

# Documento de instalación del challenge de MercadoLibre

**Por: Giovanni Martínez**

Documento con lo necesario para ejecutar el challenge de forma exitosa y fácil:

## Prerrequisitos:

- git cliente o clonar el proyecto desde un navegador.
- Tener Docker instalado de la página oficial: <https://docs.docker.com/get-docker/>, instalar la última versión de Docker en su sistema operativo.

## Instalación:

Repositorio con el challenge se encuentra en el siguiente link de github:

<https://github.com/egmartin1810/ddd-python-rest-mysql.git>

1. Clonar el repositorio con el proyecto, dentro del directorio deseado y entrar al directorio descargado **ddd-python-rest-mysql**, y revisar que se tengan los siguientes directorios y **docker-compose.yml**:

comando: `git clone https://github.com/egmartin1810/ddd-python-rest-mysql.git`

comando: `cd ddd-python-rest-mysql`

```
C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\v2>
C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\v2>
C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\v2>
C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\v2>
C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\v2>git clone https://github.com/egmartin1810/ddd-python-rest-mysql.git
Cloning into 'ddd-python-rest-mysql'...
remote: Enumerating objects: 110, done.
remote: Counting objects: 100% (110/110), done.
remote: Compressing objects: 100% (83/83), done.
Receiving object0 (delta 29), reused 98 (delta 21), pack-reused 0 eceiving objects: 52% (58/110)
Receiving objects: 100% (110/110), 37.57 KiB | 112.00 KiB/s, done.
Resolving deltas: 100% (29/29), done.
```

```
Directory of C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\ddd-python-rest-mysql

4/02/2021  10:55 PM    <DIR>          .
4/02/2021  10:55 PM    <DIR>          ..
4/02/2021  10:55 PM                1,946 .gitignore
4/02/2021  10:55 PM    <DIR>          db
4/02/2021  10:55 PM                523 docker-compose.yml
4/02/2021  10:55 PM    <DIR>          ml-items-challenge
4/02/2021  10:55 PM                311 README.md
               3 File(s)                2,780 bytes
               4 Dir(s)  52,134,633,472 bytes free

C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\ddd-python-rest-mysql>_
```

2. Crear la imagen y correr los contenedores con docker-compose:

NOTA: la aplicación en flask está corriendo por el puerto: 5000 y la base de datos en 3306, tener en cuenta que si la máquina donde se está ejecutando los contenedores ya tienen esos puertos en uso, debemos cambiarlos en el archivo y en el archivo ./ml-items-challenge/config.py cambiar el puerto de conexión de de la aplicación a la base de datos:

**config.py:**

```
class BasicConfig:
    ##### ARCHIVO #####
    # RUTA DE ARCHIVO
    ARCHIVO_NOMBRE = "archivo-a-cargar/technical_challenge_data"
    ARCHIVO_FORMATO = "csv"
    ARCHIVO_SEPARADOR = ","
    ARCHIVO_ENCODING = "utf-8"

    ##### ENDPOINTS API MELI #####
    SRVBE_ITEMS = {'idServicio': "SRVBE_ITEMS", 'url': "https://api.mercadolibre.com/items/", 'nivel': 1}
    SRVBE_CATEGORIAS = {'idServicio': "SRVBE_CATEGORIAS", 'url': "https://api.mercadolibre.com/categories/", 'nivel': 2}
    SRVBE_MONEDAS = {'idServicio': "SRVBE_MONEDAS", 'url': "https://api.mercadolibre.com/currencias/", 'nivel': 2}
    SRVBE_USUARIOS = {'idServicio': "SRVBE_USUARIOS", 'url': "https://api.mercadolibre.com/users/", 'nivel': 2}

    ##### THREADPOOLEXECUTOR #####
    # Número de hilos para procesar concurrentemente
    THREADPOOLEXECUTOR_NUM_WORKERS = 50

    ##### Database #####
    SQLALCHEMY_DATABASE_URI = "mysql+mysqlconnector://challengeml:1challengeml@db:3306/meli"
    SQLALCHEMY_ENGINE_OPTIONS = {'pool_size': 50, 'max_overflow': 5}
    SQLALCHEMY_TRACK_MODIFICATIONS = False

    ##### Otras Funcionalidades #####
    BORRAR_TODOS_REGISTROS_ITEMSBD = True
```

