

Práctica 2 - PyMySQL y Alchemy

Ingeniería de Software 2024-2

Erick Martínez Piza
ermarp@ciencias.unam.mx

Rogelio Alcantar Arenas
rogelio-aa@ciencias.unam.mx

Luis Fernando Yang Fong Baeza
fernandofong@ciencias.unam.mx

Valeria García Landa
vale.garcia.landa@gmail.com

Francisco Valdés Souto
fvaldes@ciencias.unam.mx

8 de Enero 2024

1. Introducción

Las bases de datos son herramientas demasiado útiles hoy en día para almacenar información de manera digital y consultable para cualquier persona que tenga acceso. Su importancia para cualquier sistema de software es vital puesto que todos los sistemas de software requieren de un manejo de información *óptimo* para sus propósitos.

El propósito de ésta práctica es aprender el manejo de las bases de datos a través del lenguaje **Python** para un sistema prueba (que se va a usar para el resto de las prácticas) siguiendo los lineamientos y tecnologías de hoy en día conocido como ORM (Object Relational Mapping) utilizando el framework conocido como Alchemy SQL.

2. Desarrollo

Se les entrega un diagrama relacional en 1 de un sistema para renta de películas. En la rama **2-pymysql-alchemy** del repositorio oficial del curso, se encuentra un script con extensión **.sql** que levanta dicha base de datos que se debe de ejecutar en MySQL Workbench, que además crea un usuario con todos los privilegios sobre la misma base con las siguientes credenciales:

```
username: ferfong  
password: Developer123!
```

Figura 1: Diagrama relacional del sistema Clon-buster

El propósito del usuario es para que dicho usuario tenga todas las credenciales únicamente en la base de datos que se requiere, al contrario de programar siempre con el usuario **root**, si se desea que otro usuario sea creado con este propósito, entonces se puede modificar el script para cambiar el nombre de usuario y ejecutar el query o copiar y pegar el query modificando lo deseable, no se aceptarán prácticas que estén conectándose bajo el usuario **root**. De igual manera, se espera que en su fork del repositorio oficial del curso, puedan sincronizar el repositorio para poder tener la rama 2 y realizar su práctica.

Una vez levantada la base de datos, la práctica consta de dos simples puntos, interacción con la base de datos usando cursores y el uso del framework Alchemy. Por ende, cada punto equivale a 5pts cada uno, sin embargo,

no se calificará de forma binaria, ambos puntos estarán descompuestos en sub-tareas que irán sumando para obtener el total de 5, todo esto utilizando otro entorno virtual nuevo, no necesariamente tiene que ser el entorno de la práctica anterior.

2.1. Interacción con la base de datos usando cursores

Con ayuda de la biblioteca `PyMySQL` y asumiendo que ya esté la base de datos lista para operaciones CRUD (Create, Read, Update, Delete), el alumno debe de realizar un script que pueda realizar lo siguiente:

1. Una función que inserte al menos 1 registro en cada tabla cada vez que sea ejecutada, como se tiene una tabla con llaves foráneas, se tiene que tener en cuenta que existan los registros en las otras dos tablas. (1.25pts)
2. Una función que filtre a la tabla `Usuario` a todos los usuarios cuyo apellido termine en alguna cadena especificada por el usuario. (1.25pts)
3. Una función que dado el nombre de una película (entrada de función o entrada de usuario) y un género, si dicha película existe, se le cambie el género a dicha película. (1.25pts)
4. Una función que elimine todas las rentas anteriores a 3 días a la fecha en que se ejecuta la función, ejemplo, si la ejecuto el 23 de Enero de 2024, todas las rentas que tengan fecha anterior o igual al 19 de Enero de 2024 deberán de ser eliminadas, hasta que dicha función de ejecute. (1.25pts)

Para el caso de las funciones que requieran entrada del usuario, puede ser a manera de `input` o a manera de parámetro, esto es a discreción del alumno.

2.2. Uso del framework de Alchemy

Para este punto de la práctica, el alumno debe instalar Alchemy en el entorno virtual y crear las carpetas correspondientes para respetar la creación de un proyecto de Python y en la carpeta correspondiente crear sus esquemas ORM para cada una de las tablas, si los esquemas fueron creados correctamente, esto ya equivale a 2.5pts.

Posteriormente, como `main` del proyecto, se tiene que crear un menú para darle al usuario las siguientes opciones:

1. Ver los registros de una tabla. (0.25pts)
2. Filtrar los registros de una tabla por id (solo que sea exactamente igual a). (0.25pts)
3. Actualizar la columna nombre de un registro, para el caso de la tabla `Renta`, modificar la fecha de la renta. (1pt)
4. Eliminar un registro por id o todos los registros. (1pt)

En el caso de la puntuación, para llevarse la cantidad especificada, debe de estar bien hecho para las 3 entidades de ORM, si alguna falla o falta, no se podrá dividir entre 3 la cantidad de puntos correspondientes.

3. Entregables

Se debe de marcar como entregada la actividad en Moodle y realizar el `commit/push` a la rama correspondiente en su fork cuyo nombre debe de empezar con 2-, todo esto antes de las 23:59:59 del 21 de Febrero del 2024.