



Universidad **Ricardo Palma**

RECTORADO
PROGRAMA DE ESPECIALIZACIÓN EN CIENCIA DE DATOS

Formamos seres humanos para una cultura de paz

BIG DATA APLICADO


SESIÓN 04

Expositores:

David Narváez

Eder Pineda

bigdataplicado@gmail.com

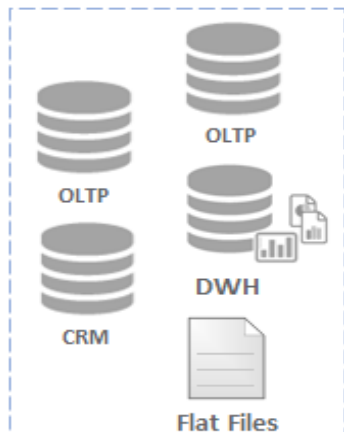


***Big data** is high-volume, high-velocity and/or high-variety information assets that demand cost-effective, innovative forms of information processing that enable enhanced insight, decision making, and process automation (**Gartner**).*



Arquitectura Lambda

DATA SOURCES

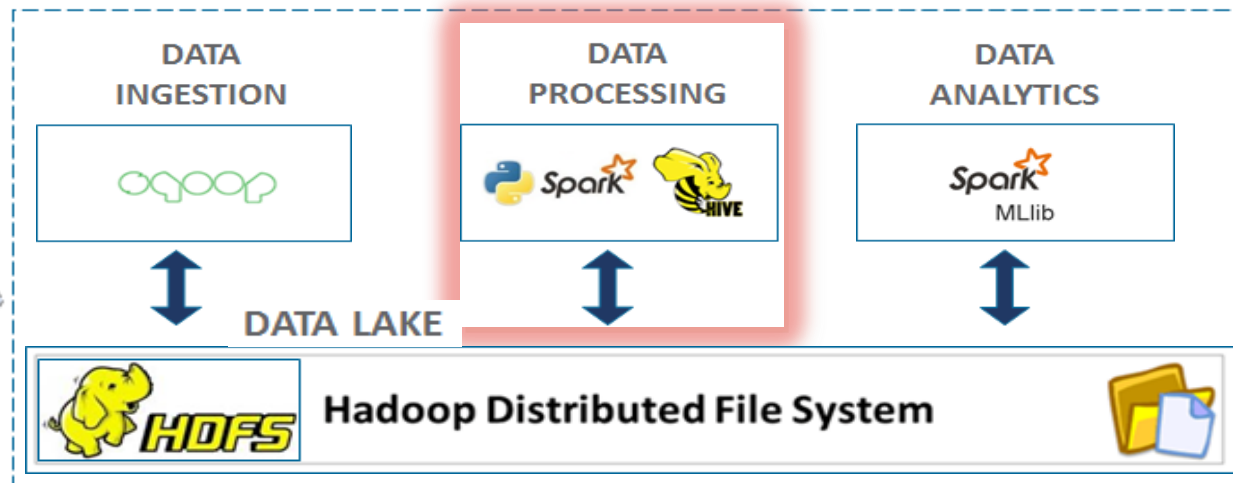


Batch

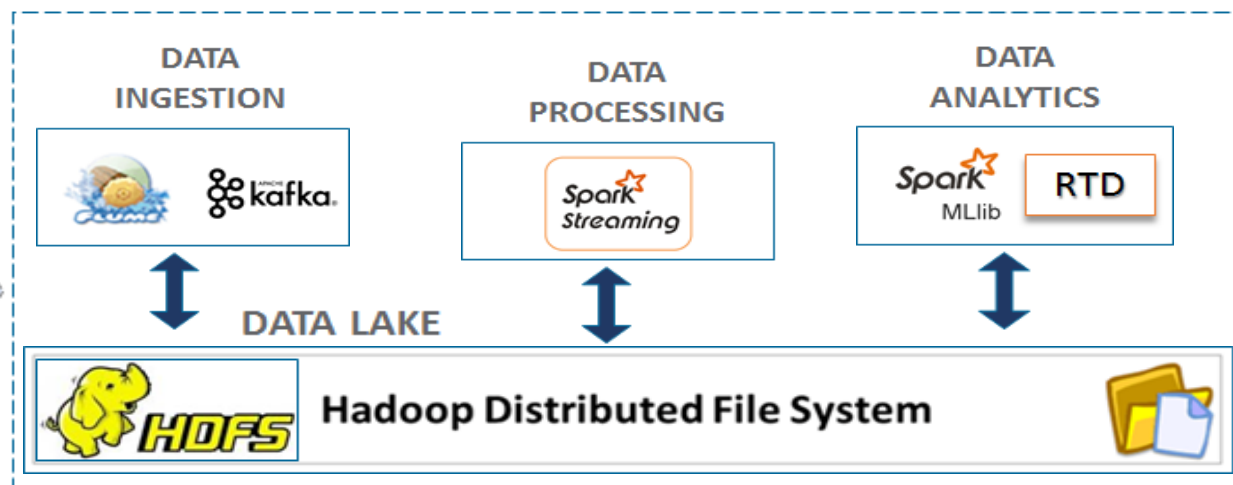
Real Time



BATCH LAYER



SPEED LAYER



SERVING LAYER

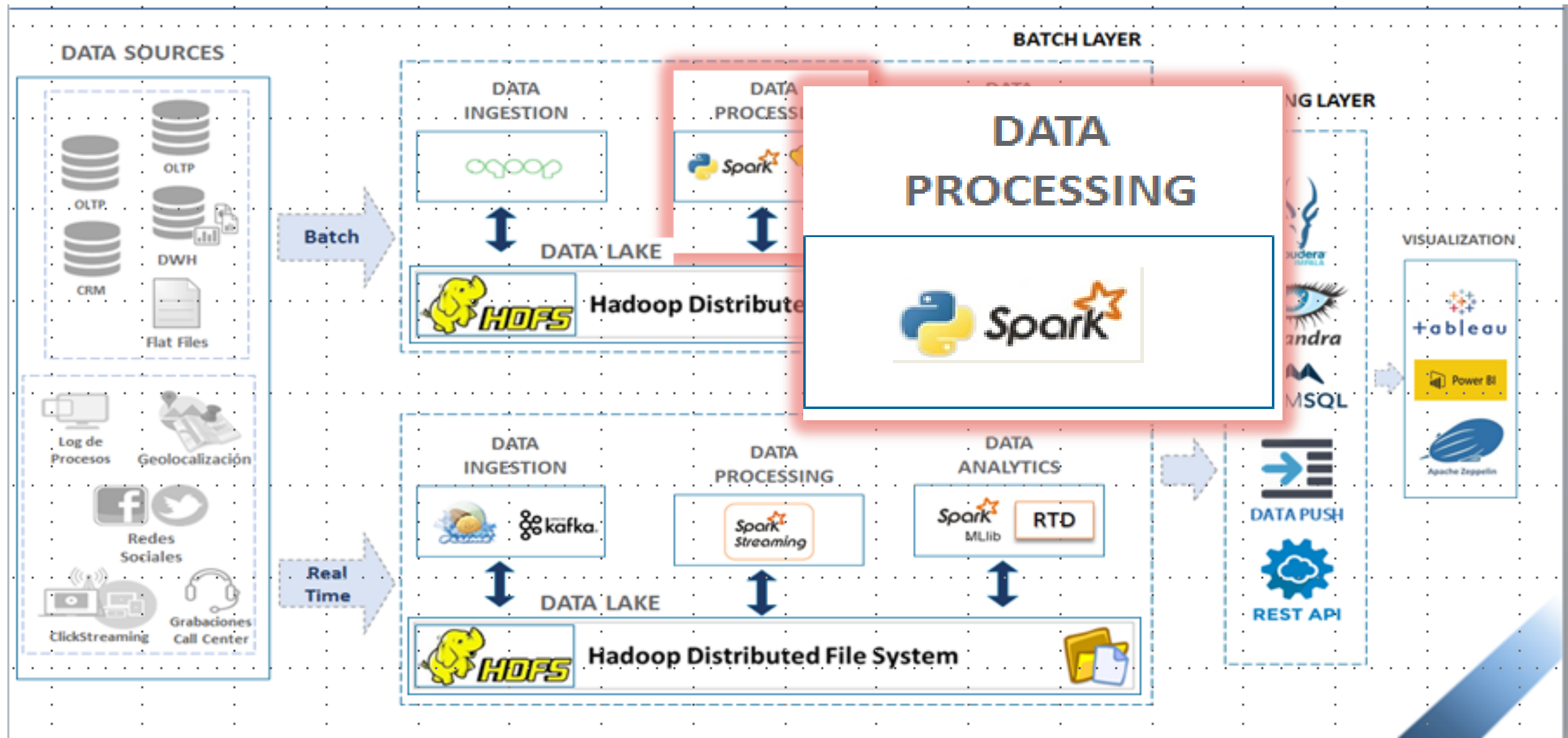


VISUALIZATION





