1)

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title></title>

</head>

<body>

<?php

$data = json\_decode( file\_get\_contents('https://api.mercadolibre.com/sites/MLA/search?seller\_id=179571326'), true );

$json\_string = json\_encode($data);

$file = 'sellers.json';

file\_put\_contents($file, $json\_string);

?>

</body>

</html>

El Lenguaje que se utilice para este ejercicio fue PHP 7 , el procedimiento que tome fue probar la API en la página de Postman para verificar que funciona y luego al ver que me devolvía genere el script por medio del servicio Xampp levantando el servicio de Http de apache , una vez generado eso , realice el siguiente script:

$data = json\_decode( file\_get\_contents('https://api.mercadolibre.com/sites/MLA/search?seller\_id=179571326'), true );

Api a executar dentro de la accion de PHP

$file = 'sellers.json';

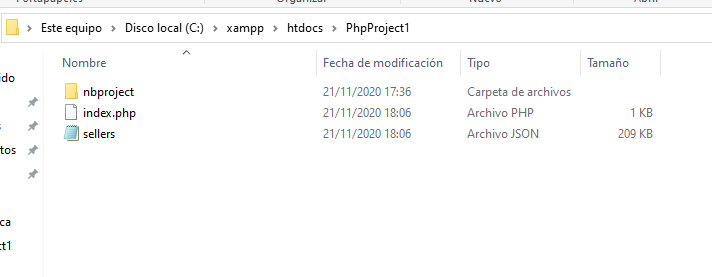
Variable en donde se guarda el archive de Json

file\_put\_contents($file, $json\_string);

Contenidos que se deben almacenar.

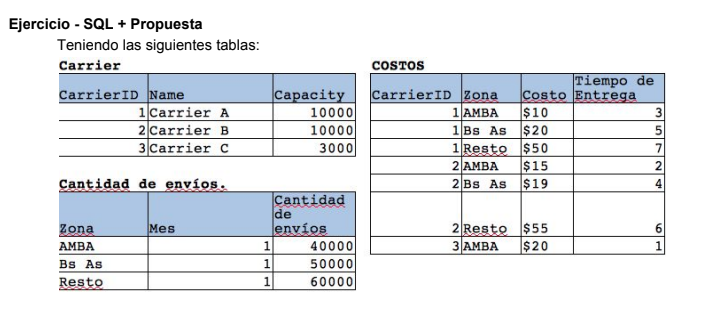
También adjunto una captura dentro de la explicación en detallando en donde se guarda de forma automática el archivo (C:\xampp\htdocs\PhpProject1)

Herramientas utilizadas para la task mencionada: Netbeans , Xampp.



2)

El ejercicio de las queryes de base de datos quedo un poco confuso de entender ya que las tablas que mencionan para desarrollar no se entienden:



¿La capacidad que poseen se distribuye...?, ya que directamente no se podrían realizar la cantidad de envíos esperada.

¿También como ubico las zonas, se manejan por un tipo de id...?, ya que no se mencionan no se podría generar un desarrollo a partir de ahí

Se generó esta primera lógica de las quieres ya que se debería realizar para este caso un Stored Procedure con todos los datos cargados ya que no es fijo, sino que van a variar dependiendo de la zona.

3)

import me.\*;

def upsPullTrkService = ctx.getBean('upsPullTrkService')

def s = Shipment.get(27528954729)

def tn = s.trackingNumber

def trackingData = upsPullTrkService.getTrkEvents([tn])

trackingData.each { td ->

println "------------------------------------------------"

println "${td.sucursal} - ${td.eventDate} - ${td.description}"

}

"Done"

* El funcionamiento principal del script es un sistema de trackeo de un solo paquete, la primera línea hace colocar el sistema de forma activa, ya que luego por parte del “get”, lo que hace es buscar un solo número de paquete en específico, nuevamente se imprime el output con las sucursal, fecha y descripción del paquete.
* Las líneas como output se imprimen una sola ya que solo busca un paquete.

4)

users\_id=(71665538 66146765 132961968 15096445 172753273 54152646) los id de usuarios en un vector

for users\_id in ${users\_id[\*]} //recorre el vector

do

curl=$(curl -s " api.mercadolibre.com/users/$users\_id/shipping\_preferences " | jq -c

'.services') //llama a la api

echo "$users\_id: $curl" ///imprime de por id de usuario el valor obtenido en la api

done

-se buscan imprimir 6 lineas como output ya que son 6 id de usuarios