

Herramientas de Bases de Datos (17MBID)

Actividad 1 – *Primera Convocatoria*



Universidad
Internacional
de Valencia

María Ángeles Rodríguez Sánchez
Curso 2022 – 2023
Edición Octubre 2022

De:
 Planeta Formación y Universidades

Instrucciones de entrega

- **Se entrega un ZIP en la plataforma de la VIU** (en la asignatura 17MBID > Actividades > Portafolio > Actividad 1) que contenga:
 - Notebook (fichero actividad1.ipynb) con el código Python y la documentación en Markdown.
 - Notebook exportado a pdf (fichero actividad1.pdf).

- **Sino se entrega un ZIP, no se corrige la actividad.** Es decir, si se entrega un RAR por ejemplo, no se corrige la actividad.
- **Sino se entregan todos los ficheros requeridos en el ZIP** (notebook ipynb y notebook pdf), no se corrige la actividad. Es decir, si falta un fichero de los mencionados no se corrige la actividad.

- La actividad 1 de la asignatura solo se puede subir una vez a la plataforma de la VIU. **No se permiten resubidas/reentregas.**
- **No se aceptan entregas de las actividades por correo electrónico.**
- **La fecha límite de entrega de la Actividad 1 se puede consultar en Anexo de la Guía Didáctica. No se admiten entregas de las actividades fuera de plazo.**

- **La Actividad 1 en segunda convocatoria lleva una penalización del 10%. Tal como indica la guía didáctica: “se puede recuperar el portafolio en una segunda convocatoria, con una penalización del 10%.”.**

Objetivos

- Diseñar una base de datos relacional (desde Notebook Jupyter).
- Desplegar una base de datos relacional (en local o en la nube, elegir una opción, la que se prefiera) con librerías de Python (desde Notebook Jupyter).
- Insertar datos en una bbdd relacional (en local o en la nube, la que se haya elegido en el despliegue) con librerías de Python (desde Notebook Jupyter).
- Realizar consultas, actualizaciones y borrados a bbdd relacional con librerías de Python (desde Notebook Jupyter).

Herramientas

- SGBD relacional en local: PostgreSQL, MySQL o Microsoft SQL Server (si se desea otra consultarlo con la profesora, SQLite no permitido).
- SGBD relacional en la nube, por ejemplo: ElephantSQL (para PostgreSQL), enlace: <https://www.elephantsql.com/>.
- Lenguaje de programación: Python versión 3, enlace: <https://www.python.org/downloads/>.
- Librerías de Python para BBDD relacional, por ejemplo: psycopg2 (para PostgreSQL), enlace: <https://www.psycopg.org/>.
- Editor de Notebooks: Jupyter (JupyterLab o Jupyter Notebook, el que se prefiera), enlace: <https://jupyter.org/install>.
- Markdown (documentación desde Notebooks): <https://daringfireball.net/projects/markdown/syntax>.

Base de Datos

- Se desea diseñar y crear una **base de datos de estadísticas del deporte béisbol**.
- Se proporcionan el nombre de las tablas y el nombre de los atributos de cada tabla.
- Se proporcionan ficheros en formato .csv con los datos de la base de datos.
- El tipo de los atributos de las tablas deben ser averiguados por el alumno. Los ficheros .csv y la descripción de los datos pueden ser de ayuda.

Documentación

El notebook debe contener los siguientes apartados:

Portada,

Índice,

1. Diseño de la base de datos,

2. Inserción de los datos a la base de datos,

3. Consultas a base de datos,

4. Actualizaciones a base de datos,

5. Borrados a base de datos y

6. Conclusiones.

Cada apartado se explica con detalle a continuación.

Detalle Documentación

Jupyter - Celda Markdown, ejemplo:

jupyter Ejemplo Last Checkpoint: hace 7 horas (unsaved changes) Python 3 (ipykernel) Logout

File Edit View Insert Cell Kernel Help Not Trusted

Run Markdown

Ejemplo celda Markdown

In []:

jupyter Ejemplo Last Checkpoint: hace 7 horas (unsaved changes) Python 3 (ipykernel) Logout

File Edit View Insert Cell Kernel Help Not Trusted

Run Markdown

Ejemplo celda Markdown

In []:

Jupyter - Celda **Código**, ejemplo:

The screenshot displays the Jupyter Notebook interface. At the top, the Jupyter logo is followed by the text "Ejemplo" and "Last Checkpoint: hace 7 horas (unsaved changes)". On the right, there is a Python logo and a "Logout" button. Below this is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", and "Help". To the right of the menu bar are "Trusted" and "Python 3 (ipykernel)" labels. A toolbar contains icons for saving, adding, cutting, copying, pasting, undo, redo, and running. The "Run" button, represented by a play icon, is highlighted with a red box. Next to it is a dropdown menu currently set to "Code", which is also highlighted with a red box. Below the toolbar is a code cell containing the text "In [1]: print(2+2)". The output of the cell, "4", is displayed below the code. The entire interface is set against a light gray background.

Apartado **Portada**

Hay que indicar por lo menos el siguiente punto, en el notebook como celda Markdown:

- Nombre y apellidos del alumno/a

Apartado **Índice**

Hay que realizar el índice de la documentación, en el notebook como celda Markdown:

1. Diseño de la base de datos
[...]

Apartado 1. Diseño de la base de datos

1.1. Diseño conceptual

Hay que indicar por lo menos los siguientes puntos:

- Crear el diagrama entidad/relación (con programa draw.io, o erdplus, ...). Exportar a imagen el diagrama e insertar como imagen en el notebook.
- Explicar el diagrama entidad/relación en el notebook como celdas Markdown.

