

Ejercicio de prueba - Ombúes

Esta consigna busca dejar en claro el tipo de requerimientos que se espera manejar de manera habitual, así como las herramientas más comunes, y también dar la posibilidad de conocer y evaluar las capacidades técnicas y resolutivas de los potenciales miembros del equipo.

Dado que se trata de un ejercicio de prueba, la temática no está directamente relacionada con el objeto del negocio que se va a estar trabajando posteriormente.

Requerimiento

Registrar en una base de datos la información de los ombúes que están en el barrio de la Ciudad de Buenos Aires que más tiene de ellos. Devolver la cantidad total de ombúes y su participación en la ciudad.

Detalle

Generar un script que obtenga el archivo actual del detalle de arbolado de la Ciudad de Buenos Aires. Contar la cantidad ombúes que hay en cada uno de los barrios, y almacenar en la base de datos un registro por cada uno de los registros del barrio correspondiente (con todos sus campos). El script debe devolver por pantalla la cantidad total de ombúes que hay en la Ciudad, y el porcentaje que estos representan del total de árboles en la misma.

Requerimientos adicionales

- Como parte del pre-proceso del script se debe limpiar la base de datos (cada vez que corre el script se debe limpiar previamente la base de datos)
- Utilizar el archivo CSV del sitio del gobierno de la ciudad (<https://data.buenosaires.gob.ar/dataset/arbolado-publico-lineal/>)
- Utilizar una base de datos MongoDB (crear una cuenta gratuita de prueba) (<https://www.mongodb.com/cloud/atlas>)
- Hacer uso del script db_connect.py (detallado abajo), ampliarlo, editarlo, etc.
- Abrir una cuenta de GitHub y guardar los archivos desarrollados en este repositorio. Agregar al usuario 'chelowite' (sin las comillas) como colaborador
- Llevar registro del tiempo requerido para llevar a cabo esta tarea
- Consultar cualquier duda que pueda surgir acerca de la consigna o cuestiones técnicas

Script db_connect.py

```
import datetime
import pymongo

CONN_ATLAS = "atlas"

class DB:
    _instance = None
    __conn = None
    database = None

    def __new__(cls):
        if cls._instance is None:
            cls._instance = super(DB, cls).__new__(cls)
        return cls._instance

    def __init__(self):
        if self.__conn is None:
            conn_type = CONN_ATLAS
            if conn_type == CONN_ATLAS:
                # Conexión con la DB de Atlas
                self.__conn =
pymongo.MongoClient("mongodb+srv://[revisar_datos]")
            else:
                raise ValueError("Connection error")
            self.database = self.__conn["nombre_base_datos"]

# Test section
if __name__ == "__main__":
    db = DB()
    data = db.database["nombre_coleccion"]
    data_query = {'campo': 'valor'}
    data_count = data.find(data_query).count()
    print("Cantidad (", datetime.datetime.now(), "):", data_count)
```