**docker-compose.yml:**

```
version: "2"
services:
  app:
    build: ./ml-items-challenge
    restart: always
    links:
      - db
    ports:
      - "5000:5000"

  db:
    image: mysql:8.0.23
    command: '--default-authentication-plugin=mysql_native_password'
    restart: always
    ports:
      - "3306:3306"
    environment:
      MYSQL_ROOT_PASSWORD: root
      MYSQL_USER: challengeml
      MYSQL_PASSWORD: 1challengeml
      MYSQL_DATABASE: meli
    volumes:
      - ./db/init.sql:/docker-entrypoint-initdb.d/init.sql:ro
```

comando a ejecutar: **docker-compose up -d --build**

y revisar su ejecución exitosa, como se ve en la siguiente imagen:

```
C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\ddd-python-rest-mysql>
C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\ddd-python-rest-mysql>docker-compose up -d --build
Creating network "ddd-python-rest-mysql_default" with the default driver
Pulling db (mysql:8.0.23)...
8.0.23: Pulling from library/mysql
Digest: sha256:c35eb76bbccfd0138c8c68ccb9b4cffe42c488a27f64ddc31a2b5f65aa93fce6
Status: Downloaded newer image for mysql:8.0.23
Building app
failed to get console mode for stdout: The handle is invalid.
[+] Building 2.2s (10/10) FINISHED
=> [internal] load build definition from Dockerfile 0.1s
=> => transferring dockerfile: 433B 0.0s
=> [internal] load .dockerignore 0.1s
=> => transferring context: 2B 0.0s
=> [internal] load metadata for docker.io/library/python:3.9.1 1.7s
=> [1/5] FROM docker.io/library/python:3.9.1@sha256:ca8bd3c91af8b12c2d04 0.0s
=> [internal] load build context 0.1s
=> => transferring context: 68.05kB 0.1s
=> CACHED [2/5] WORKDIR /app 0.0s
=> CACHED [3/5] COPY requirements.txt . 0.0s
=> CACHED [4/5] RUN pip install -r requirements.txt 0.0s
=> [5/5] COPY . . 0.1s
=> exporting to image 0.1s
=> => exporting layers 0.1s
=> => writing image sha256:58432dd8f6214c4b149e2bec4a25cbf3d9f658f4dedeb 0.0s
=> => naming to docker.io/library/ddd-python-rest-mysql_app 0.0s
Successfully built 58432dd8f6214c4b149e2bec4a25cbf3d9f658f4dedeb4bb26e72484c34f1ba2
Creating ddd-python-rest-mysql_db_1 ... done
Creating ddd-python-rest-mysql_app_1 ... done
C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\ddd-python-rest-mysql>
```

3. Verificar su ejecución exitosa del contenedor de la aplicación y de la Base de Datos:

comando: `docker-compose ps -a`

resultado:

```
C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\ddd-python-rest-mysql>
C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\ddd-python-rest-mysql>docker-compose ps -a
      Name                                Command                                State      Ports
-----
ddd-python-rest-mysql_app_1  python main.py                        Up         0.0.0.0:5000->5000/tcp
ddd-python-rest-mysql_db_1  docker-entrypoint.sh --def ...      Up         0.0.0.0:3306->3306/tcp, 33060/tcp
C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\ddd-python-rest-mysql>
```

4. Teniendo en cuenta el nombre de los contenedores, en este caso “ddd-python-rest-mysql\_app\_1”, revisar los logs, que la aplicación esté arriba:

comando: `docker logs ddd-python-rest-mysql_app_1`

resultado:

```
C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\ddd-python-rest-mysql>docker logs ddd-python-rest-mysql_app_1
* Serving Flask app "main" (lazy loading)
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: off
* Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
```

El contenedor de la BD vamos a entrar para verificar que se haya creado la tabla de “item” en el esquema “meli” con el nombre del contenedor, en este caso, “ddd-python-rest-mysql\_db\_1”

comando: `docker exec -it ddd-python-rest-mysql_db_1 mysql -uroot -p`

A continuación de ejecutar el comando nos solicita el **password**, el cual es: **root**

