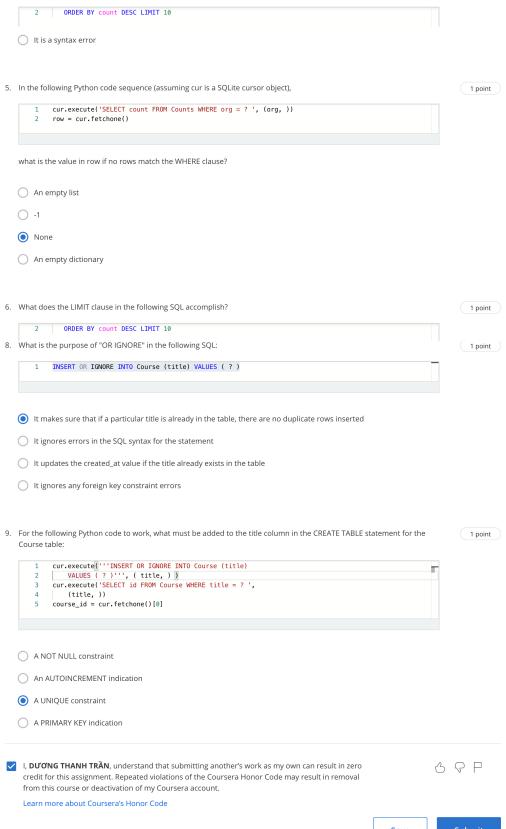# Many-to-Many Relationships and Python

TOTAL POINTS 9

1. How do we model a many-to-many relationship between two database tables?　　　　1 point

  ○ We add 10 foreign keys to each table with names like artict_id_1, artist_id2, etc.

  ○ We use a BLOB column in both tables

  ⦿ We add a table with two foreign keys

  ○ We use the ARRAY column type in both of the tables

2. In Python, what is a database "cursor" most like?　　　　1 point

  ○ A Python dictionary

  ○ A method within a class

  ⦿ A file handle

  ○ A function

3. What method do you call in an SQLIte cursor object in Python to run an SQL command?　　　　1 point

  ○ socket()

  ○ send()

  ○ run()

  ⦿ execute()

4. In the following SQL,　　　　1 point

```
1    cur.execute('SELECT count FROM Counts WHERE org = ? ', (org, ))
```

  what is the purpose of the "?"?

  ○ It allows more than one boolean operation in the WHERE clause

  ○ It is a search wildcard

  ⦿ It is a placeholder for the contents of the "org" variable

  ○ It is a syntax error

5. In the following Python code sequence (assuming cur is a SQLite cursor object),　　　　1 point

```
1    cur.execute('SELECT count FROM Counts WHERE org = ? ', (org, ))
2    row = cur.fetchone()
```

  what is the value in row if no rows match the WHERE clause?

  ○ An empty list

  ○ -1

  ⦿ None

  ○ An empty dictionary

6. What does the LIMIT clause in the following SQL accomplish?　　　　1 point

```
2       ORDER BY count DESC LIMIT 10
```

○ It is a syntax error

5. In the following Python code sequence (assuming cur is a SQLite cursor object), **1 point**

```
1    cur.execute('SELECT count FROM Counts WHERE org = ? ', (org, ))
2    row = cur.fetchone()
```

what is the value in row if no rows match the WHERE clause?

○ An empty list

○ -1

◉ None

○ An empty dictionary

6. What does the LIMIT clause in the following SQL accomplish? **1 point**

```
2       ORDER BY count DESC LIMIT 10
```

8. What is the purpose of "OR IGNORE" in the following SQL: **1 point**

```
1    INSERT OR IGNORE INTO Course (title) VALUES ( ? )
```

◉ It makes sure that if a particular title is already in the table, there are no duplicate rows inserted

○ It ignores errors in the SQL syntax for the statement

○ It updates the created_at value if the title already exists in the table

○ It ignores any foreign key constraint errors

9. For the following Python code to work, what must be added to the title column in the CREATE TABLE statement for the Course table: **1 point**

```
1    cur.execute('''INSERT OR IGNORE INTO Course (title)
2        VALUES ( ? )''', ( title, ) )
3    cur.execute('SELECT id FROM Course WHERE title = ? ',
4        (title, ))
5    course_id = cur.fetchone()[0]
```

○ A NOT NULL constraint

○ An AUTOINCREMENT indication

◉ A UNIQUE constraint

○ A PRIMARY KEY indication

Save   Submit