</talentlabs>

App Fullstack Developer Program
Work with MySQL workbench

Instruction:

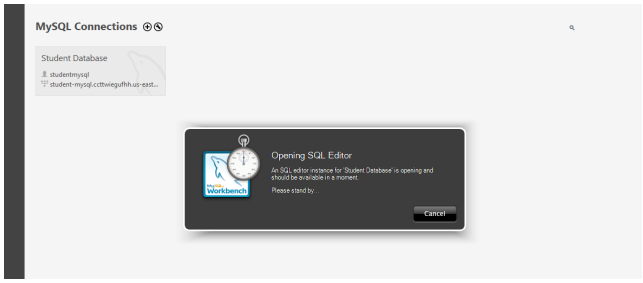
1. Please download this file to your local computer.
2. For each question, fill in your answers in the box.
3. For each checkpoint, paste a screenshot there.
4. For submission,
    - Save this document to a PDF file.
    - Upload your source code to Talentlabs Classroom. Please don't include the node_modules folder.
    - There will be 2 submission files: code.zip and assignment.pdf

Prepare: Setting up MySQL WorkBench

Instruction

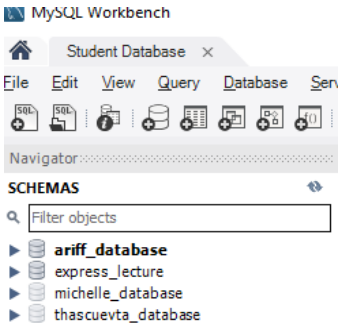
1. Download from here https://www.mysql.com/products/workbench/
2. Install the application.
3. Create a new Database connection by clicking the "+" button:

4. Fill in the connection details:
   Connection Name: Any name is ok here
   Hostname: student-mysql.ccttwiegufhh.us-east-2.rds.amazonaws.com
   Username: studentmysql
   Password: studentmysql
5. Click "Test connection" and it should work.
6. Click "OK" to store the connection.
7. Double click the connection to connect:



Task: Select your own schema as the default [10~15 mins]

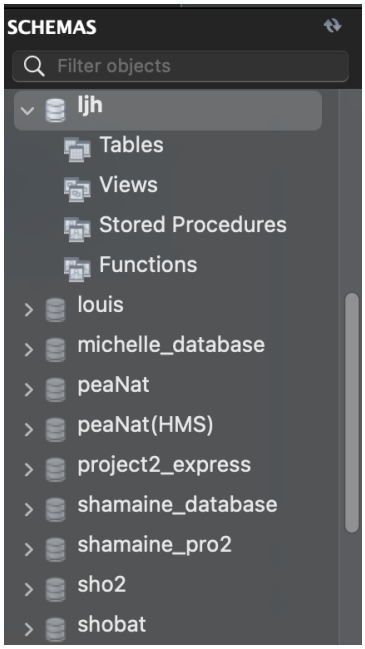
You should see this now in the application:



There are multiple schemas in this database, they are isolated. If it is your first time connecting to this database, you can create a new schema for your project. Please select your schema as the default. You can do it by **right-clicking and selecting "Set as Default Schema"**. Don't access and update other people's data.

After this, the selected schema should be bold.

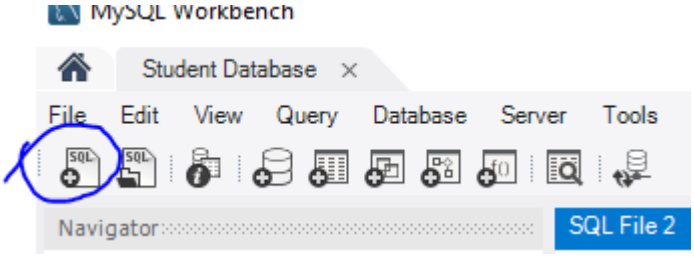
Checkpoint:
Paste a screenshot here. The assigned schema now should be bold.



</talentlabs>

Task: Execute SQL statement [10 ~ 15 mins]

Click the button shown below to create a new SQL editor.