Arquitectura Lambda



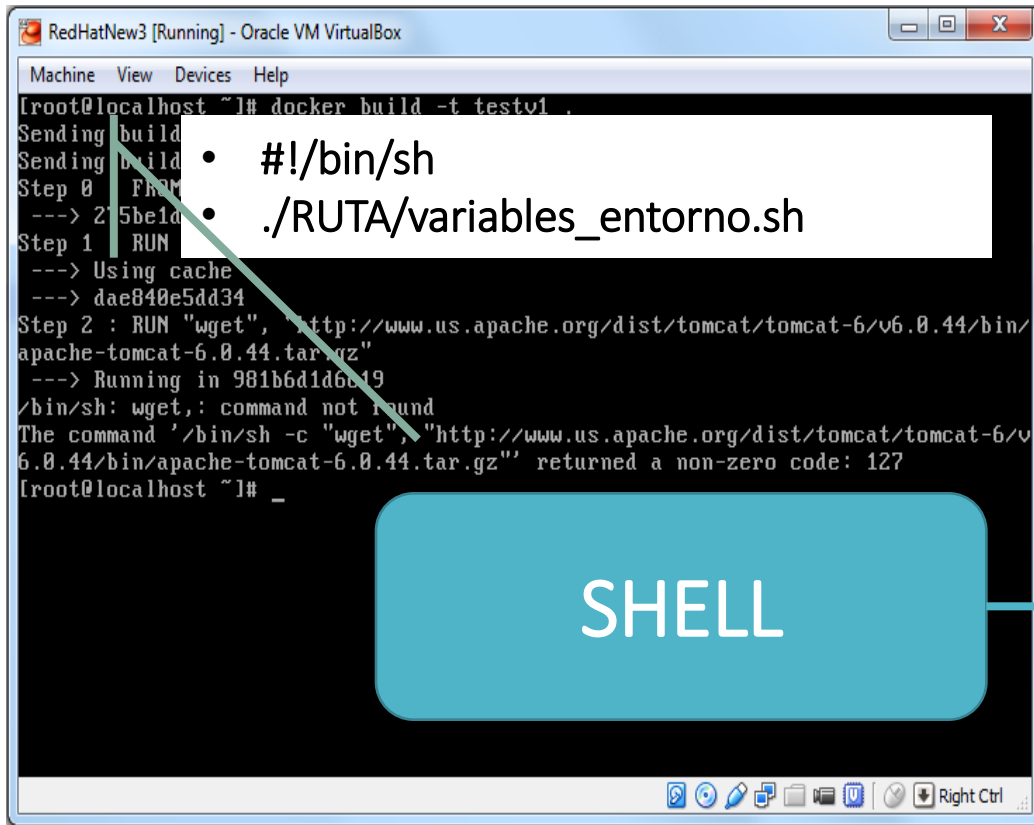
1. Industrializando Ejecución de Proceso

- Cambiar de entorno exploratorio a productivo
- Permite enviar el código de tu aplicación a un cluster y ejecutarlo desde ahí
- Ofrece varios controles con los que puede especificar los recursos que su aplicación
- El comando para realizar esta tarea es el `spark-submit`