Entramos a la consola de mysql y ejecutamos

comando: `select * from meli.item;`

resultado: la tabla creada con 0 registros:

```
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS                               NAMES
33d2e6b8adb3   ddd-python-rest-mysql_app           "python main.py"        8 minutes ago   Up 8 minutes   0.0.0.0:5000->5000/tcp            ddd-python-rest-mysql_app_1
931912d5cc6f   mysql:8.0.23                        "docker-entrypoint.s..." 8 minutes ago   Up 8 minutes   0.0.0.0:3306->3306/tcp, 33060/tcp  ddd-python-rest-mysql_db_1

C:\Users\DianaCarolinaHerrera>docker exec -it ddd-python-rest-mysql_db_1 mysql -uroot -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.23 MySQL Community Server - GPL

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> select * from meli.item;
Empty set (0.00 sec)
```

5. Ejecutar el servicio rest expuesto por la aplicación creada (<http://127.0.0.1:5000/api/v1/almacenardatositems>, método: Post)

Las siguientes propiedades están configuradas por defecto, para la ejecución concurrente del procesamiento de cada registro (Patrón Thread Pool) y el pool de conexiones a la Base de Datos, los cuales son parámetros configurables dentro del archivo `./ml-items-challenge/config.py`:

```
class BasicConfig:
    ##### ARCHIVO #####
    # RUTA DE ARCHIVO
    ARCHIVO_NOMBRE = "archivo-a-cargar/technical_challenge_data"
    ARCHIVO_FORMATO = "csv"
    ARCHIVO_SEPARADOR = ","
    ARCHIVO_ENCODING = "utf-8"

    ##### ENDPOINTS API MELI #####
    SRVBE_ITEMS = {'idServicio': "SRVBE_ITEMS", 'url': "https://api.mercadolibre.com/items/", 'nivel': 1}
    SRVBE_CATEGORIAS = {'idServicio': "SRVBE_CATEGORIAS", 'url': "https://api.mercadolibre.com/categorias/", 'nivel': 2}
    SRVBE_MONEDAS = {'idServicio': "SRVBE_MONEDAS", 'url': "https://api.mercadolibre.com/currencias/", 'nivel': 2}
    SRVBE_USUARIOS = {'idServicio': "SRVBE_USUARIOS", 'url': "https://api.mercadolibre.com/users/", 'nivel': 2}

    ##### THREADPOOLEXECUTOR #####
    # Número de hilos para procesar concurrentemente
    THREADPOOLEXECUTOR_NUM_WORKERS = 50

    ##### Database #####
    SQLALCHEMY_DATABASE_URI = "mysql+mysqlconnector://challengeml:1challengeml@db:3306/meli"
    SQLALCHEMY_ENGINE_OPTIONS = {'pool_size': 50, 'max_overflow': 5}
    SQLALCHEMY_TRACK_MODIFICATIONS = False

    ##### Otras Funcionalidades #####
    BORRAR_TODOS_REGISTROS_ITEMSBD = True
```

como también algunos otros parámetros como el nombre del archivo a procesar.