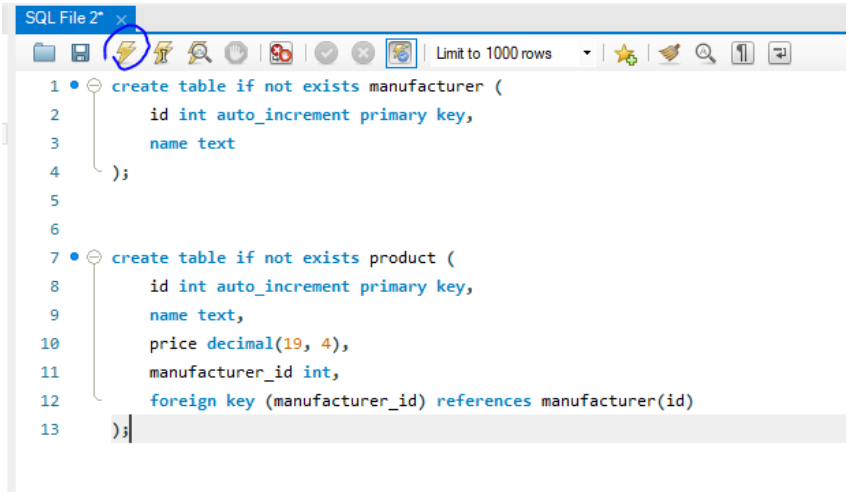


Enter the following SQL statements into the editor:

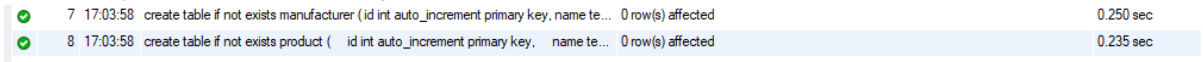
```
create table if not exists manufacturer (
      id int auto_increment primary key,
      name text
);


create table if not exists product (
    id int auto_increment primary key,
    name text,
    price decimal(19, 4),
    manufacturer_id int,
    foreign key (manufacturer_id) references manufacturer(id)
);
```

Execute the statement by clicking this button



After this you should see the green check icons in the action output below:

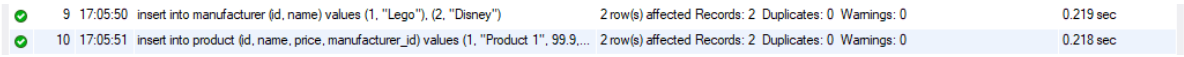


Next let's try to insert some data into the database by executing the following statement:

```
insert into manufacturer (id, name)
values (1, "Lego"), (2, "Disney");

insert into product (id, name, price, manufacturer_id)
values
    (1, "Product 1", 10, 1),
    (2, "Product 2", 20, 1),
    (3, "Product 3", 30, 1),
    (4, "Product 1", 40, 2),
    (5, "Product 1", 50, 2);
```

After the insertion, you should see 2 checks as well.



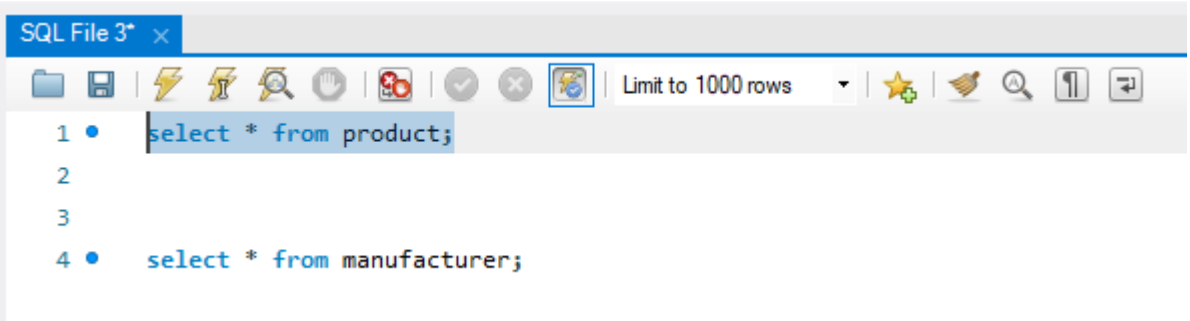
Let's try to query the item out using

```
select * from product;
```

and

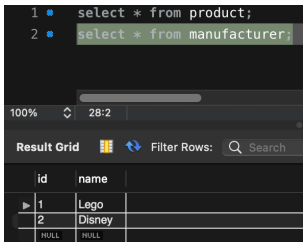
```
select * from manufacturer;
```

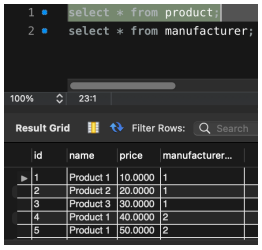
If you have more than one SQL statements in the editor, you can execute the selected sql statement only:



