

Prueba Técnica #00010

Autores

ADC Gestión de la Operación

Centro de Excelencia de Operación
Vicepresidencia de Servicios para Clientes y Empleados



Historia del Documento

Fecha (dd/mm/aaaa)	Versión	Descripción/Cambio realizado	Autor y/o Participantes
25/07/2023	1.0	Creación Documento	Andrés Felipe Callejas Jaramillo

Pública



Contenido

- 1. Introducción 4
- 2. Preguntas de entendimiento de los datos (10%) 5
- 3. Evaluación Conocimiento Técnico..... 5
 - 3.1. SQL parte 1 (20%) 5
 - 3.2 SQL parte 2 (20%) 6
 - 3.3. Python parte 1 (20%)..... 7
 - 3.4. Python parte 2 (20%)..... 8
- 4. NOTA (10% Control de Versiones) : 8

Pública

1. Introducción

El propósito de esta prueba es medir tus capacidades para manipular datos mediante diferentes lenguajes. Se recomienda que no le dediques más de 15 horas, incluyendo el tiempo para documentar lo desarrollado.

Se adjuntan 2 archivos XLSX con su respectiva metadata para desarrollar la prueba técnica. Estos archivos son:

- **Obligaciones_clientes.xlsx:**

Este conjunto de datos parece contener información sobre las obligaciones de los clientes con un banco. Algunas de las columnas son:

- **radicado:** identificador único para cada registro.
- **num_documento:** identificador único para cada cliente.
- **cod_segm_tasa, cod_subsegm_tasa:** códigos relacionados con la tasa de interés aplicada.
- **cal_interna_tasa:** calificación interna de la tasa de interés.
- **id_producto, tipo_id_producto:** información relacionada con el producto financiero adquirido por el cliente.
- **valor_inicial:** monto del préstamo inicial.
- **fecha_desembolso:** la fecha en que se desembolsó el préstamo.
- **plazo:** el plazo del préstamo.
- **cod_periodicidad, periodicidad:** información relacionada con la periodicidad del pago del préstamo.
- **saldo_deuda:** el saldo de la deuda del cliente.
- **modalidad:** si el préstamo es nuevo o si se está actualizando.
- **tipo_plazo:** si el plazo es a corto plazo (CP) o largo plazo (LP).

- **tasas_productos.xlsx:**

Este conjunto de datos parece contener información sobre diferentes productos financieros y las tasas de interés correspondientes. Algunas de las columnas son:

- **cod_segmento, segmento:** códigos y descripciones de diferentes segmentos de clientes.
- **cod_subsegmento:** código relacionado con los subsegmentos dentro de los segmentos de clientes.
- **calificacion_riesgos:** calificación del riesgo asociado con el producto financiero.
- **tasa_cartera, tasa_operacion_especifica, tasa_hipotecario, tasa_leasing, tasa_sufi, tasa_factoring, tasa_tarjeta:** parecen ser tasas de interés para diferentes tipos de productos financieros.

2. Preguntas de entendimiento de los datos (10%)

Realice un análisis exploratorio de los datos de los archivos “Obligaciones_clientes.xlsx” y “tasas_productos.xlsx”, responda las siguientes preguntas que se encuentran en el siguiente enlace

<https://forms.office.com/r/tPzUq2XwSb>

3. Evaluación Conocimiento Técnico

3.1. SQL parte 1 (20%)

Esta primera parte se debe desarrollar con SQL y para esto puede hacer uso de cualquier motor de base de datos.

1. Se requiere tomar las obligaciones de cada cliente y agregar la tasa correspondiente al producto asignado
2. Se debe convertir la tasa a una tasa efectiva, para ello debemos aplicar la siguiente formula:

$$te = \frac{\left(\left((1 + t)^{\frac{1}{n}} - 1 \right) * n \right)}{n}$$

- a. te : Tasa efectiva
 - b. t: Tasa
 - c. n: n=12/periodicidad
 - d. nota: Si la periodicidad es Mensual el valor de esta es 1, si es bimestral el valor es 2, si es semestral es 6 y así sucesivamente.
3. Tomar la tasa efectiva, multiplicarla por el valor_inicial y dejar este resultado como valor_final, el resultado de esta tabla debe quedar almacenado; ya que este nos servirá como un insumo de la parte 2.

4. Se necesita sumar el valor_final de todas las obligaciones por cliente y dejar únicamente las que tenga un valor superior a ... el resultado de esta tabla debe quedar almacenado; ya que este nos servirá como un insumo de la parte 3.2.

3.2 SQL parte 2 (20%)

Dadas las siguientes tablas, con las relaciones especificadas.