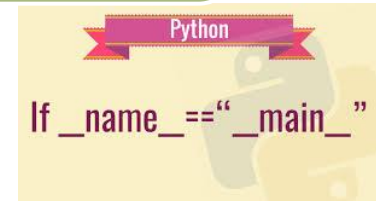
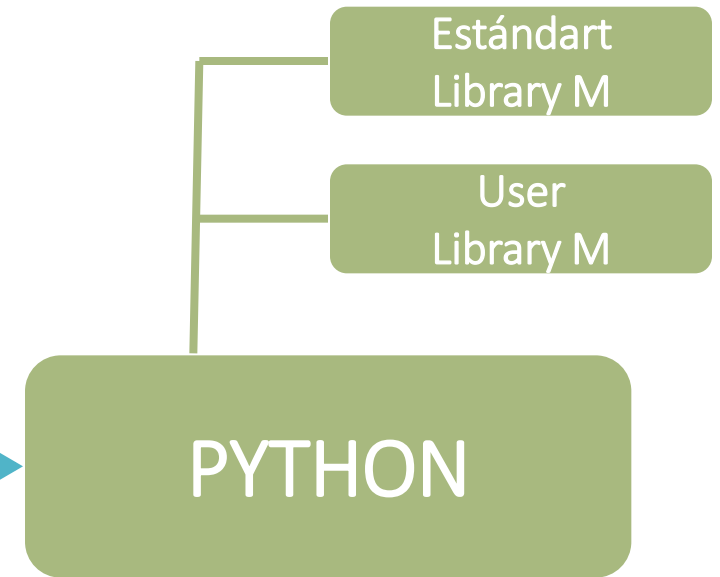
1. Industrializando Ejecución de Proceso

- Cambiar de entorno exploratorio a productivo
- Permite enviar el código de tu aplicación a un cluster y ejecutarlo desde ahí
- Ofrece varios controles con los que puede especificar los recursos que su aplicación
- El comando para realizar esta tarea es el `spark-submit`

1. Industrializando Ejecución de Proceso



```
./bin/spark-submit \  
--class <main-class> \  
--master <master-url> \  
--deploy-mode <deploy-mode> \  
--conf <key>=<value> \  
... # other options  
<application-jar-or-script> \  
[application-arguments]
```



1. Introducción a SPARK

Creando el archivo shell: miShell.sh



usr_big_data@lmappbigdata04:~/data/shellunix

```
[usr_big_data@lmappbigdata04 shellunix]$ touch miShell.sh
[usr_big_data@lmappbigdata04 shellunix]$ ls -ltr
total 0
-rw-rw-r-- 1 usr_big_data usr_big_data 0 Aug 22 12:06 miShell.sh
[usr_big_data@lmappbigdata04 shellunix]$
```

1. Introducción a SPARK

Creando el archivo de variables de entorno: set_env.sh



usr_big_data@lmappbigdata04:~/data/shellunix

```
[usr_big_data@lmappbigdata04 shellunix]$ touch set_env.sh
```

```
[usr_big_data@lmappbigdata04 shellunix]$ ls -ltr
```

```
total 8
```

```
-rw-rw-r-- 1 usr_big_data usr_big_data    0 Aug 22 14:40 set_env.sh
```

1. Introducción a SPARK

El archivo python se creara en Jupyter y se ubicará en la misma ruta donde se encuentran los dos archivos anteriores



