CODE / COURSE	**DFN40323- PROGRAMMING ESSENTIALS IN PYTHON**	**PRACTICAL TASK**	**~~5~~**
PROGRAM / CLASS	**DDT4**	**DURATION**	**3 HOURS**
STUDENT'S NAME	**1) SHELAN A/L PONNAN** **2) MUHAMMAD AFIQ MUHAIMIN BIN MOHD ZAINI**	**CLO 1**	**P3**
REG. NO.	**1) 32DDT20F2001** **2) 32DDT20F2029**	**TOTAL MARKS**	**/40**
LECTURER'S NAME	**SHARIZAN BINTI ABDUL JAMIL**		

Learning Outcome:

By the end of this practical, student will able to:

Construct a software application using the Python programming language (CLO1, P3, PLO3).

Instructions:

Answer all the questions given. Students need to discuss in groups of two (2) and upload the findings of the discussion in report and .py file through CIDOS. Students will be accessed according to the Rubric given.

Question 1

By using Python codes,

1. Create a database name **Social Media Application**. The database will consist of four tables:
   i. **users** contain general information about users and has the following attributes:
      a) id
      b) name
      c) age
      d) gender
      e) nationality

   ii. **likes** contain information about user who likes the posts and has the following attributes:
      a) id

b) user_id

c) post_id

iii. **posts** contain information about posts and has the following attributes:

a) id

b) title

c) description

d) user_id

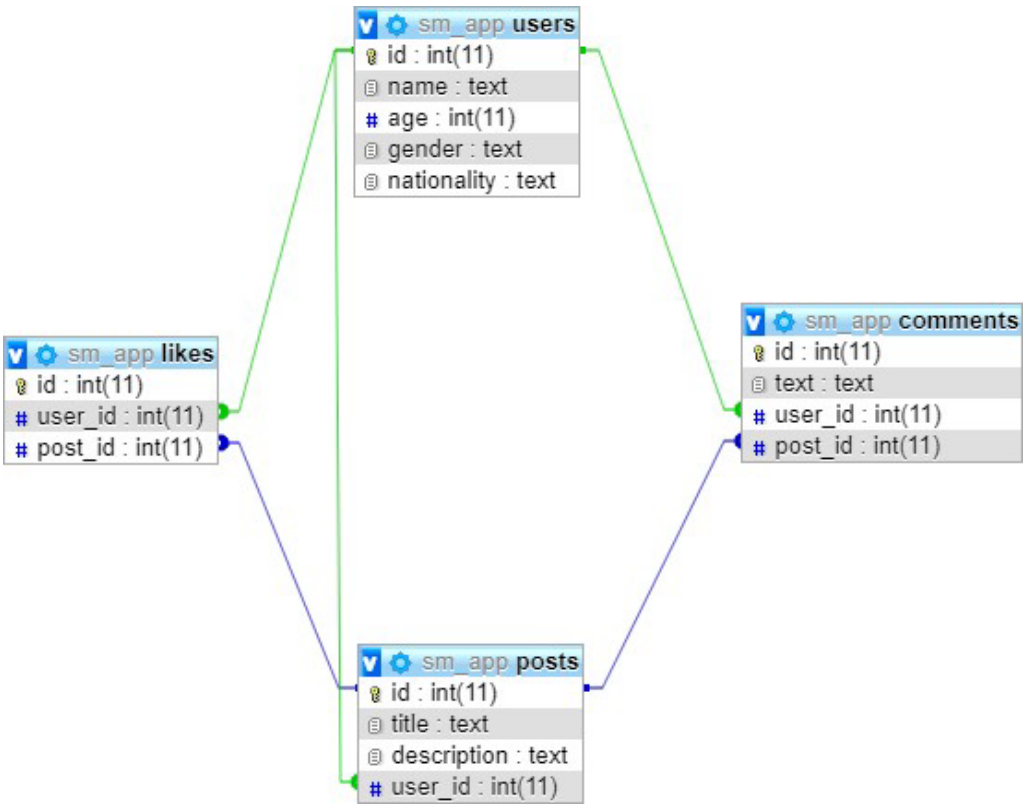
iv. **comments** contain information about user who comments the posts and has the following attributes:

a) id

b) text

c) user_id

d) post_id



fo

Figure 1: Schema Diagram for Social Media Application

2. Add **FOUR (4)** suitable data into each table created.
3. View all data from each table using correct syntax.

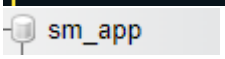
(25 marks)