Checkpoint

Show the result of the above 2 select SQL statements. You should see there are 2 manufacturers and 5 products.



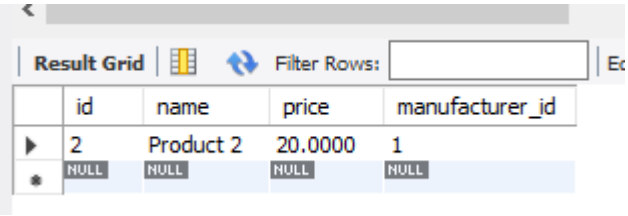


</talentlabs>

Task: SQL Review [30 ~ 60 mins]

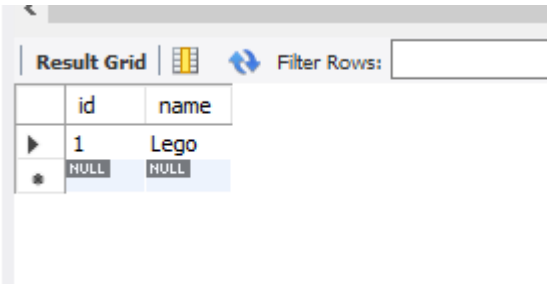
In this section please write the SQL statement to do the following

1. Get a product with id = 2.



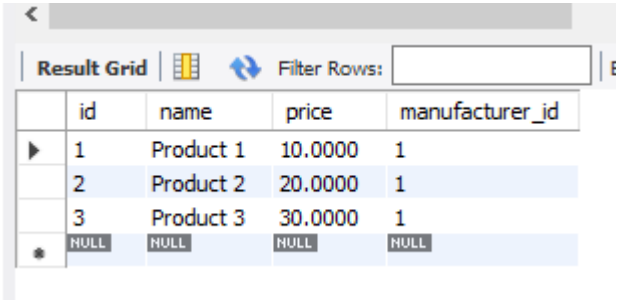
id	name	price	manufacturer_id
2	Product 2	20.0000	1
NULL	NULL	NULL	NULL

2. Get a manufacturer with id = 1.



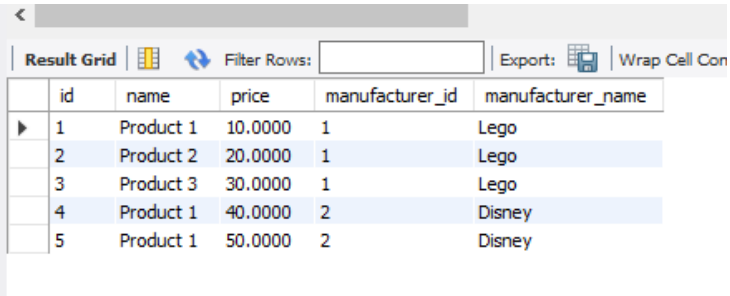
id	name
1	Lego
NULL	NULL

3. Get all products with manufacturer_id = 1



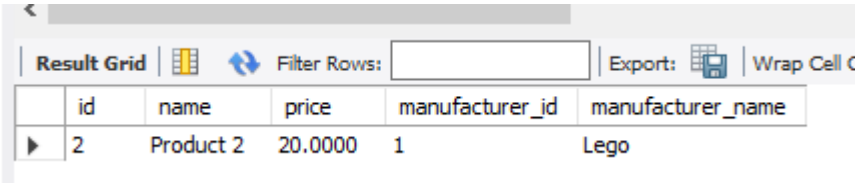
id	name	price	manufacturer_id
1	Product 1	10.0000	1
2	Product 2	20.0000	1
3	Product 3	30.0000	1
NULL	NULL	NULL	NULL

