



Python Automation Developer for Data Science, LATAM

PayU Colombia

Technical Test

El dataset adjunto contiene datos de transacciones con tarjeta de crédito hechas entre el 21 de junio de 2020 y el 31 de diciembre de 2020. Las columnas incluidas son las siguientes:

Nombre de la Variable	Descripción
trans_date_trans_time	Fecha de la transacción
cc_num	Tarjeta de crédito
merchant	Nombre del comercio
category	Categoría del comercio
amt	Monto de la transacción
currency	Moneda de la transacción
first	Nombre del pagador
last	Apellido del pagador
gender	Género
street	Dirección del pagador
city	Ciudad
state	Estado
zip	Código ZIP
lat	Latitud
long	Longitud
city_pop	Población de la ciudad
job	Ocupación
dob	Fecha de Nacimiento
trans_num	Id de la transacción
unix_time	Tiempo Unix
merch_lat	Latitud del comercio
merch_long	Longitud del comercio
is_fraud	Variable binaria. 1 si la transacción es fraudulenta, 0 si es genuina

1. Cree una base de datos alojada localmente en un motor de bases de datos como MySQL o SQL Server con la información del dataset.
2. En Python extraiga de la base de datos las transacciones de los comercios de las categorías (columna 'category') que tienen menos de 15.000 registros.

Por ejemplo, si en la base de datos las categorías 'Retail', 'Finance' y 'Airlines' tienen 20.000, 10.000 y 14.000 registros, usted debe incluir las transacciones de los comercios de las categorías de 'Finance' y 'Airlines'.

A partir de este punto, por favor utilice la librería pandas de Python:

3. Conéctese a una API gratuita para extraer la tasa de cambio vigente de USD a COP. Por ejemplo: <https://github.com/Formicka/exchangerate.host>. Convierta en una nueva columna llamada 'amt_cop' la columna 'amt' a pesos colombianos dependiendo de la divisa indicada en la columna 'currency'.
4. Calcule una nueva columna con los 6 primeros números de la columna 'cc_num' y llámela 'bin'.
5. Calcule en una nueva columna en la misma tabla la cuenta de números de tarjeta únicos por 'bin' y por 'merchant' en toda la base.

Bin	Merchant	Tarjeta	Cálculo
411111	213	41111185926327	2
411111	258	41111185926327	1
411111	213	41111198724528	2
589625	258	589625987458	1

6. De manera similar, en otra columna, calcule cuántos códigos ZIP se registraron por cada ciudad.
7. En columnas diferentes, calcule el número de transacciones que cada pagador (nombre y apellido) hizo en la misma hora y en el mismo día de la fecha de cada transacción.
8. Asigne el valor 1 en la columna 'is_fraud' cuando los valores calculados en los tres puntos anteriores se ubiquen por encima del promedio. En caso contrario, deje los valores que ya están en esa variable.
9. En una nueva tabla, calcule para cada comercio ('merchant'), el número de transacciones, la suma del monto de todas las transacciones en pesos colombianos, el número de fraudes, la suma del monto con fraude, el porcentaje de fraude por número de transacciones y por monto, el número de tarjetas distintas ('cc_num').
10. Cree una tabla en la base de datos con el resultado del punto anterior.
11. (Bono) Cree un Dashboard utilizando la tecnología web de su preferencia que muestre los datos calculados en el punto 9 para el top 10 de merchants con más % de fraude en monto y que permita visualizar una gráfica del número de transacciones para cada uno a través del tiempo.
12. Guarde todo el código desarrollado en un repositorio de GitHub y envíe el enlace a más tardar el miércoles 27 de septiembre a medianoche a los correos daniela.sanabria@payu.com y camilo.navas@payu.com. Para cualquier duda podrá comunicarse a cualquiera de esos correos.

Todo lo desarrollado anteriormente tendrá que ser expuesto por el participante. Si necesita una extensión al plazo, por favor comunicarlo con anterioridad.

https://1drv.ms/u/s!AiEnR80f_m1AgasdaMWK--2h7B-IXA?e=D0hpAU

Si lo considera necesario, puede hacer el ejercicio con una submuestra de los datos.