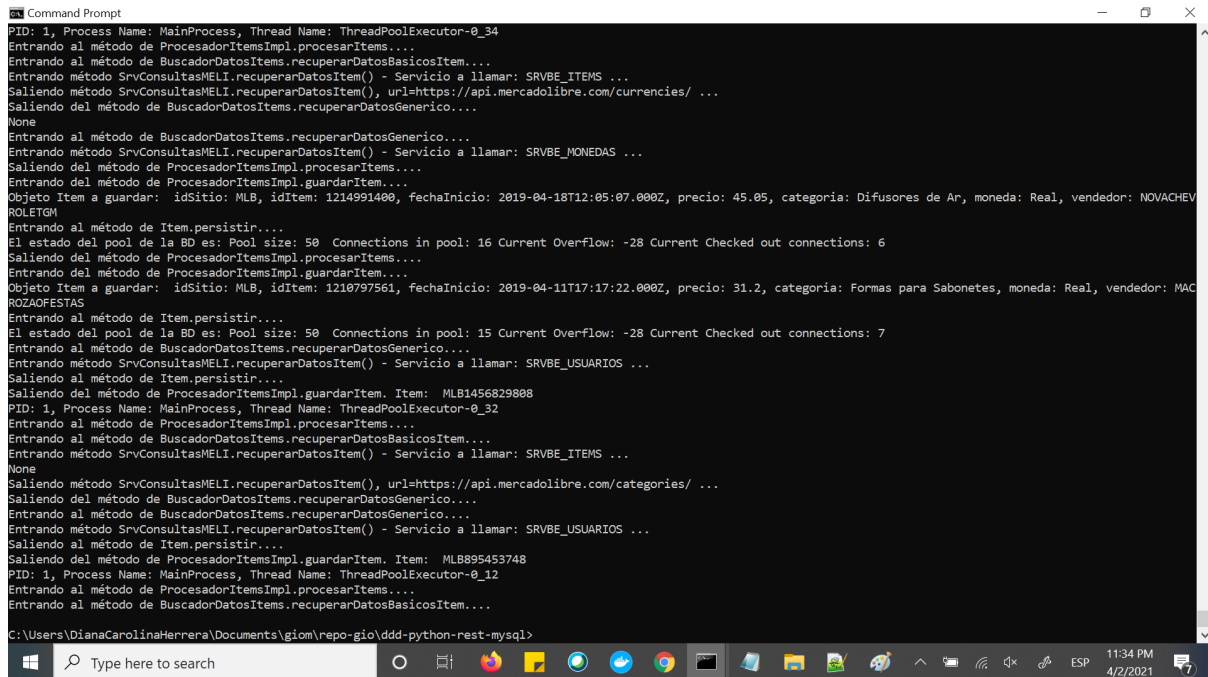
En otra consola ejecutar el siguiente comando para consumir el servicio Rest o desde un otro cliente que se tenga:

comando: `curl -X POST http://127.0.0.1:5000/api/v1/almacenardatositems`

```
C:\Users\DianaCarolinaHerrera>curl -X POST http://127.0.0.1:5000/api/v1/almacenardatositems
```

6. Validación Ejecución: mientras en otra terminal ejecutamos el comando de nuevo de logs para revisar que la aplicación se esté ejecutando exitosamente:

comando: `docker logs ddd-python-rest-mysql_app_1`



```
Command Prompt
PID: 1, Process Name: MainProcess, Thread Name: ThreadPoolExecutor-0_34
Entrando al método de ProcesadorItemsImpl.procesarItems....
Entrando al método de BuscadorDatosItems.recuperarDatosBasicosItem....
Entrando método SrvConsultasMELI.recuperarDatosItem() - Servicio a llamar: SRVBE_ITEMS ...
Saliendo método SrvConsultasMELI.recuperarDatosItem(), url=https://api.mercadolibre.com/currencias/ ...
Saliendo del método de BuscadorDatosItems.recuperarDatosGenerico....
None
Entrando al método de BuscadorDatosItems.recuperarDatosGenerico....
Entrando método SrvConsultasMELI.recuperarDatosItem() - Servicio a llamar: SRVBE_MONEDAS ...
Saliendo del método de ProcesadorItemsImpl.procesarItems....
Entrando del método de ProcesadorItemsImpl.guardarItem....
Objeto Item a guardar: idSitio: MLB, idItem: 1214991400, fechaInicio: 2019-04-18T12:05:07.000Z, precio: 45.05, categoria: Difusores de Ar, moneda: Real, vendedor: NOVACHEV
ROLETGM
Entrando al método de Item.persistir....
El estado del pool de la BD es: Pool size: 50 Connections in pool: 16 Current Overflow: -28 Current Checked out connections: 6
Saliendo del método de ProcesadorItemsImpl.procesarItems....
