

Como desarrolle los ejercicios de la prueba técnica.

Punto 1A

Primero debía seleccionar la columna nombres de la tabla empleados como solo me piden el nombre y no se especifica que si hay varios empleados con el mismo nombre deben de diferenciarse, entonces decido usar DISTINCT, para que no me aparezcan nombres repetidos, luego renombro la tabla employees como e, luego relaciono los empleados con las ordenes que han atendido y luego relaciono la tabla ordenes con su respectivo cliente, y para terminar es agregar el filtro de la fecha para que los nombres de los empleados obtenidos sean los que nacieron en 1960 o 1963.

Punto 1B

Se que tengo que agrupar los productos por el país de origen y debo calcular el total del precio pagado por esos productos, entonces debo obtener el precio total pagado por los productos de cada país, esto lo hago relacionando la información que hay en la tabla OrderDetails y Products, luego debo obtener de que Supplier viene el producto entonces relaciono ambas tablas, agrupo por país y calculo el total de cada producto multiplicando las unidades compradas, por el precio.

Punto 1C

Se que debo obtener la empresa de transporte, la cantidad transportada y el nombre del producto, entonces debo relacionar la tabla Shippers con la tabla Orders, ahora necesito los detalles de cada orden, entonces relaciono esta tabla con OrderDetails, mediante el orderId, para saber que productos se transportan en estas ordenes debo relacionarme con la tabla Products, luego de esto es donde aplico el filtro, para solo obtener los que me interesan "Tofu" y "NuNuCa Nuß-Nougat-Creme", finalmente obtengo el nombre de la empresa transportista, el nombre del producto y las unidades por producto.

Punto 2

Ya que debo desarrollar un API al cual se le pase x metros y devuelva este valor en kilómetros, me parece que la opción de usar Sprint Boot así sea más complejo para desarrollar esta funcionalidad que relativamente es sencilla, da muestra de que conozco Sprint Boot y soy capaz de usarlo, por eso me decanto por este framework, la construcción es relativamente fácil, debo recibir un dato y cerciorarme que sea un número, independiente de que sea un numero entero o decimal, luego debo hacer la conversión y retornar el dato, para este servicio decido también devolver el dato de entrada guardo los valores en un hash table y los devuelvo en un entityResponse(json).

Punto 3

Lo primero que debo hacer es abrir el archivo de Excel luego darle al usuario la entrada para que copie que marque auto desea buscar, luego de que el usuario ingreso la marca verifico que la marca exista, esto para saber si continuo con el flujo o le aviso al usuario de que la marca no existe, si la marca existe continuo con el proceso, obtengo los registros del Excel de sus diferentes hojas y con esta información creo el dataframe que me piden y la muestro, acto seguido, creo en memoria un archivo de Excel, guardo el dataframe en este archivo y le agrego una línea que dice marca, con su marca respectiva, luego de este proceso guardo el archivo en el directorio donde se este ejecutando el script de Python, imprimo un mensaje existo si todo ocurrió sin problemas y se termina el proceso del script.