Apartado 1. Diseño de la base de datos

1.2. Diseño lógico

Hay que indicar por lo menos los siguientes puntos:

- Convertir el diagrama entidad/relación en esquema lógico. No es obligatorio normalizar la base de datos. El esquema lógico es una lista de entidades(atributos), con sus claves primarias y claves ajenas resultantes, y dominios (si es el caso). No es obligatorio indicar las reglas de las claves ajenas (nulos, borrado, modificación). Poner el esquema lógico en el notebook como celdas Markdown.
- Explicar el esquema lógico en el notebook como celdas Markdown.

Apartado 1. Diseño de la base de datos

1.2. Diseño lógico

AYUDA para el esquema lógico:

TABLA1 (atributo1, atributo2, atributo3, atributo4, ...)

PK: {atributo1}

FK: {TABLA1.atributo4 es clave ajena a TABLA2.atributo} o {
TABLA1.atributo4 -> TABLA2.atributo}

(No obligatorio) ¿Acepta nulos?:

(No obligatorio) Borrado:

(No obligatorio) Modificación:

Apartado 1. Diseño de la base de datos

1.3. Diseño físico

Hay que indicar por lo menos los siguientes puntos:

- Convertir el esquema lógico en esquema físico. No es obligatorio realizar vistas. El esquema físico es obtener todos los create table, create domain, etc. Poner el esquema físico en el notebook como celdas de código.
- Ejecutar el esquema físico en local o en la nube, ejecutar la celda código del notebook.

Apartado 2. Inserción de los datos a la base de datos

- Escoger dos registros de cada tabla y crear sentencias INSERT en SQL (código de inserción de datos). Los inserts ponerlos como celda código en el notebook y ejecutarlo en local o en la nube (ejecutar las celdas código del notebook).
- Eliminar de los ficheros los registros insertados anteriormente desde un editor de ficheros (sino os dará error de duplicado al insertar). Luego en el notebook insertar todos los datos restantes de los ficheros proporcionados en la actividad en la base de datos local o en la nube. No puede perderse ningún dato, tienen que insertarse todos los datos dados. Poner el código de insertar los datos de los ficheros en la bbdd en el notebook como celdas de código y después ejecutar las celdas código del notebook.

Apartado 3. Consultas a base de datos

3.1 Consultas simples/básicas – *obligatorio* -

Se deben realizar mínimo dos consultas simples. Para cada consulta, hay que indicar por lo menos los siguientes puntos:

- Descripción de la consulta, ejemplo: Mostrar el nombre de los jugadores. Celda Markdown en notebook.
- Código consulta bbdd. Celda Código en notebook y ejecutarla.

Apartado 3. Consultas a base de datos

3.2 Consultas multitable (con JOIN) – *obligatorio* -

Se deben realizar mínimo dos consultas con operador JOIN. Para cada consulta, hay que indicar por lo menos los siguientes puntos:

- Descripción de la consulta, ejemplo: Mostrar el nombre de los jugadores que han jugado más de 5 partidos. Celda Markdown en notebook.
- Código consulta bbdd. Celda Código en notebook y ejecutarla.

Apartado 3. Consultas a base de datos

3.3. Consultas con función de agregación (operaciones sobre conjuntos de filas) – *obligatorio* -

Se deben realizar mínimo dos consultas con operadores de agregación: COUNT, MAX, MIN, SUM, AVG, Y también se puede usar GROUP BY y HAVING.

Para cada consulta, hay que indicar por lo menos los siguientes puntos:

- Explicar consulta. Celda Markdown en notebook.
- Código consulta bbdd. Celda Código en notebook y ejecutarla.

Apartado 3. Consultas a base de datos

3.4. Subconsultas – *obligatorio* -

Se deben realizar mínimo dos subconsultas. Para cada consulta, hay que indicar por lo menos los siguientes puntos:

- Explicar consulta. Celda Markdown en notebook.
- Código consulta bbdd. Celda Código en notebook y ejecutarla.

Apartado 4. Actualizaciones a base de datos

- Actualizar el valor de un campo de una tabla de la bbdd.
- Cambiar el tipo de datos de un campo de una tabla de la bbdd.

Se debe realizar mínimo una actualización de cada tipo. Hay que indicar por lo menos los siguientes puntos:

- Explicar actualización. Celda Markdown en notebook.
- Código actualización bbdd. Celda Código en notebook y ejecutarla.

Apartado 5. Borrados a base de datos

- Borrar un registro de la bbdd.
- Eliminar un campo de una tabla de la bbdd.
- Borrar una tabla de la bbdd.

Se debe realizar mínimo un borrado de cada tipo. Hay que indicar por lo menos los siguientes puntos:

- Explicar borrado. Celda Markdown en notebook.
- Código borrado bbdd. Celda Código en notebook y ejecutarla.

Apartado 6. Conclusiones

- Indicar las conclusiones de la actividad 1 en celdas de Markdown.



viu

Universidad
Internacional
de Valencia

universidadviu.com

17MBID– Actividad 1

De:
 Planeta Formación y Universidades