



IIC2115 – Programación como Herramienta para la Ingeniería (II/2023)

Ejercicio Formativo 1 Capítulo 6

Aspectos generales

- **Objetivos:** aplicar los contenidos de bases de datos relacionales para modelar entidades y sus relaciones, poblarlas con datos y realizar consultas sobre ellas.
- **Lugar de entrega:** lunes 13 de NOVIEMBRE a las 17:30 hrs. en repositorio privado. NO OLVIDE ADEMÁS RESPONDER EL TICKET DE SALIDA.
- **Formato de entrega:** archivo Python Notebook (**C6E1.ipynb**) con el avance logrado para el ejercicio. El archivo debe estar ubicado en la carpeta **C6**. Utilice múltiples celdas de texto y código para facilitar el trabajo del cuerpo docente.
- **ULTRA IMPORTANTE:** todas las celdas utilizadas deben estar ejecutadas al momento de entregar el ejercicio, de modo que las salidas generadas sean visibles. En caso de no cumplir esto, su entrega no será considerada como validación del ticket de salida.

Introducción

Con el fin practicar los contenidos de bases de datos relacionales, en este ejercicio deberá crear una base de datos con tablas que modelen las entidades presentes en un conjunto de datos que recopila información de partidos de diversas ligas de fútbol europeo, y las relaciones entre estas entidades. Luego, deberá cargar los elementos del conjunto de datos en las tablas creadas, para posteriormente realizar consultas sobre estas tablas.

Descripción de los datos

Considere la base de datos *Football Results*, que recopila información de 12.290 partidos de diversas ligas de fútbol europeo entre los años 2014 y 2020. Estos datos consideran los siguientes indicadores por cada equipo participante en cada partido:

- **league**: liga europea a la que corresponde el partido.
- **season**: año en que comenzó la temporada donde se realizó el partido.
- **date**: fecha en que se jugó el partido.
- **team**: equipo para el cual se indican los datos del partido.
- **h_a**: localía (h) o visita (a) del equipo analizado.
- **result**: resultado del partido para el equipo analizado: triunfo (w), derrota (l), empate (d).
- **pts**: puntos obtenidos en el partido por el equipo analizado.
- **goals_scored**: goles anotados por el equipo analizado.
- **goals_missed**: goles recibidos por el equipo analizado.
- **deep_passes**: pases exitosos realizados por el equipo analizado en los últimos 20 metros de la cancha (pases profundos).
- **deep_passes_allowed**: pases profundos exitosos realizados por el contrincante.
- **ppda**: índice de presión defensiva, calculado a partir de los pases realizados por el equipo contrincante por cada acción defensiva realizada por el equipo analizado (mientras más bajo, más presión).
- **oppda**: índice de presión defensiva del equipo contrincante.

Los datos se encuentran en un archivo en formato *json*, que puede ser abierto y manipulado utilizando la librería **json** de Python. Se recomienda explorar inicialmente el contenido del archivo y familiarizarse con el formato en que está almacenada la información.

Misiones

Misión 1: modelación de entidades

En base a la estructura de los datos almacenados en el archivo, deberá utilizar conceptos de bases de datos relacionales para crear tablas que modelen al menos tres entidades y las relaciones entre ellas, incluyendo las restricciones de integridad correspondientes.

Misión 2: carga de datos

Una vez cumplida la misión anterior, deberá extraer los datos del archivo y cargarlos en las tablas correspondientes, creando y respetando las relaciones existentes en los datos.

Misión 3: consultas

Escriba consultas SQL para responder a las preguntas que se indican a continuación. Considere que cada pregunta debe ser respondida utilizando **una única consulta**.

- Encuentre para cada liga, la cantidad de equipos distintos que han participado en ella.
- Encuentre el nombre del equipo que tiene mejor diferencia entre goles anotados y goles perdidos a lo largo de los años.
- Para cada liga, encuentre la temporada donde hubo mayor diferencia de puntaje entre el primer y último lugar, e indique esta diferencia.