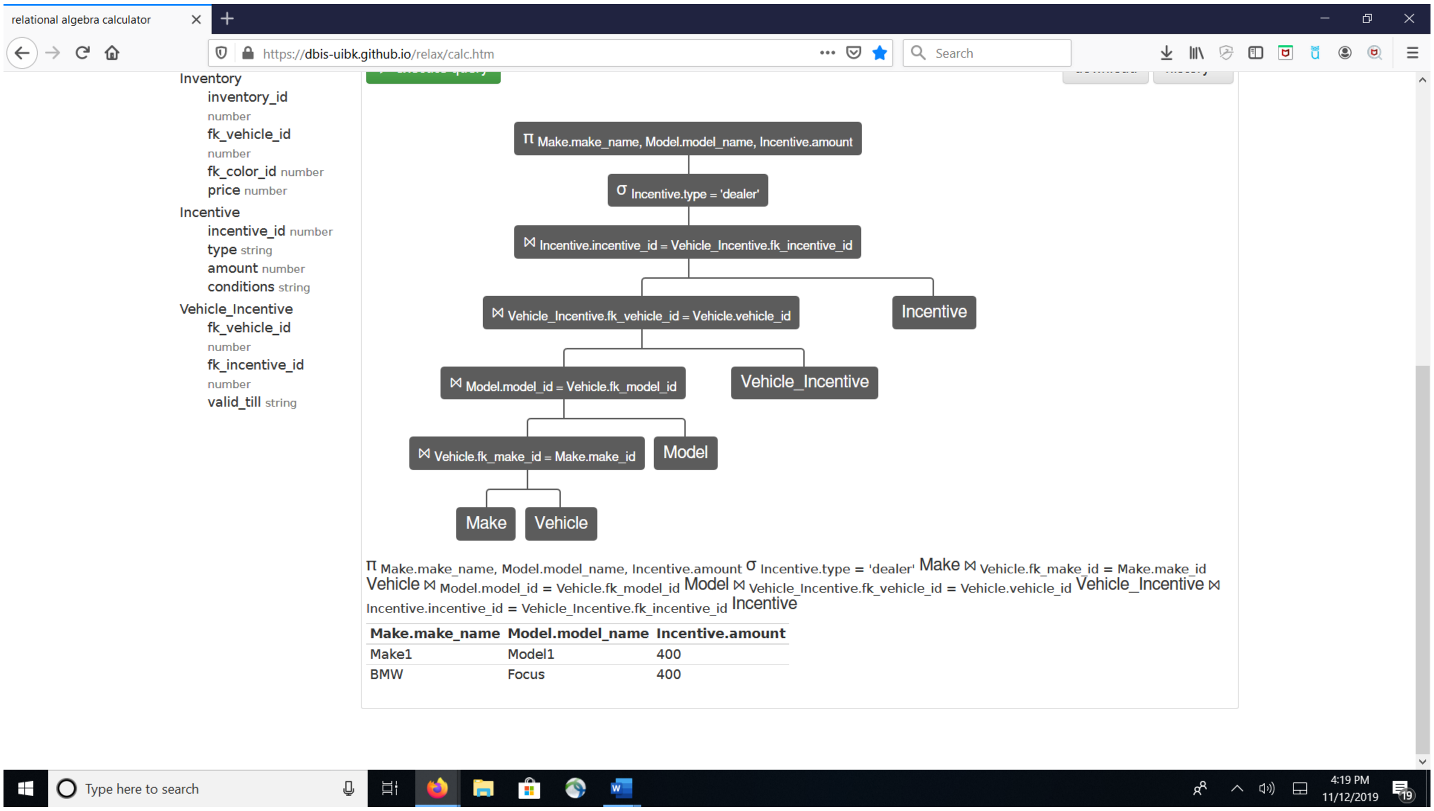
Incentive
- incentive_id number
- type string
- amount number
- conditions string

Vehicle_Incentive
- fk_vehicle_id number
- fk_incentive_id number
- valid_till string

```
SELECT Make.make_name, Model.model_name, Incentive.amount FROM Make
INNER JOIN Vehicle ON Vehicle.fk_make_id = Make.make_id
INNER JOIN Model ON Model.model_id = Vehicle.fk_model_id
INNER JOIN Vehicle_Incentive ON Vehicle_Incentive.fk_vehicle_id = Vehicle.vehicle_id
INNER JOIN Incentive ON Incentive.incentive_id = Vehicle_Incentive.fk_incentive_id
WHERE Incentive.type = 'dealer'
```

Warning: DISTINCT is missing; relational algebra uses implicit duplicate elimination

second page


relational algebra calculator
https://dbis-uibk.github.io/relax/calc.htm
Search
π Make.make_name, Model.model_name, Incentive.amount
σ Incentive.type = 'dealer'
⋈ Incentive.incentive_id = Vehicle_Incentive.fk_incentive_id
⋈ Vehicle_Incentive.fk_vehicle_id = Vehicle.vehicle_id
Incentive
⋈ Model.model_id = Vehicle.fk_model_id
Vehicle_Incentive
⋈ Vehicle.fk_make_id = Make.make_id
Model
Make
Vehicle
Type here to search
4:19 PM
11/12/2019

Inventory
- inventory_id number
- fk_vehicle_id number
- fk_color_id number
- price number

Incentive
- incentive_id number
- type string
- amount number
- conditions string

Vehicle_Incentive
- fk_vehicle_id number
- fk_incentive_id number
- valid_till string

π Make.make_name, Model.model_name, Incentive.amount σ Incentive.type = 'dealer' Make ⋈ Vehicle.fk_make_id = Make.make_id Vehicle ⋈ Model.model_id = Vehicle.fk_model_id Model ⋈ Vehicle_Incentive.fk_vehicle_id = Vehicle.vehicle_id Vehicle_Incentive ⋈ Incentive.incentive_id = Vehicle_Incentive.fk_incentive_id Incentive

Make.make_name	Model.model_name	Incentive.amount
Make1	Model1	400
BMW	Focus	400