

- ✗ 1. Why is writing Java MapReduce programs not ideal for data analysis and data processing tasks such as ETL?
- ☐ A analyst needs to be a Java programmer
 - ☒ B development is slow (compiling, debugging, deploying)
 - ☒ C writing MapReduce programs offers not enough flexibility
 - ☐ D data analysis and data processing are not Big data tasks
 - ☒ E tasks need multiple MR jobs (challenging to handle, data between jobs needs to be stored on HDFS)

- ✓ 2. How can you kill a running PIG job with job ID 'xxx' ? Enter one command as if you would execute it in the unix terminal.

The answer to this question is not in the slides.

HINT: Once your job is being executed it *is* a MapReduce job.

```
mapred job -kill xxx
```

- ✓ 3. Step 1: Instead of working in the GRUNT shell locally, how can you run a PIG job named **my_job.pig** locally? Provide the command as you would enter it in the terminal.


```
pig -x local my_job.pig
```

- ✓ 4. Step 2: How many fields does ad_data1 have after ETL processing?

- ☐ A 4
- ☐ B 5
- ☐ C 6
- ☒ D 8
- ☐ E none of the above


- ⊘ 5. Step 2: How many records does ad_data1 have after you completed your ETL processing?

- ☐ A 734,579
- ☐ B 250,461
- ☐ C 384,399
- ☒ D 438,389
- ☐ E None of the above

 6. Step 2: Why are we using **integers** to represent *cpc* (instead of **double** or **float**)?
it is enough to represent the cpc.
it save the storage.

 7. Step 3: How many fields does ad_data2 have after ETL processing?

- ☐ A 4
- ☐ B 5
- ☐ C 6
- ☒ D 8
- ☐ E none of the above

 8. Step 3: How many records does ad_data2 have after you completed your ETL processing?

- ☐ A 221,516
- ☒ B 350,563
- ☐ C 474,037
- ☐ D 578,443
- ☐ E None of the above