

Definición Reposición #3 Práctico Unidad 2

El examen a continuación se compone de un enunciado que corresponde con el 100% de la calificación.

Lea primero Entre más enriquecidas sean sus respuestas, usted obtendrá una mejor calificación. El ejercicio le ofrece un escenario y le entrega información obligatoria a obtener, sin embargo, se espera que usted complete dicha información mediante la abstracción de la vida real.

No se confíe de los comentarios que escuchó o de afirmaciones que nunca probó. Muchos comentarios en clase se complementaron con “excelente, pruebe lo que indica su compañero” ya que dichos comentarios podrían tener algo de “correctitud” pero al mismo tiempo podrían no entregar el resultado esperado. Sin haberlo probado usted por su cuenta, arrastrará omisiones en la implementación de código.

Haga uso del 100% del conocimiento exclusivo de la Unidad 2 de esta asignatura (contenido como procedimientos almacenados, triggers, expresiones regulares, entre muchos muchos otros).

- El SGBD de examen es idéntico al usado en clase.
- Es obligatorio el aliasing en el DML y los comments en el DDL.
- Mantenga sus clases (imports) separadas de su main de Jupyter.
- Al ser este un examen remoto y de libro abierto, se le recuerda la importancia de los capítulos de lectura del manual del SGBD que contienen numerosas funcionalidades que poseen en común múltiples lenguajes de programación.
- Con realizar este examen el estudiante indica que él es el único autor de las respuestas de su examen.
- Para asegurar una buena revisión haga su código claro, implementando TABULADOS y SALTOS DE LÍNEA en su escritura a mano en su SQL.

Objetivo Se desea hacer una hoja de demostración de Jupyter Notebook incluyendo y reutilizando imports realizados en clase, para generar el código, la documentación, las clases, las instancias y el uso de las instancias para un sistema que abarque los componentes de construcción de tablas, llenado de datos y análisis de los datos, usando los contenidos de la Unidad 2 de la asignatura de BDI. Su entregable será un archivo .7z con todos los archivos necesarios.

Haga un programa de Python que posea sus respectivas clases/objetos, consultas SQL e instancias para la generación de una conexión y uso de múltiples transacciones de SQL dentro de un SGBD MySQL/MariaDB que resuelvan el siguiente problema, tal y como se han ejecutado ejercicios en clase mediante Python3.

Para hacer este programa Python con acceso a base de datos, los diseñadores del sistema han establecido una serie de elementos que el programador debe traducir a código y documentación, los cuales se listan a continuación.

Enunciado del examen Una empresa de desarrollo lo está contratando para implementar un registro de cursos en línea, donde se desea almacenar toda la información de los dispositivos y navegadores web que

visitan dicho sitio, y en donde distintos profesores y estudiantes de todo el mundo crearán y visualizarán, respectivamente, cursos de distintos temas, categorías y capítulos.

Provea de una solución formal listando no menos de 5 entidades del modelo relacional, que estén normalizadas 2.5 NF.

Lea los ejercicios al final para obtener más contexto.

En sus clases de Python (los archivos .py que viven afuera del archivo de Jupyter Main) deben de actualizarse los siguientes elementos. Estos archivos .py serán importados desde su Jupyter Main tal y como fue hecho en clase.

Componente 1 (10 puntos):

Debe existir un método `select()` el cual debe recibir la query, junto con un arreglo con el encabezado de la tabla a seleccionar, y junto con el máximo límite de resultados a imprimir. Este método ya debe imprimir el resultado directamente en pantalla, en lugar de retornarlo.

Debe existir un método `dmsDds()` el cual debe servir para ejecutar instrucciones dms y/o dds no fetch.

Componente 2 (12 puntos):

Sea cual sea el ejercicio a resolver por parte del examen, cada query de SELECT y cada query de DDS de inserción y actualización que se ejecute a través de Python se debe guardar en una tabla de base de datos mediante SQL. Esta tabla debe llamarse `QueryExecutionHistory` y debe ser creada por el estudiante en el momento de la creación de la base de datos del ejercicio.

Su archivo Jupyter Main debe contener (18 puntos):

Scripts de creación de la base de datos.

Diseño de la base de datos junto con sus componentes Triggers, Procedimientos Almacenados, Funciones, Vistas y otros pertenecientes a la Unidad 2 que permitan resolver el problema.

Codificación del diseño:

Instancia de conexión a la base de datos.

Ejecuciones DDS.

Llamado de cada método del ejercicio.

Su librería de Python debe contener un método paramétrico para responder cada uno de los siguientes casos (60 puntos):

- El administrador quiere saber cuáles son los usuarios que han recibido más cursos en toda la plataforma. Deben aparecer máximo X cantidad de usuarios y solo pueden aparecer usuarios que han realizado un mínimo Z de cursos recibidos.
- El administrador quiere saber cuáles son los usuarios que han matriculado más cursos en toda la plataforma, sin haberlos finalizado. Deben aparecer máximo X cantidad de usuarios y solo pueden aparecer usuarios que han realizado un mínimo Z de matrículas.

- El administrador quiere saber cuáles son los usuarios que han visualizado más cursos en toda la plataforma, sin haberlos matriculado. Deben aparecer máximo X cantidad de usuarios y solo pueden aparecer usuarios que han realizado un mínimo Z de visualizaciones.
- El administrador quiere saber cuáles son los docentes que han recibido una calificación negativa en al menos un curso, para aquellos estudiantes que ya terminaron el curso. Por defecto, cuando un estudiante se matricula en un curso, dicho registro debe manejarse como un 0% de avance en el mismo.
- rear un curso.
- Registrar a un estudiante en el curso.
- Entregar una calificación a un curso.
- Actualizar la información de un curso.
- Borrar la calificación entregada al curso.
- Entregar una calificación al estudiante para un curso.
- Listar todas las instrucciones DDS/DMS del histórico de transacciones.