4. Get all products and in the result you should show its manufacturer's name (Hint: Joining)



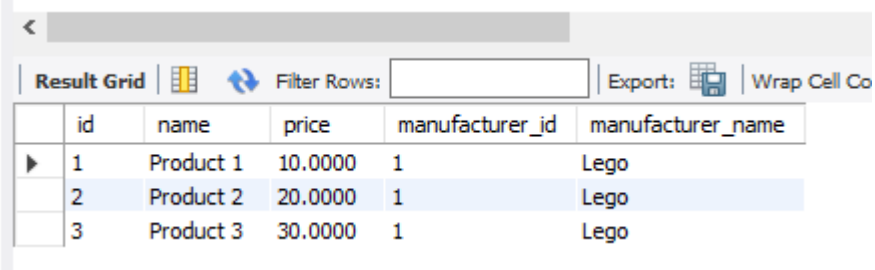
id	name	price	manufacturer_id	manufacturer_name
1	Product 1	10.0000	1	Lego
2	Product 2	20.0000	1	Lego
3	Product 3	30.0000	1	Lego
4	Product 1	40.0000	2	Disney
5	Product 1	50.0000	2	Disney

5. Get a product with id = 2 and in the result you should show its manufacturer's name (Hint: Joining)



id	name	price	manufacturer_id	manufacturer_name
2	Product 2	20.0000	1	Lego

6. Get all products with manufacturer_id = 1and in the result you should show there manufacturer's name (Hint: Joining)

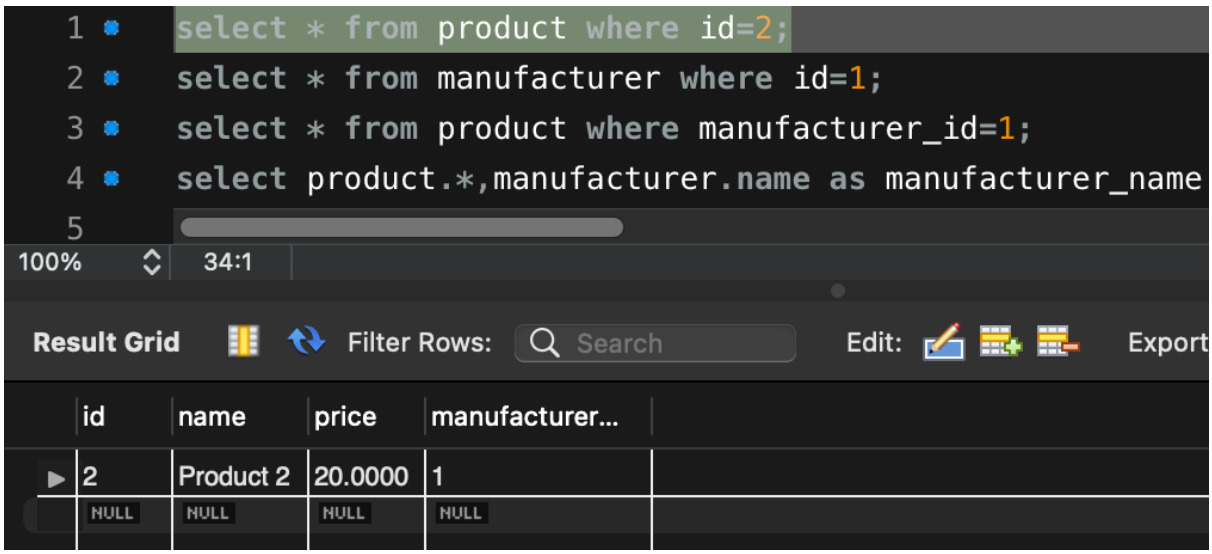


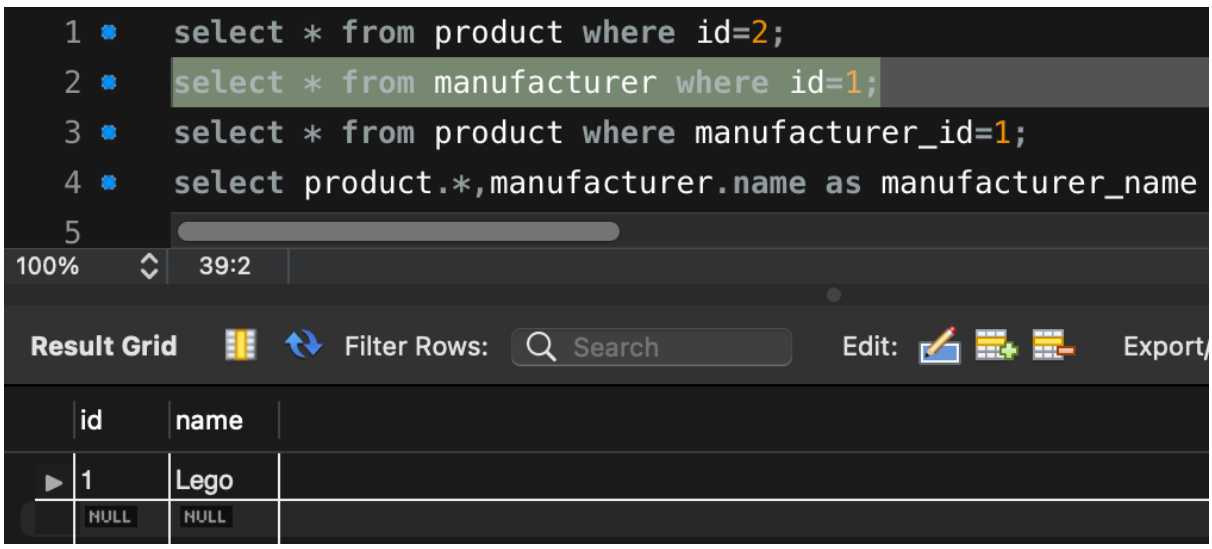
Result Grid | Filter Rows: | Export: | Wrap Cell Co

id	name	price	manufacturer_id	manufacturer_name