Entrando del método de ProcesadorItemsImpl.guardarItem....
Objeto Item a guardar: idSitio: MLB, idItem: 1210797561, fechaInicio: 2019-04-11T17:17:22.000Z, precio: 31.2, categoria: Formas para Sabonetes, moneda: Real, vendedor: MAC
ROZAOFEFSTAS
Entrando al método de Item.persistir....
El estado del pool de la BD es: Pool size: 50 Connections in pool: 15 Current Overflow: -28 Current Checked out connections: 7
Entrando al método de BuscadorDatosItems.recuperarDatosGenerico....
Entrando método SrvConsultasMELI.recuperarDatosItem() - Servicio a llamar: SRVBE_USUARIOS ...
Saliendo al método de Item.persistir....
Saliendo del método de ProcesadorItemsImpl.guardarItem. Item: MLB1456829808
PID: 1, Process Name: MainProcess, Thread Name: ThreadPoolExecutor-0_32
Entrando al método de ProcesadorItemsImpl.procesarItems....
Entrando al método de BuscadorDatosItems.recuperarDatosBasicosItem....
Entrando método SrvConsultasMELI.recuperarDatosItem() - Servicio a llamar: SRVBE_ITEMS ...
None
Saliendo método SrvConsultasMELI.recuperarDatosItem(), url=https://api.mercadolibre.com/categories/ ...
Saliendo del método de BuscadorDatosItems.recuperarDatosGenerico....
Entrando al método de BuscadorDatosItems.recuperarDatosGenerico....
Entrando método SrvConsultasMELI.recuperarDatosItem() - Servicio a llamar: SRVBE_USUARIOS ...
Saliendo al método de Item.persistir....
Saliendo del método de ProcesadorItemsImpl.guardarItem. Item: MLB895453748
PID: 1, Process Name: MainProcess, Thread Name: ThreadPoolExecutor-0_12
Entrando al método de ProcesadorItemsImpl.procesarItems....
Entrando al método de BuscadorDatosItems.recuperarDatosBasicosItem....
C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\ddd-python-rest-mysql>
```