usr_big_data@lmappbigdata04:~/data/shellunix

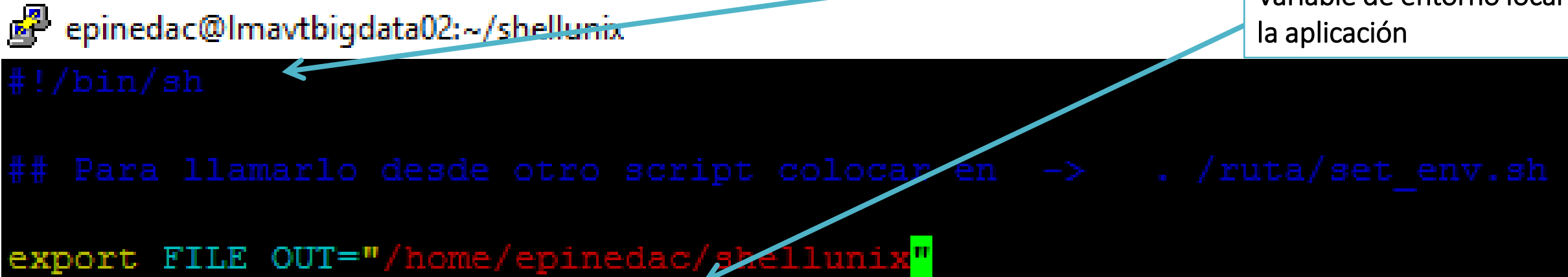
```
[usr_big_data@lmappbigdata04 shellunix]$ touch set_env.sh
[usr_big_data@lmappbigdata04 shellunix]$ ls -ltr
total 8
-rwxrwxrwx 1 usr_big_data usr_big_data 266 Aug 22 14:22 miShell.sh
-rw-rw-r-- 1 usr_big_data usr_big_data 1223 Aug 22 14:35 myPythonFile.py
-rw-rw-r-- 1 usr_big_data usr_big_data 0 Aug 22 14:40 set_env.sh
```

1. Introducción a SPARK

Editando el archivo de variables de entorno:
Utilizamos el comando “vim” de unix para editar el archivo

ruta en la que se encuentra
el intérprete

Variable de entorno local de
la aplicación

A terminal window showing a shell script. The prompt is 'epinedac@lmavtbigdata02:~/shellunix'. The script content includes a shebang line, a comment about calling the script from another script, and an export statement for 'FILE_OUT'. Three blue arrows point from text boxes to specific parts of the script: one to the shebang line, one to the path in the comment, and one to the value in the export statement.

```
epinedac@lmavtbigdata02:~/shellunix
#!/bin/sh

## Para llamarlo desde otro script colocar en -> . /ruta/set_env.sh

export FILE_OUT="/home/epinedac/shellunix"
```

1. Introducción a SPARK

Editando el archivo de shell:

Utilizamos el comando “vim” de unix para editar el archivo

```
usr_big_data@lmappbigdata04:~/data/shellunix
```

```
[usr_big_data@lmappbigdata04 shellunix]$ vim miShell.sh
```



```
epinedac@lmavtbigdata02:/home/epinedac/shellunix
```

```
#!/bin/sh
```

```
. /home/epinedac/shellunix/set_env.sh # Para inicializar nuestras variables
```

```
echo $FILE_OUT
```

```
spark2-submit \
```

```
--conf "spark.yarn.executor.memoryOverhead=512M" \
```

```
$FILE_OUT/myPythonFile.py $FILE_OUT > $FILE_OUT/resultado.log
```


1. Introducción a SPARK

Editando el archivo de Python: myPythonFile.py

```
epinedac@lmavtbigdata02:/home/epinedac/shellunix
#####
#   Organizacion      : MI_EMPRESA
#   Programa          : myPythonFile.py
#   Creado por         : Eder Pineda
#   Fecha Creacion    : 22/08/2018
#   Proposito          : imprime cantidad de filas de dataframe
#
#   Fuentes de datos  : Archivo.csv
#   Destino            : NA
#
#####
def main():
    mi_dictcionario={
        'cod':['0','1','2','3','4','5','6','7'],
        'desc':['cero','primer','segundo','tercero','cuarto','quinto','sexto','septimo']
    }

    pd_dataframe=pd.DataFrame(mi_dictcionario)
    cantidad=pd_dataframe.count()
    print("cantidaddddddd : ",cantidad)
    print("-----")

import findspark
findspark.init("/opt/cloudera/parcels/SPARK2-2.2.0.cloudera1-1.cdh5.12.0.p0.142354/lib/spark2/")
import pyspark
from pyspark.sql import SparkSession
import pandas as pd

spark = SparkSession.builder.appName('imprime_catidad_registros').getOrCreate()

if __name__ == '__main__':
    main()
```

Thank you

A close-up photograph of a right hand holding a silver ballpoint pen, writing the words "Thank you" in a cursive script on a white surface. The pen is positioned at the end of the word "you", and the hand is angled towards the bottom right of the frame. The background is a plain, light-colored surface.