SOURCE CODE & OUTPUT:

createdb.py

```
#Import mysql connector
import mysql.connector

#Connect to database
carwashdb = mysql.connector.connect(host="localhost", user="root", password="")
#declare cursor as the worker for our database
executor = carwashdb.cursor()
#Create database named sm_app
executor.execute("CREATE DATABASE sm_app")
```



createconn.py

```
#Import mysql connector
import mysql.connector

#declare smdb and check connection to db
smdb = mysql.connector.connect(host="localhost", user="root", password="")
#print the result
print(smdb)
```

```
<mysql.connector.connection_cext.CMySQLConnection object at 0x0000017611D17580>
```

users.py

```
#Import mysql connector
import mysql.connector

#Connect to database named sm_app
smdb = mysql.connector.connect(host="localhost",
                               user="root",
                               password="",
                               database="sm_app")
executor = smdb.cursor()
#Declare userRecord to create table named users
userRecord = """create table users(
    id int(11) auto_increment not null,
    name varchar(255),
    age int(11),
    gender varchar(7),
    nationality varchar(100),
    primary key(id)
    )"""
#Execute the command
executor.execute(userRecord)
#Close the connection
smdb.close()
```

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	id	int(11)			No	*None*		AUTO_INCREMENT	Change Drop More
☐	2	name	varchar(255)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More
☐	3	age	int(11)			Yes	*NULL*			Change Drop More
☐	4	gender	varchar(7)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More
☐	5	nationality	varchar(100)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More

posts.py

```
#Import mysql connector
import mysql.connector

#Connect to database named sm_app
smdb = mysql.connector.connect(host="localhost",
                               user="root",
                               password="",
                               database="sm_app")
executor = smdb.cursor()
#Declare userRecord to create table named posts
userRecord = """create table posts(
    id int (11) auto_increment not null,
    title varchar (60),
    description varchar (255),
    user_id int (11),
    primary key (id),
    foreign key (user_id) references users (id)
    )"""
#Execute the command
executor.execute(userRecord)
#Close Connection
smdb.close()
```

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	id	int(11)			No	*None*		AUTO_INCREMENT	Change Drop More
☐	2	title	varchar(60)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More
☐	3	description	varchar(255)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More
☐	4	user_id	int(11)			Yes	*NULL*			Change Drop More

likes.py

```
#Import mysql connector
import mysql.connector

#Connect to database named sm_app
smdb = mysql.connector.connect(host="localhost",
                               user="root",
                               password="",
                               database="sm_app")
executor = smdb.cursor()
#Declare userRecord to create table named likes
userRecord = """create table likes(
    id int (11)auto_increment not null,
    user_id int(11),
    post_id int(11),
    primary key(id),
    foreign key(post_id) references posts(id),
    foreign key(user_id) references users(id)
    )"""
#Execute the command
executor.execute(userRecord)
#Close the connection
smdb.close()
```

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	id	int(11)			No	*None*		AUTO_INCREMENT	Change Drop More
☐	2	user_id	int(11)			Yes	*NULL*			Change Drop More
☐	3	post_id	int(11)			Yes	*NULL*			Change Drop More

comments.py

```
#Import mysql connector
import mysql.connector

#Connect to databased named sm_app
smdb = mysql.connector.connect(host="localhost",
                               user="root",
                               password="",
                               database="sm_app")
executor = smdb.cursor()
#Declare userRecord to create table named comments
userRecord = """
create table comments(
    id int(11) auto_increment not null,
    text varchar(255),
    user_id int(11),
    post_id int(11),
    primary key (id),
    foreign key (post_id) references posts (id),
    foreign key (user_id) references users (id)
)"""
#Execute the command
executor.execute(userRecord)
#Close the connection
smdb.close()
```