La Respuesta al servicio rest con el comando curl;

```
C:\Users\DianaCarolinaHerrera>
C:\Users\DianaCarolinaHerrera>
C:\Users\DianaCarolinaHerrera>
C:\Users\DianaCarolinaHerrera>curl -X POST http://127.0.0.1:5000/api/v1/almacenardatositems
{"mensaje":"Ejecucion Exitosa","numWorkersHilos":50,"registrosProcesados":2000,"tiempo":"Tiempo de procesamiento 119.34777307510376 seconds"}
```

Como podemos ver se demoró 119,347seg al procesar 2000 registros, con 50 hilos concurrentes para el procesamiento de los registros, consultas a los métodos de la API (los cuales también se ejecutan de forma concurrente, después de hacer la primera consulta a ítems).

7. Validación en la Base de Datos: por último volvemos a entrar a la consola de mysql:

comando: `docker exec -it ddd-python-rest-mysql_db_1 mysql -uroot -p`

password: root

y ejecutamos un select a la tabla meli.item:

comando: `select * from meli.item;`

y podemos ver que los 2000 registros se han almacenado en la tabla:

```
mysql>
mysql> select * from meli.item;

+-----+-----+-----+-----+-----+-----+-----+-----+
| site | id      | price | start_time | name      | description | nickname | Timestamp |
+-----+-----+-----+-----+-----+-----+-----+-----+
| BAR  | 1255420109 | NULL | NULL      | NULL      | NULL        | NULL      | 2021-04-03 04:47:10 |
|      | 800585525 | NULL | NULL      | NULL      | NULL        | NULL      | 2021-04-03 04:46:46 |
|      | 844812956 | NULL | NULL      | NULL      | NULL        | NULL      | 2021-04-03 04:46:12 |
| FOO  | 1324932171 | NULL | NULL      | NULL      | NULL        | NULL      | 2021-04-03 04:47:03 |
| MLA  | 603333426 | 250.00 | 2016-01-31T14:04:13.000Z | Libros    | Peso argentino | EL OCTOGONO | 2021-04-03 04:46:03 |
| MLA  | 603523445 | 300.00 | 2016-02-01T14:20:07.000Z | Otros     | Peso argentino | LIBROSPROLOGO BECCAR | 2021-04-03 04:45:52 |
| MLA  | 603924637 | NULL | NULL      | NULL      | NULL        | NULL      | 2021-04-03 04:45:25 |
| MLA  | 603995132 | 599.00 | 2016-02-03T16:09:51.000Z | Libros    | Peso argentino | GERCELESIA | 2021-04-03 04:45:46 |
| MLA  | 604189521 | 2774.92 | 2016-03-05T02:00:00.000Z | Libros    | Peso argentino | ESTEFANIA_CBA | 2021-04-03 04:46:11 |
| MLA  | 604563399 | 500.00 | 2016-02-07T04:09:42.000Z | Otros     | Peso argentino | HALPOSTA | 2021-04-03 04:46:02 |
| MLA  | 604611841 | 100.00 | 2016-02-07T17:27:20.000Z | Bermudas y Shorts | Peso argentino | IMPORT-CHIC | 2021-04-03 04:45:34 |
| MLA  | 605397579 | 800.00 | 2016-02-12T06:34:49.000Z | Revistas  | Peso argentino | ROCKCRIOLLO | 2021-04-03 04:46:10 |
| MLA  | 605831239 | NULL | 2016-02-14T23:44:34.000Z | Remeras, Musculosas y Chombas | NULL | DEHOYNOPASA_ML | 2021-04-03 04:46:29 |
| MLA  | 606067990 | 2813.00 | 2016-02-16T00:59:19.000Z | Cuidado Facial | Peso argentino | NOELIAGAL | 2021-04-03 04:46:15 |
| MLA  | 606252251 | 17447.36 | 2016-02-16T22:44:26.000Z | Otros     | Peso argentino | LUISBALDOMOTOS | 2021-04-03 04:46:09 |
| MLA  | 606506163 | 290.00 | 2016-02-18T12:04:30.000Z | Libros    | Peso argentino | MC LIBROS_ANTIGUOS | 2021-04-03 04:46:19 |
| MLA  | 606598214 | 12900.00 | 2016-03-02T07:00:00.000Z | Camperas, Tapados y Trenches | Peso argentino | GIACCA MODA | 2021-04-03 04:46:11 |
| MLA  | 606691586 | 1484.40 | 2016-02-19T13:19:01.000Z | Chips Leds | Peso argentino | ELUMILED | 2021-04-03 04:45:32 |
| MLA  | 607119257 | 160.00 | 2016-02-22T02:49:12.000Z | Libros    | Peso argentino | CAMILAAZCONA | 2021-04-03 04:45:30 |
| MLA  | 607438646 | 90.00 | 2016-02-23T19:11:24.000Z | Libros    | Peso argentino | CAMILAAZCONA | 2021-04-03 04:45:31 |
