

Tarea 1 BigData

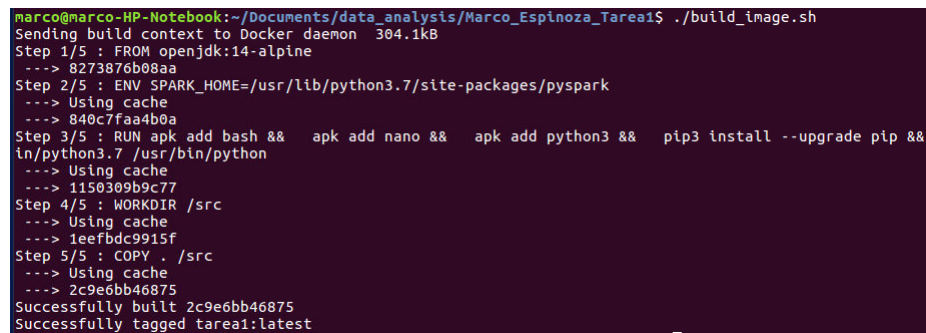
Marco Espinoza Murillo

Julio 2020

1 Instrucciones de ejecución

Desde el path raíz, hacer build de la imagen:

`./build_image.sh`

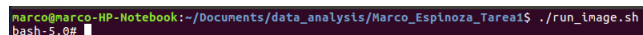


```
marco@marco-HP-Notebook:~/Documents/data_analysis/Marco_Espinoza_Tarea1$ ./build_image.sh
Sending build context to Docker daemon 304.1kB
Step 1/5 : FROM openjdk:14-alpine
--> 8273876b08aa
Step 2/5 : ENV SPARK_HOME=/usr/lib/python3.7/site-packages/pyspark
--> Using cache
--> 840c7faa4b0a
Step 3/5 : RUN apk add bash && apk add nano && apk add python3 && pip3 install --upgrade pip &&
in/python3.7 /usr/bin/python
--> Using cache
--> 1150309b9c77
Step 4/5 : WORKDIR /src
--> Using cache
--> 1eefbdc9915f
Step 5/5 : COPY . /src
--> Using cache
--> 2c9e6bb46875
Successfully built 2c9e6bb46875
Successfully tagged tareal:latest
```

Figure 1: Generación de la imagen

Después, se ejecuta la imagen:

`./run_image.sh`



```
marco@marco-HP-Notebook:~/Documents/data_analysis/Marco_Espinoza_Tarea1$ ./run_image.sh
bash-5.0#
```

Figure 2: Ejecución de la imagen

Una vez en la imagen, el programa se ejecuta de la siguiente manera:

`spark-submit programaestudiante.py estudiante.csv curso.csv nota.csv`

O también, se puede ejecutar utilizando el siguiente comando:

`./programaestudiante.sh`

Como resultado, se obtiene lo siguiente para un $n=3$:

```
20/07/23 02:18:07 INFO CodeGenerator: Code generated in 11.675998 ms
```

carrera	carnet	nombre	Promedio	rank
Mantenimiento	200963493	Xinia Lopez	92.1875	1
Mantenimiento	201304095	Jorge Rojas	89.1875	2
Mantenimiento	200957883	Melany Maroto	88.15789473684211	3
Electronica	201541413	Lourdes Navas	89.1875	1
Electronica	201309739	Gerardo Quesada	89.0625	2
Electronica	201543275	Marco Benavidez	88.06666666666666	3
Industrial	201347020	Lisette Fernandez	90.3529411764706	1
Industrial	201680416	Heizel Benavidez	89.6842105263158	2
Industrial	200926538	Luis Murillo	89.5	3
Administracion	201788127	Gerardo Guardia	91.66666666666667	1
Administracion	201567750	Marco Arrieta	90.83333333333333	2
Administracion	201179436	Heizel Cordero	89.33333333333333	3
Civil	201086048	Marco Cordero	91.35	1
Civil	201471999	Esteban Castro	89.1	2
Civil	200921207	Armando Chinchilla	88.58823529411765	3
Computacion	201688123	Marco Chinchilla	89.45	1
Computacion	201364518	Esteban Chinchilla	88.42105263157895	2
Computacion	201119164	Luis Rodriguez	87.86666666666666	3
Biotecnologia	201169188	Xinia Ortiz	89.8	1
Biotecnologia	201169270	Gerardo Lopez	87.73684210526316	2
Biotecnologia	201790717	Lourdes Serrano	87.15384615384616	3

Figure 3: Resultado de la ejecución de programaestudiante.py

Finalmente, para ejecutar las pruebas a través de pytest, solamente se debe ejecutar lo siguiente:

pytest

```
bash-5.0# pytest
===== test session starts =====
platform linux -- Python 3.7.7, pytest-5.4.3, py-1.9.0, pluggy-0.13.1
rootdir: /src
collected 8 items

test_programaestudiante.py ..... [100%]

===== warnings summary =====
/usr/lib/python3.7/site-packages/pyspark/sql/context.py:77
/usr/lib/python3.7/site-packages/pyspark/sql/context.py:77: DeprecationWarning: Deprecated in 3.0.0. Use SparkSession.builder.getOrCreate() instead.
  DeprecationWarning)

-- Docs: https://docs.pytest.org/en/latest/warnings.html
===== 8 passed, 1 warning in 25.66s =====
bash-5.0#
```

Figure 4: Resultado de la ejecución de pytest

En los comentarios del código de pytest se explica cada una de las pruebas realizadas.

2 Solución de problemas

En caso de algún problema con los archivos de la tarea, se puede clonar la solución desde el siguiente path de git:

```
git clone https://github.com/mespinoza86/BigData.git
```