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	id	int(11)			No	*None*		AUTO_INCREMENT	Change Drop More
☐	2	text	varchar(255)	utf8mb4_general_ci		Yes	*NULL*			Change Drop More
☐	3	user_id	int(11)			Yes	*NULL*			Change Drop More
☐	4	post_id	int(11)			Yes	*NULL*			Change Drop More

insertuser.py

```python
#Import mysql connector
import mysql.connector

#Connect to database named sm_app
smdb = mysql.connector.connect(host="localhost",
                               user="root",
                               password="",
                               database="sm_app")
executor = smdb.cursor()

#Insert multiple sql exec at once
comm = "INSERT INTO users (name,age,gender,nationality) VALUES (%s,%s,%s,%s)"
val = [("Abu", 19, "Male", "Malaysian"), ("Zaid", 20, "Male", "Indian"),
       ("Chong", 30, "Male", "Chinese"), ("Zalya", 25, "Female", "Malaysian")]

#Exec more then one command
executor.executemany(comm, val)
#Confirm the changes
smdb.commit()
#show how much record are inserted
print(executor.rowcount, "Record inserted")
```

```
4 Record inserted
```

←T→			id	name	age	gender	nationality
☐	Edit	Copy Delete	1	Abu	19	Male	Malaysian
☐	Edit	Copy Delete	2	Zaid	20	Male	Indian
☐	Edit	Copy Delete	3	Chong	30	Male	Chinese
☐	Edit	Copy Delete	4	Zalya	25	Female	Malaysian

insertposts.py

```python
#Import mysql connector
import mysql.connector

#Connect to database named sm_app
smdb = mysql.connector.connect(host="localhost",
                               user="root",
                               password="",
                               database="sm_app")
executor = smdb.cursor()

#Insert multiple sql exec at once
comm = "INSERT INTO posts (ID, TITLE, DESCRIPTION, USER_ID) VALUES (%s, %s, %s, %s)"
val = [(111, "Anda Mahu Makan?", "Makanan", 1),
       (112, "Anda Mahu Tidur?", "Tidur", 2),
       (115, "Anda Lapar?", "Masak Nasik", 3),
       (114, "Anda mahu jadi pro?", "training ah", 4)]
      You, 7 hours ago • pt5 comment broken
#Exec more then one command
executor.executemany(comm, val)
#Confirm the changes
smdb.commit()
#show how much record are inserted
print(executor.rowcount, "Record inserted")
```

```
4 Record inserted
```

	id	title	description	user_id
Edit Copy Delete	111	Anda Mahu Makan?	Makanan	1
Edit Copy Delete	112	Anda Mahu Tidur?	Tidur	2
Edit Copy Delete	114	Anda mahu jadi pro?	training ah	4
Edit Copy Delete	115	Anda Lapar?	Masak Nasik	3

insertcomments.py

```python
You, 8 seconds ago | 1 author (You)
#Import mysql connector
import mysql.connector

#Connect to database named sm_app
smdb = mysql.connector.connect(host="localhost",
                               user="root",
                               password="",
                               database="sm_app")
executor = smdb.cursor()

#Insert multiple sql exec at once
comm = "INSERT INTO comments (ID, TEXT, USER_ID, POST_ID) VALUES (%s, %s, %s, %s)"
val = [(561, "Ko nak makan apa?", 1, 111), (566, "Tido kat ane?", 2, 112),
       (571, "Training apa tu miska?", 3, 114),
       (545, "Nasi je ke, lauknya mana?", 4, 115)]

#Exec more then one command
executor.executemany(comm, val)
#Confirm the changes
smdb.commit()
#show how much record are inserted
print(executor.rowcount, "Record inserted")
```

```
4 Record inserted
```

