

Prueba de Conocimiento SQL, Java y Microservicios

Contexto

Para el negocio es necesario manejar diferentes precios para nuestros productos, esto depende de algunos factores como la línea de negocio, ubicación, oferta y temporada. Por lo que muchas veces es necesario tener reglas definidas que permitan calcular de manera masiva los nuevos precios. Estos procesos deben de ser automáticos, rápidos y eficientes con el fin de ayudar a las diferentes áreas para automatizar actividades que antes se hacen de manera manual.

Objetivo

Crear un API REST que le permita al usuario consultar la información de un producto guardado en una base de datos, además se necesita tener procesos de Base de datos que calculen valores para cada producto, guardarlos en la base de datos y presentarlos en el response del API.

Requisitos

1. Crear Script SQL para la creación de tabla "Product". La cual cuenta con los siguientes campos
 - SKU: código del producto,
 - Description: nombre del producto,
 - Price: Precio de Venta,
 - Increment: valor del % de incremento para calcular futuros precios.

Los tipos de datos quedan a discreción del participante en función del excel.

El archivo para cargar es data_productos_prueba.xlsx.

2. Elaborar un programa de Java para lectura de archivo en formato XLSX y carga de data en la Tabla "Product". Para la persistencia de datos se debe usar un ORM.
3. Diseñar una función de Base de Datos que permita calcular el precio de venta con incremento de un producto. La función debe recibir como parámetro el SKU y devolver como parámetro de salida el precio con el incremento, la lógica de la función es la siguiente:
 - a. Se calcula en función del PVP e Incremento del SKU.
 - b. Productos con PVP menor a \$15.00 no aplican el incremento.
 - c. El incremento representa un valor porcentual por encima del precio.
 - d. Si la parte decimal del precio con incremento es menor a \$0.50, se asigna al precio con incremento como el entero menor, caso contrario se asigna el entero mayor y al final se le resta \$0.01.

Ejemplo:

Precio con Incremento: \$19.40
Precio con Incremento final: \$18.99

Precio con Incremento: \$19.60
Precio con Incremento final: \$19.99

4. Crear un procedimiento de base de datos que permita obtener el máximo número de cuotas para el pago en función del precio con incremento. El procedimiento ha de planchar los valores de precio con incremento y máximo de cuotas en una tabla relacionada a "Product" llamada "ProductPrice" con los campos "IncrementedPrice" y "MaxDuesNo", el proceso de base de datos debe seguir la siguiente lógica:
 - a. Debe considerar el precio con incremento
 - b. A partir de \$1000, se permiten 30 cuotas.
 - c. A partir de \$300, se permiten 24 cuotas.
 - d. A partir de \$250, se permiten 18 cuotas
 - e. Para valores inferiores a \$250, se establece el máximo de cuotas (3, 6, 9, 12) siempre y cuando el valor de la cuota sea mayor a \$10 con relación a la cuota seleccionada.
5. Creación de un microservicio para la consulta de productos, debe presentar la información del producto, precio, precio con incremento y máximo de cuotas, debe recibir como parámetro el SKU del producto a consultar, si no se especifica el SKU debe mostrar toda la lista de productos. La consulta ha de permitir paginación en caso de que se muestre todo el listado de productos.
Considerar lo siguiente:
 - a. Se puede hacer por medio de un proyecto SpringBoot o NodeJs, obligatorio el uso de un ORM.
 - b. Opcional: Contenerización del proyecto en Docker y uso de base de datos NoSQL. Ej: MongoDB.
 - c. Opcional: Elaboración del diseño de proyecto Front End para consulta de productos (Angular, React, Node JS)

Tiempo de Desarrollo

4 días/32 horas

Entregables

1. Video Mostrando el funcionamiento de las funciones/procedimientos, así como el microservicio de consulta
2. Repositorio Git Público con:
 - a. Scripts SQL para creación de DB, Tablas, Funciones y procedimientos.
 - b. Código Fuente de programa java para carga de registros en BD.
 - c. Código Fuente proyecto servicio de consulta de producto.
 - d. Usar un repositorio git público para la publicar el código fuente.