Sleep Disorder Classification

# BigData

## Dataset:

El dataset utilizado es [***Sleep Health and Lifestyle***](https://www.kaggle.com/datasets/uom190346a/sleep-health-and-lifestyle-dataset?resource=download), el cual nos aporta datos relacionados con el sueño y conductas saludables. El tamaño inicial de este es de 374 registros por 13 columnas, las cuales son:

* Person ID: Identificador de cada individuo.
* Gender: El género del individuo,
* Age: La edad del individuo.
* Occupation: La profesión u ocupación.
* Sleep Duration: Número de horas de sueño diarias.
* Quality of Sleep: La calidad del sueño en un rango entre 1 y 10. Es subjetivo.
* Physical Activity Level: Minutos diarios en donde el individuo hace ejercicio físico.
* Stress Level: Un rango subjetivo del nivel de estrés del 1 al 10.
* BMI Category: El índice de masa corporal.
* Blood Pressure: La presión sanguínea del individuo. Sigue el formato “presión sistólica/presión diastólica”.
* Heart Rate: Pulsaciones por minuto del corazón del individuo.
* Daily Steps: Número de pasos por día.
* Sleep Disorder: La presencia o no de desórdenes del sueño.

## MySQL:

Dado que nuestros datos son estructurados, utilizaremos una base de datos SQL para guardarlos. Además, las bases de datos relacionales son ideales para mantener la integridad de los datos al ser capaces de aplicar restricciones y reglas en las tablas.

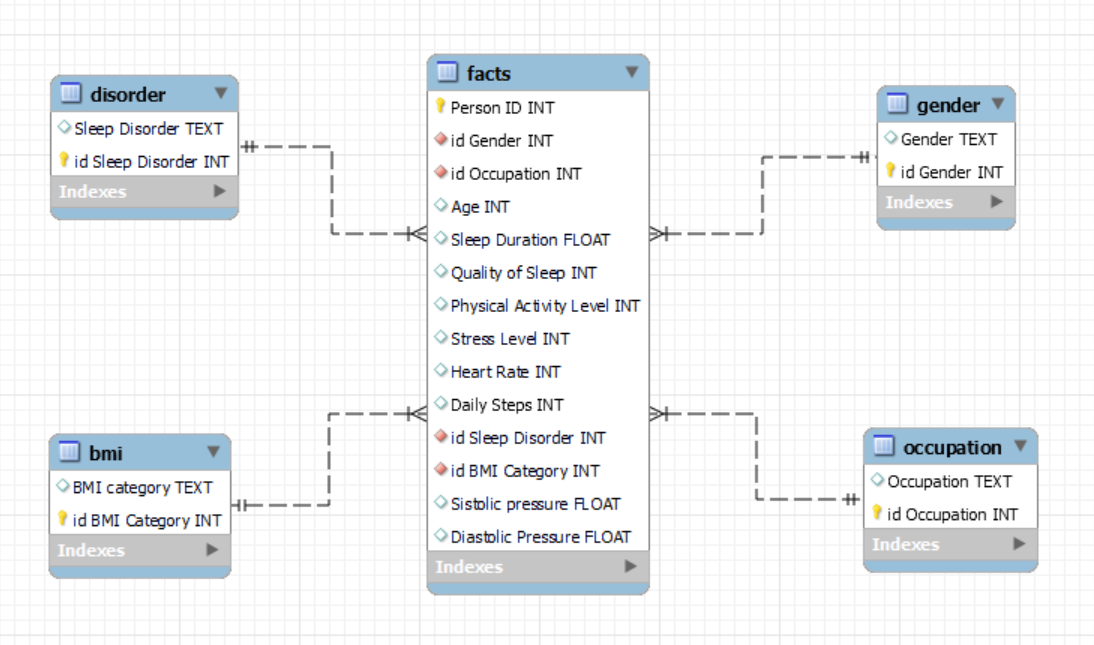
Nuestra base de datos estará ubicada en local y la ingesta de datos se realizará desde un contenedor de Docker creado a partir de la imagen [jupyter/pyspark-notebook](https://hub.docker.com/r/jupyter/pyspark-notebook). La comunicación entre la base de datos y el contenedor se realizará a través de la IP localhost y el puerto 3306.

Nos serviremos de MySQL Installer para realizar las instalaciones necesarias, que son: MySQL Server, MySQL Workbench (Sistema gestor de bases de datos), el driver Connector/J y MySQL Command Line Client. Además, para asignar las claves primarias y foráneas generaremos un archivo SQL que se ejecutará en el gestor de bases de datos. Del mismo modo aplicaremos algunas restricciones.

### Modelo de datos:

Utilizaremos el modelo multidimensional de estrella debido a que el proyecto estará orientado a la generación de informes y el análisis de los datos. Lo óptimo sería utilizar una estructura de estrella debido a que nuestras tablas de dimensiones tendrían demasiadas pocas columnas como para hacer un modelo copo de nieve.

La siguiente imagen es un diagrama entidad-relación del modelo:



Vemos que las cardinalidades de cada relación indican que la tabla de hechos puede tener 0, 1 o varios registros relacionados con un solo registro de una de las tablas de dimensiones (cardinalidad [\* - 1] para todas las relaciones).