CLIENTE		
PK	Cedula	Alfanumérico
	Nombre	Alfanumérico
	Region	Alfanumérico
	Edad	Numeric

CUENTAS			
PK	Num_cuenta	Alfanumérico	
FK	Cedula_cliente	Alfanumérico	
	Estado	Numeric	0: Inactiva 1: Activa
	Saldo	Numeric	
	Fecha_apertura	Numeric	Formato: aaaammdd

CLAVE_DINAMICA		
FK	cedula_cliente	alfanumérico
	correo	alfanumérico
	celular	alfanumérico

Tenga en cuenta lo siguiente:

Relación uno a muchos entre la tabla CLIENTE y CUENTAS.

Relación uno a uno entre la tabla CLIENTE y la tabla CLAVE DINAMICA.

Realizar las siguientes consultas en SQL que permitan hacer lo siguiente:

- a) (4%) Seleccionar todos los clientes.
- b) (4%) Seleccionar los clientes que pertenezcan a la región “Centro”.
- c) (4%) Seleccionar los clientes que tengan más de 3 cuentas abiertas en estado Activo.
- d) (4%) Seleccionar solamente el nombre de aquellos clientes que tienen clave dinámica.
- e) (8%) Seleccionar los clientes que no tienen clave dinámica.
- f) (8%) Mostrar el saldo total de todas las cuentas agrupado por la región del cliente.
- g) (8%) Seleccionar el saldo total de las cuentas activas, abiertas en el mes de mayo de 2018, cuyos clientes tengan clave dinámica.

3.3. Python parte 1 (20%)

Se requiere desarrollar de forma visual en Python versión 3.9.2, para exponer información. Puede usar Jupyter o Google colab para dar respuesta a los siguientes puntos tenga presente los datos de los archivos “Obligaciones_clientes.xlsx” y “tasas_productos.xlsx”.

- a) De la información almacenada del punto 3.1 se requiere mostrar la información de productos, tasas efectivas y valor final del cliente que sea consultado (Son varios registros dependiendo del cliente consultado).
- b) De la información almacenada del punto 3.1 se requiere mostrar el valor total del cliente que sea consultado (Un único registro por cliente).
- c) Cargue ambos conjuntos de datos en Python utilizando pandas.
- d) Fusione ambos conjuntos de datos en función de las columnas relevantes.
- e) Limpie y preprocese los datos si es necesario (por ejemplo, maneje los valores perdidos, convierta los tipos de datos si es necesario).
- f) Cree una función para calcular el monto total de la deuda para cada segmento de cliente.
- g) Realice un análisis de agrupación (por ejemplo, k-means) en los clientes basado en su deuda y las tasas de interés de sus productos financieros.
- h) Construya un modelo de regresión para predecir el saldo de la deuda del cliente en función de las otras características.
- i) Evalúe el rendimiento del modelo utilizando métricas de evaluación adecuadas.
- j) Proporcione interpretaciones de los coeficientes del modelo de regresión.
- k) Realice una validación cruzada del modelo para asegurarse de su robustez.
- l) Identifique y explique cualquier posible sesgo o limitación en su análisis.

3.4. Python parte 2 (20%)

Los siguientes problemas deben ser realizados en Python puro (sin utilizar librerías como numpy o pandas):

- (5%) Realizar un programa que lea una secuencia de números por teclado. La secuencia finaliza al digitar el valor de -1. Para la secuencia leída el algoritmo debe calcular el promedio de los valores mayores al promedio de la secuencia.
- (5%) Realizar una función, que reciba como parámetros una lista (vector o array) ordenada y un número x . La función debe imprimir por pantalla el par de números, cuya suma es más cercana al valor de x .

Ejemplo:

Entrada: `list[] = (10, 22, 28, 29, 30, 40), x = 54`

Salida: `(22, 30)`

- (10%) Realizar una función, que reciba una matriz de dimensión $m \times n$ y un número p . Se debe rotar p veces e imprimir la matriz resultante. La rotación debe ser en sentido antihorario si $p < 0$ y en sentido horario cuando $p > 0$.

Condiciones:

- $2 \leq m, n \leq 100$
- $1 \leq p \leq 10^7$
- $\min(m, n) \% 2 = 0$

Ejemplo:

- Entrada:** `matrix = [[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12], [13, 14, 15, 16]], p = 2`

Salida:

```

3 4 8 12
2 11 10 16
1 7 6 15
5 9 13 14

```

4. Control de Versiones (10%) :

Todo el código debe estar alojado y respaldado en el repositorio de Gitlab asignado.

Evaluación Para la evaluación se tendrán varios criterios, mencionados a continuación:

- Adjuntar código.
- Documentación del código y código limpio