1	Product 1	10.0000	1	Lego
2	Product 2	20.0000	1	Lego
3	Product 3	30.0000	1	Lego

Checkpoint

Paste the screenshots of the results of the above 5 queries here:





```
1  select * from product where id=2;
2  select * from manufacturer where id=1;
3  select * from product where manufacturer_id=1;
4  select product.*,manufacturer.name as manufacturer_name
5
```

100% 47:3

Result Grid | Filter Rows: Search | Edit: | Export/In

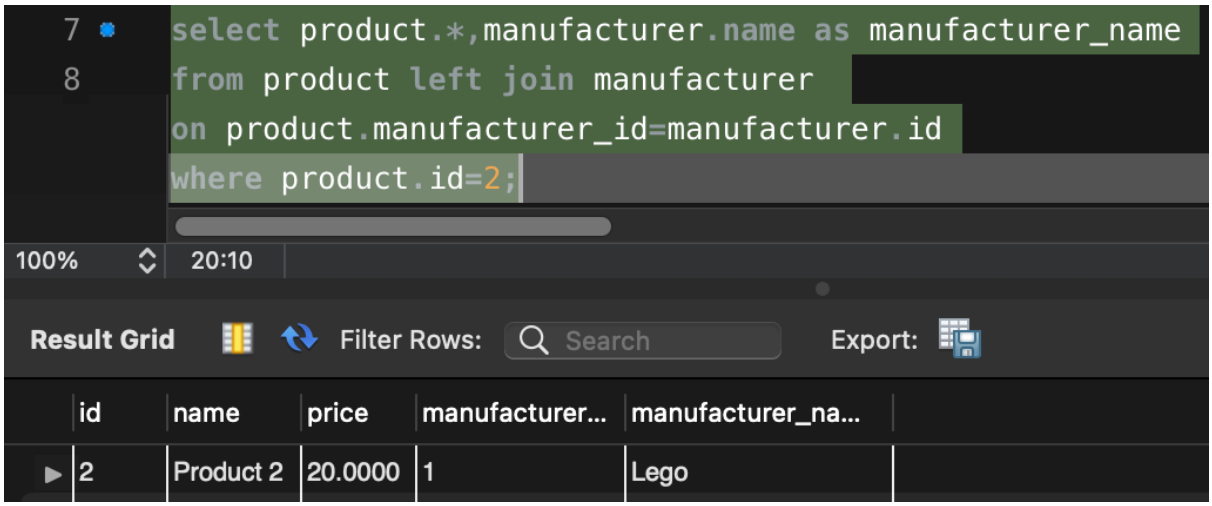
id	name	price	manufacturer...
1	Product 1	10.0000	1
2	Product 2	20.0000	1
3	Product 3	30.0000	1
NULL	NULL	NULL	NULL

```
4  select product.*,manufacturer.name as manufacturer_name
5  from product left join manufacturer
   on product.manufacturer_id=manufacturer.id;
   select product.*,manufacturer.name as manufacturer_name
```