| MLA  | 607528496 | 550.00 | 2016-02-24T04:01:47.000Z | Fotos     | Peso argentino | ELCANDUCHO IMPRESION | 2021-04-03 04:46:03 |
| MLA  | 608030900 | 450.00 | 2016-02-26T19:28:08.000Z | Libros    | Peso argentino | THESAURUS | 2021-04-03 04:45:53 |
| MLA  | 608437127 | 6178.00 | 2016-02-29T13:08:55.000Z | Espirales | Peso argentino | REPUESTOS_ROMA | 2021-04-03 04:45:31 |
| MLA  | 608437293 | 7746.00 | 2016-02-29T13:08:58.000Z | Espirales | Peso argentino | REPUESTOS_ROMA | 2021-04-03 04:46:17 |
| MLA  | 608458654 | 295.13 | 2016-02-29T13:58:47.000Z | Música    | Peso argentino | LOSTOMASOL | 2021-04-03 04:46:03 |

+-----+-----+-----+-----+-----+-----+-----+-----+
| MLB  | 970757907 | 529.00 | 2018-02-01T15:25:02.000Z | Emblemas  | Real         | STIAUTOPARTS | 2021-04-03 04:46:42 |
|      | 972911695 | NULL | NULL      | NULL      | NULL        | NULL        | 2021-04-03 04:47:12 |
|      | 974213803 | 49.00 | 2018-02-06T23:17:45.000Z | Tapetes   | Real         | JKAUTOMOTIVO | 2021-04-03 04:46:44 |
|      | 978957485 | 43.62 | 2018-02-15T21:53:51.000Z | Outros    | Real         | JKAUTOMOTIVO | 2021-04-03 04:46:53 |
|      | 979877408 | 33.57 | 2018-02-17T09:21:10.000Z | Outros    | Real         | TSBTOOLS | 2021-04-03 04:46:40 |
|      | 981600456 | NULL | NULL      | NULL      | NULL        | NULL        | 2021-04-03 04:46:34 |
|      | 982779272 | 39.00 | 2018-02-22T19:01:08.000Z | Alavanca Freio de Mão | Real         | CROWOPARTSONLINE | 2021-04-03 04:47:11 |
|      | 983385627 | 401.24 | 2018-02-23T19:28:49.000Z | Hélices   | Real         | RADICAL PEI8AS2013 | 2021-04-03 04:46:52 |
|      | 986573821 | 35.42 | 2018-03-02T15:00:51.000Z | Anéis     | Real         | DIVINAPROVIDENCIARELIGIOSOS | 2021-04-03 04:47:30 |
|      | 986791922 | 33.11 | 2018-03-03T00:31:48.000Z | Camisetas | Real         | MAIS TRANSFER | 2021-04-03 04:47:16 |
|      | 987225702 | NULL | 2018-03-04T12:54:42.000Z | Outros    | NULL        | DENTAL_KIWI | 2021-04-03 04:47:01 |
|      | 987582138 | 159.00 | 2018-03-05T13:42:35.000Z | Sonda Lambda | Real         | MASIL AUTOPEI8AS | 2021-04-03 04:46:56 |
|      | 988429135 | 122.46 | 2018-03-06T20:14:00.000Z | Bandejas de Suspensão | Real         | ACCIOLY GM | 2021-04-03 04:47:11 |
|      | 989784956 | 249.99 | 2018-03-08T15:38:14.000Z | Teclados  | Real         | VENDASE-GUGA | 2021-04-03 04:46:44 |
|      | 989919544 | 359.00 | 2018-03-08T19:37:48.000Z | Amperímetros | Real         | DMS-SHOP- BELO-HORIZONTE | 2021-04-03 04:46:48 |
|      | 990273026 | NULL | NULL      | NULL      | NULL        | NULL        | 2021-04-03 04:46:53 |
|      | 995683468 | NULL | NULL      | NULL      | NULL        | NULL        | 2021-04-03 04:46:41 |
|      | 995862025 | NULL | NULL      | NULL      | NULL        | NULL        | 2021-04-03 04:47:21 |
|      | 996630686 | 89.00 | 2018-03-20T01:14:08.000Z | Almofadas | Real         | ABELHINHAS LEMBRANCINHAS | 2021-04-03 04:46:55 |
|      | 997048525 | 199.00 | 2018-03-20T18:20:07.000Z | Póira-Lamas | Real         | DESTAKEAUTOPECAS | 2021-04-03 04:46:48 |
|      | 997065680 | 99.00 | 2018-03-20T18:37:17.000Z | Outros    | Real         | FERRO-VELHO-TIBAGI | 2021-04-03 04:47:28 |
|      | 997525661 | 79.86 | 2018-03-21T13:23:12.000Z | Buchas    | Real         | ALCXIMPORTS | 2021-04-03 04:46:41 |
|      | 998239385 | 95.00 | 2018-03-22T13:15:02.000Z | Mochilas  | Real         | OLIST | 2021-04-03 04:47:07 |
|      | A         | NULL | NULL      | NULL      | NULL        | NULL        | 2021-04-03 04:46:31 |

+-----+-----+-----+-----+-----+-----+-----+-----+
2000 rows in set (0.01 sec)

mysql>
```

