**Examen Practicante - Analítica**

En el Contact Center se administran las interacciones de los clientes y usuarios en redes sociales.

En el archivo Redes\_Sociales encontraras dos pestañas.

Casos: Contiene la cantidad de casos atendidos por red y por tipo durante el mes de julio en redes sociales.

Meta: Contiene la meta diaria de los casos que se deben de atender por red y por tipo en redes sociales.

**NOTA:** para la realización de este proyecto se necesita que se cree un ambiente virtual en una versión de Python 3.9.12.

**Ejercicio 1:**

* Utilizando Python debes de leer la pestaña Casos y convertirlo en un Dataframe con la siguiente estructura y exportarlo a Excel. El Dataframe final debe de llamarse df\_casos\_bd

|  |  |  |  |
| --- | --- | --- | --- |
| Red | Tipo | Fecha | Cant\_Casos |

Nota: No se debe de realizar ninguna agrupación.

Entregable:

* Archivo de Excel con el Dataframe df\_casos\_bd y el código en un archivo .py con el script para la transformación.

**Ejercicio 2.**

* Utilizando Python, hacer una unión del df\_casos\_bd con el Dataframe de la pestaña Meta.
* El resultado final debe de ser otro Daframe con la siguiente estructura. El Dataframe final debe de llamarse df\_casos\_union

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Red | Tipo | Fecha | Cant\_Casos | Meta | % Cumplimiento |

Nota: El % de cumplimiento se calcula con la formula (Cant\_Casos / Meta) y NO se debe de realizar ninguna agrupación.

Entregable:

* Archivo de Excel con el Dataframe df\_casos\_union y el código en un archivo .py con el script para la transformación.

**Ejercicio 3.**

* Utilizando SQL, crear un Query para consultar la información del Dataframe df\_casos\_union y generar una tabla con la siguiente estructura.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Red | Tipo | Año | Mes | Cant\_Casos |

Nota: Se debe de agrupar la información por Red, Tipo, Año y Mes. La cantidad de Casos se debe de sumar.

Entregable:

* Archivo de Excel con el resultado de la consulta y el código en un archivo .txt con el Query.

**Ejercicio 4.**

Se solicita que tome el proyecto que ya ha desarrollado previamente y lo suba a un nuevo repositorio en GitHub. A continuación, enumere los comandos de Git necesarios para crear el repositorio, enlazarlo con GitHub y subir los archivos. Luego, explique brevemente cada paso realizado.

Git init

Este comando inicializa un repositorio local en la carpeta actual

Git add .

Este comando añade todos los archivos y carpetas de la carpeta actual al área de preparación, preparándolos para hacer el commit.

Git commit -m “Ejercicios 1, 2 y 3”

Este comando realiza un commit, es decir, guarda el estado actual del proyecto en el repositorio local con un mensaje.

git remote add origin git@github.com:Pukks31/Prueba\_tecnica\_practicas.git

Este comando establece la conexión entre el repositorio local y el remoto.

git push -u origin master

Este comando sube los cambios desde el repositorio local al repositorio remoto en GitHub.Indicaciones:

1. Cree un repositorio en GitHub llamado Prueba\_tecnica\_practicas.
2. Utilice comandos de Git para añadir el proyecto al repositorio y subirlo (explicar brevemente los comandos utilizados)
3. Incluya un archivo README.md en el repositorio, explicando brevemente en qué consiste el proyecto.

**Ejercicio 5.**

* Crear un Dashboard en PowerBI con la información de las pestañas Caso y Meta. NO se deben de cargar los Dataframes creados, solo las pestañas como tablas independientes. Estas tablas se deben de relacionar en PowerBI (Power Query). El Dashboard debe responder preguntas de negocio como ¿cual es el cumplimiento de casos? ¿Como ha sido el comportamiento de los casos por día? ¿Por qué red y tipo de red se atienden mas casos? Etc.

Entregable:

* Archivo de PowerBI con el Dashboard.