100% 44:6

Result Grid | Filter Rows: Search | Export:

id	name	price	manufacturer...	manufacturer_na...
1	Product 1	10.0000	1	Lego
2	Product 2	20.0000	1	Lego
3	Product 3	30.0000	1	Lego
4	Product 1	40.0000	2	Disney
5	Product 1	50.0000	2	Disney

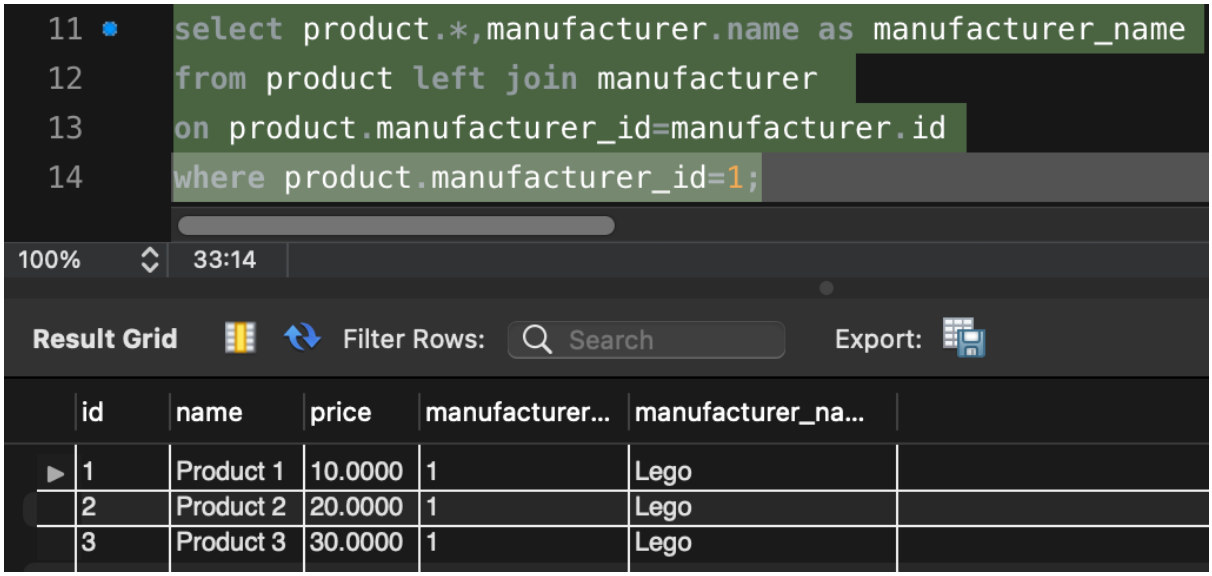


```
7  select product.*,manufacturer.name as manufacturer_name
8  from product left join manufacturer
   on product.manufacturer_id=manufacturer.id
   where product.id=2;
```

100% 20:10

Result Grid Filter Rows: Search Export:

id	name	price	manufacturer...	manufacturer_na...
2	Product 2	20.0000	1	Lego



```
11 select product.*,manufacturer.name as manufacturer_name
12 from product left join manufacturer
13 on product.manufacturer_id=manufacturer.id
14 where product.manufacturer_id=1;
```

100% 33:14

Result Grid Filter Rows: Search Export:

id	name	price	manufacturer...	manufacturer_na...
1	Product 1	10.0000	1	Lego
2	Product 2	20.0000	1	Lego
3	Product 3	30.0000	1	Lego