#### Entidades:

El modelo presenta las siguientes entidades:

1. Tabla de hechos: En esta tabla se guardan los datos que son cuantitativos. En este caso, tienen tipado float o integer. La tabla es:

* facts.

1. Tabla de dimensiones: En estas tablas guardamos los datos descriptivos, los cuales son de tipo string. Además, tendrán una columna tipo integer que será la clave primaria. Las tablas son:

* gender.
* occupation.
* disorder.
* bmi.

Se puede apreciar dos nuevas columnas: Sistolic pressure y Diastolic pressure. Estas dos columnas proceden de Blood Pressure. Esta columna es tipo string y los datos siguen este formato: ‘presión\_sistólica/presión\_diastólica’. El proceso llevado acabo es separa los valores numéricos por la ‘/’, guardarlos en dos columnas distintas (Sistolic pressure y Diastolic pressure) y parsearlos a float.

## PySpark

Se levantará un contenedor de Docker con la imagen de Jupyter/PySpark-notebook. Será aquí donde tendrán lugar la limpieza e inserción de datos.

Este proceso se reflejará en el código de la siguiente manera:

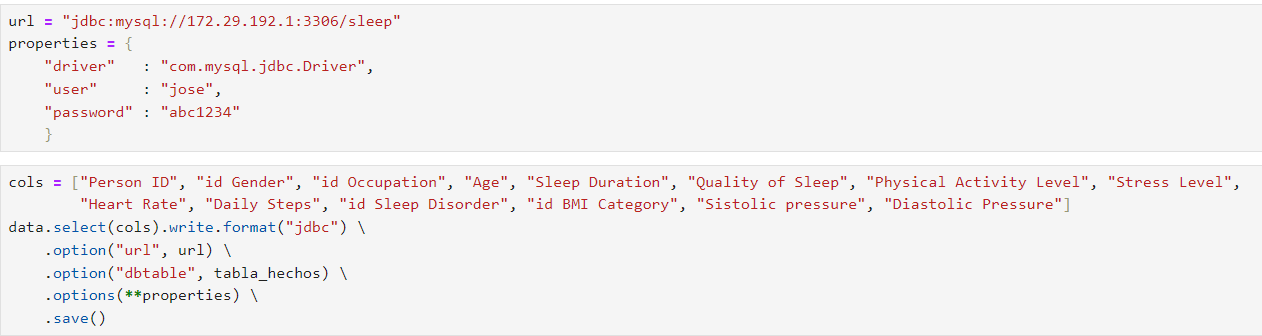
### Limpieza de datos:



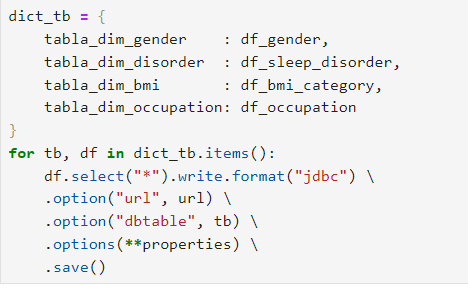
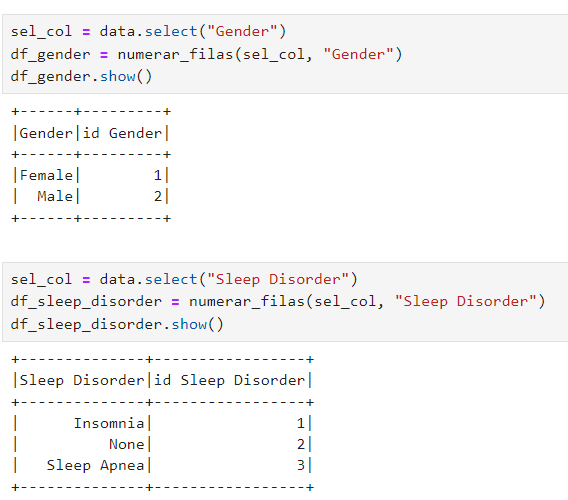
En la imagen anterior se observan tres funciones:

* numerar\_filas: Se encarga de generar las columnas ID de las distintas columnas descriptivas. Estos ID pasarán a ser las claves primarias de las tablas dimensiones.
* check\_duplicates: Esta función comprobará que haya o no duplicados en la columna ID. Si detecta duplicados, los elimina.
* limpiado: Aquí es donde se depuran los datos: se parsea la columna ‘Sleep Duration’ a float, se unifican los valores ‘Normal’ y ‘Normal weight’ de la columna ‘BMI Category’ y se separan los valores de la columna ‘Blood Pressure’, guardandolos en las columnas ‘Sistolic pressure’ y ‘Diastolic pressure’.

### Inserción:



En la imagen se observa cómo procede la inserción de datos en la base de datos llamada ‘sleep’. Se vuelcan en la tabla de hechos los datos cuantitativos guardados en la variable ‘cols’.



En las últimas imágenes vemos como se crean los dataframes que posteriormente se volcarán en las tablas de dimensiones.