O hacemos un count():

comando: `select count(1) from meli.item;`

resultado:

```
mysql>
mysql> select count(1) from meli.item;
+-----+
| count(1) |
+-----+
|      2000 |
+-----+
1 row in set (0.01 sec)
```

## Características Importantes a tener en cuenta:

- El programa tiene la característica de almacenar los datos que no sean correctos o que no obtienen resultados de la consulta a los métodos de la API de Mercado Libre, para su posterior análisis.



- Otra característica es que se puede volver a ejecutar el ejercicio, debido a que cuando arranca se borran todos los registros de la tabla meli.item y es parametrizable por medio la propiedad **BORRAR\_TODOS\_REGISTROS\_ITEMSBD** que está en el archivo config.py con valor “True”:

```
class BasicConfig:
    ##### ARCHIVO #####
    # RUTA DE ARCHIVO
    ARCHIVO_NOMBRE = "archivo-a-cargar/technical_challenge_data"
    ARCHIVO_FORMATO = "csv"
    ARCHIVO_SEPARADOR = ","
    ARCHIVO_ENCODING = "utf-8"

    ##### ENDPOINTS API MELI #####
    SRVBE_ITEMS = {'idServicio': "SRVBE_ITEMS", 'url': "https://api.mercadolibre.com/items/", 'nivel': 1}
    SRVBE_CATEGORIAS = {'idServicio': "SRVBE_CATEGORIAS", 'url': "https://api.mercadolibre.com/categories/", 'nivel': 2}
    SRVBE_MONEDAS = {'idServicio': "SRVBE_MONEDAS", 'url': "https://api.mercadolibre.com/monedas/", 'nivel': 2}
    SRVBE_USUARIOS = {'idServicio': "SRVBE_USUARIOS", 'url': "https://api.mercadolibre.com/users/", 'nivel': 2}

    ##### THREADPOOLEXECUTOR #####
    # Número de hilos para procesar concurrentemente
    THREADPOOLEXECUTOR_NUM_WORKERS = 50

    ##### Database #####
    SQLALCHEMY_DATABASE_URI = "mysql+mysqlconnector://challenge1:challenge1@db:3306/meli"
    SQLALCHEMY_ENGINE_OPTIONS = {'pool_size': 50, 'max_overflow': 5}
    SQLALCHEMY_TRACK_MODIFICATIONS = False

    ##### Otras Funcionalidades #####
    BORRAR_TODOS_REGISTROS_ITEMSBD = True
```

- Si se requiere cambiar alguna propiedad del archivo **./ml-items-challenge/config.py**, para que los cambios tengan efecto se debe bajar los contenedores:

comando: **docker-compose down -v**

y luego de nuevo volver a construir el contenedor:

comando: **docker-compose up -d --build.**

```
C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\ddd-python-rest-mysql>
C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\ddd-python-rest-mysql>docker-compose down -v
Stopping ddd-python-rest-mysql_app_1 ... done
Stopping ddd-python-rest-mysql_db_1 ... done
Removing ddd-python-rest-mysql_app_1 ... done
Removing ddd-python-rest-mysql_db_1 ... done
Removing network ddd-python-rest-mysql_default

C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\ddd-python-rest-mysql>
C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\ddd-python-rest-mysql>docker-compose up -d --build
Creating network "ddd-python-rest-mysql_default" with the default driver
Building app
failed to get console mode for stdout: The handle is invalid.
[+] Building 2.0s (10/10) FINISHED
=> [internal] load build definition from Dockerfile 0.0s
=> => transferring dockerfile: 32B 0.0s
=> [internal] load .dockerignore 0.0s
=> => transferring dockerfile: 2B 0.0s
=> [internal] load metadata for docker.io/library/python:3.9.1 1.7s
=> [1/5] FROM docker.io/library/python:3.9.1@sha256:ca8bd3c91af8b12c2d04 0.0s
=> [internal] load build context 0.0s
=> => transferring context: 4.73kB 0.0s
=> CACHED [2/5] WORKDIR /app 0.0s
=> CACHED [3/5] COPY requirements.txt . 0.0s
=> CACHED [4/5] RUN pip install -r requirements.txt 0.0s
=> [5/5] COPY . . 0.1s
=> exporting to image 0.1s
=> => exporting layers 0.1s
=> => writing image sha256:02e59b706885cee1396b95a1df489d8b672a3c06a595c 0.0s
=> => naming to docker.io/library/ddd-python-rest-mysql_app 0.0s
Successfully built 02e59b706885cee1396b95a1df489d8b672a3c06a595cb4c96e4b7573b27b9e2
Creating ddd-python-rest-mysql_db_1 ... done
Creating ddd-python-rest-mysql_app_1 ... done

C:\Users\DianaCarolinaHerrera\Documents\giom\repo-gio\ddd-python-rest-mysql>
```