**FINAL Exam: DATABASE**

**WEP 2022**

**{3h 00}**

15 Mar 2022

* All documents allowed
* Chatting and talking to other students are forbidden
* You are allowed to use PhpMyAdmin or VSCode or any other software
  + *But it is not required*

|  |  |
| --- | --- |
| EXERCICES | POINTS |
| EXERCICE 1 | 50 |
| EXERCICE 2 | 50 |
| **TOTAL** | **100** |

# Exercise 1 – BUS TRAVELS

Before start:

* Open PHPMyAdmin and import the database database/pnc\_bus\_station.sql

We want to manage the bus trips in Cambodia

# 

**Q1 – (10 POINTS)** Write a query to display the first name and last name of the customers who are in **bus number 2**.

|  |
| --- |
| SELECT customers.first\_name, customers.last\_name FROM customers  INNER JOIN customer\_travels ON customer\_travels.customer\_id = customers.id  INNER JOIN travels ON travels.travel\_number = customer\_travels.travel\_id  WHERE travels.bus\_number = 2; |

**Expected result:**



**Q2 – (10 POINTS)** Write a query to **create a view** to display data like expected result of the travels managed by the driver named *“HIM HEY”*.

|  |
| --- |
| CREATE VIEW travel\_manage\_by\_him AS  SELECT travels.travel\_number, provinces.name as"Province", travels.departure\_date, travels.arrival\_date FROM travels  INNER JOIN provinces ON travels.arrival\_provice\_id = provinces.id  INNER JOIN drivers ON drivers.id = travels.driver\_id  WHERE drivers.first\_name = "HIM" AND drivers.last\_name = "HEY"; |

**Expected result:**



**Q3 – (10 POINTS)** Use the view above to display only the travels of driver *“HIM HEY”* on departure date: **2021-09-17**

*Note: you need to use date function to format datetime in MYSQL.*

*Ex: DATE (“*2021-09-17 10:29:20”) = ”2021-09-17”.

|  |
| --- |
| SELECT \* FROM `travel\_manage\_by\_him` WHERE date(departure\_date) = "2021-09-17"; |

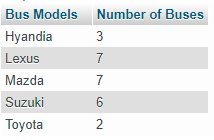
**Expected result:**

****

**Q4 – (10 POINTS)** Write the query to display **the number of buses** for each bus model

|  |
| --- |
| SELECT bus\_models.name as"Bus model", COUNT(buses.bus\_number) as "Number of buses"  FROM bus\_models  LEFT JOIN buses ON bus\_models.id = buses.bus\_model\_id  GROUP BY bus\_models.name; |

**Expected result:**



**Q5 – (10 POINTS)** Write query to display the **bus model** include the **first name** and **last name** of the customers who have traveled with bus **number 1**.

|  |
| --- |
| SELECT customers.first\_name, customers.last\_name, bus\_models.name as"Bus model" FROM customers  INNER JOIN customer\_travels ON customers.id = customer\_travels.customer\_id  INNER JOIN travels ON travels.travel\_number = customer\_travels.travel\_id  INNER JOIN buses ON travels.bus\_number = buses.bus\_number  INNER JOIN bus\_models ON bus\_models.id = buses.bus\_model\_id  WHERE buses.bus\_number = 1; |

**Expected result:**



# Exercise 2 – PNC Facebook

We want to create a new **Facebook application** for PNC’s students. Please read the requirements and description of database for this application.

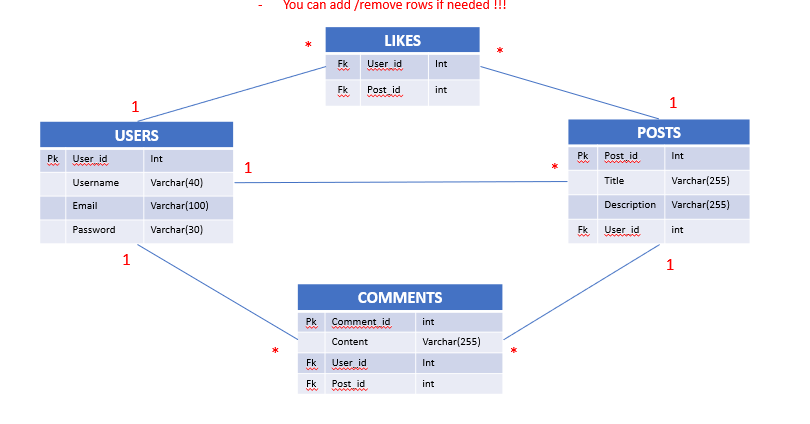
* **Users** are described by their full name, gender, email and password.
* **Posts** are described by their title, description and image (the name of the image file).
* **Comments** are described by their content.
* **Likes** contain user and post information.

Additional information about the relations between entities:

* A user can share many posts.
* A post is created by one user.
* A post can be commented by many users.
* A comment is created by one user.
* A post can have many likes from many users, and one like is related to one post.

**Q1 – (20 POINTS)** Design the ERD Physical Model related to this problem

* Create your ERD in the file EXERCICE-2-ERD.PPTX
* include the keys, types, and all necessary tables



**Q2 – (10 POINTS)** Insert fake data in the database: at least 5 rows in each table.

|  |
| --- |
|  |

**Q3 – (5 POINTS) -** Write a query to display **the user who have posted the most** on PNC Facebook.

|  |
| --- |
| SELECT users.user\_id, users.username, COUNT(posts.post\_id) as "number of posts" FROM posts INNER JOIN users ON users.user\_id = posts.user\_id GROUP BY (user\_id) ORDER BY(COUNT(posts.post\_id)) DESC LIMIT 1; |

**Q4 – (5 POINTS)** - Create a **view** to show the **title of the post** with the **highest number of likes**

|  |
| --- |
| CREATE VIEW postThatGetTheMostLike AS  SELECT posts.post\_id, posts.title, COUNT(likes.user\_id) as "number of likes" FROM likes INNER JOIN posts ON posts.post\_id = likes.post\_id GROUP BY(posts.post\_id) ORDER BY(COUNT(likes.user\_id)) DESC LIMIT 1; |

**Q5 – (10 POINTS) -** Write the query to display, for each comment **on the post of ID 1**, the username of the comment’s author and the comment’s content.

|  |
| --- |
| [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) users.username, comments.content FROM comments INNER JOIN users ON users.user\_id = comments.user\_id INNER join posts ON comments.post\_id = posts.post\_id WHERE comments.post\_id = 1; |