	id	text	user_id	post_id
Edit Copy Delete	545	Nasi je ke, lauknya mana?	4	115
Edit Copy Delete	561	Ko nak makan apa?	1	111
Edit Copy Delete	566	Tido kat ane?	2	112
Edit Copy Delete	571	Training apa tu miska?	3	114

insertlikes.py

```
You, 13 seconds ago | 1 author (You)
#Import mysql connector
import mysql.connector

#Connect to database named sm_app
smdb = mysql.connector.connect(host="localhost",
                               user="root",
                               password="",
                               database="sm_app")
executor = smdb.cursor()

#Insert multiple sql exec at once
comm = "INSERT INTO likes (ID, USER_ID, POST_ID) VALUES (%s, %s, %s)"
val = [(666, 1, 111), (777, 2, 115), (888, 3, 114), (999, 4, 112)]

#Exec more then one command
executor.executemany(comm, val)
#Confirm the changes
smdb.commit()
#show how much record are inserted
print(executor.rowcount, "Record inserted")
```

```
4 Record inserted
```

		id	user_id	post_id
☐	Edit Copy Delete	666	1	111
☐	Edit Copy Delete	777	2	115
☐	Edit Copy Delete	888	3	114
☐	Edit Copy Delete	999	4	112

Display data

displayusers.py

```
#Import mysql connector
import mysql.connector

#Connect to database named sm_app
smdb = mysql.connector.connect(host="localhost",
                               user="root",
                               password="",
                               database="sm_app")
executor = smdb.cursor()

#Execute command to list all users from db
executor.execute("SELECT * FROM USERS")
usr_list = executor.fetchall()
print(usr_list)
```

```
[(1, 'Abu', 19, 'Male', 'Malaysian'), (2, 'Zaid', 20, 'Male', 'Indian'), (3, 'Chong', 30, 'Male', 'Chinese'), (4, 'Zalya', 25, 'Female', 'Malaysian')]
```

displayposts.py

```
#Import mysql connector
import mysql.connector

#Connect to database named sm_app
smdb = mysql.connector.connect(host="localhost",
                               user="root",
                               password="",
                               database="sm_app")
executor = smdb.cursor()

#Execute command to list all posts from db
executor.execute("SELECT * FROM POSTS")
posts_list = executor.fetchall()
print(posts_list)
```

```
[(111, 'Anda Mahu Makan?', 'Makanan', 1), (112, 'Anda Mahu Tidur?', 'Tidur', 2), (114, 'Anda mahu jadi pro?', 'training ah', 4), (115, 'Anda Lapar?', 'Masak Nasik', 3)]
```

displaycomments.py

```
#Import mysql connector
import mysql.connector

#Connect to database named sm_app
smdb = mysql.connector.connect(host="localhost",
                               user="root",
                               password="",
                               database="sm_app")
executor = smdb.cursor()

#Execute command to list all comments from db
executor.execute("SELECT * FROM COMMENTS")
comments_list = executor.fetchall()
print(comments_list)
```

```
[(545, 'Nasi je ke, lauknya mana?', 4, 115), (561, 'Ko nak makan apa?', 1, 111), (566, 'Tido kat ane?', 2, 112), (571, 'Training apa tu miska?', 3, 114)]
```

displaylikes.py

```
#Import mysql connector
import mysql.connector

#Connect to database named sm_app
smdb = mysql.connector.connect(host="localhost",
                               user="root",
                               password="",
                               database="sm_app")
executor = smdb.cursor()

#Execute command to list all likes from db
executor.execute("SELECT * FROM LIKES")
likes_list = executor.fetchall()
print(likes_list)
```

```
[(666, 1, 111), (777, 2, 115), (888, 3, 114), (999, 4, 112)]
```

Conclusion:

For conclusion, basically we learned on how to manipulate databases with python using multiple attributes that contribute to all of the databases feature and learn on how to insert data into db and learn on how to display all of the data in databases using the help of python