DIGITAL TALENT SCHOLARSHIP 2019

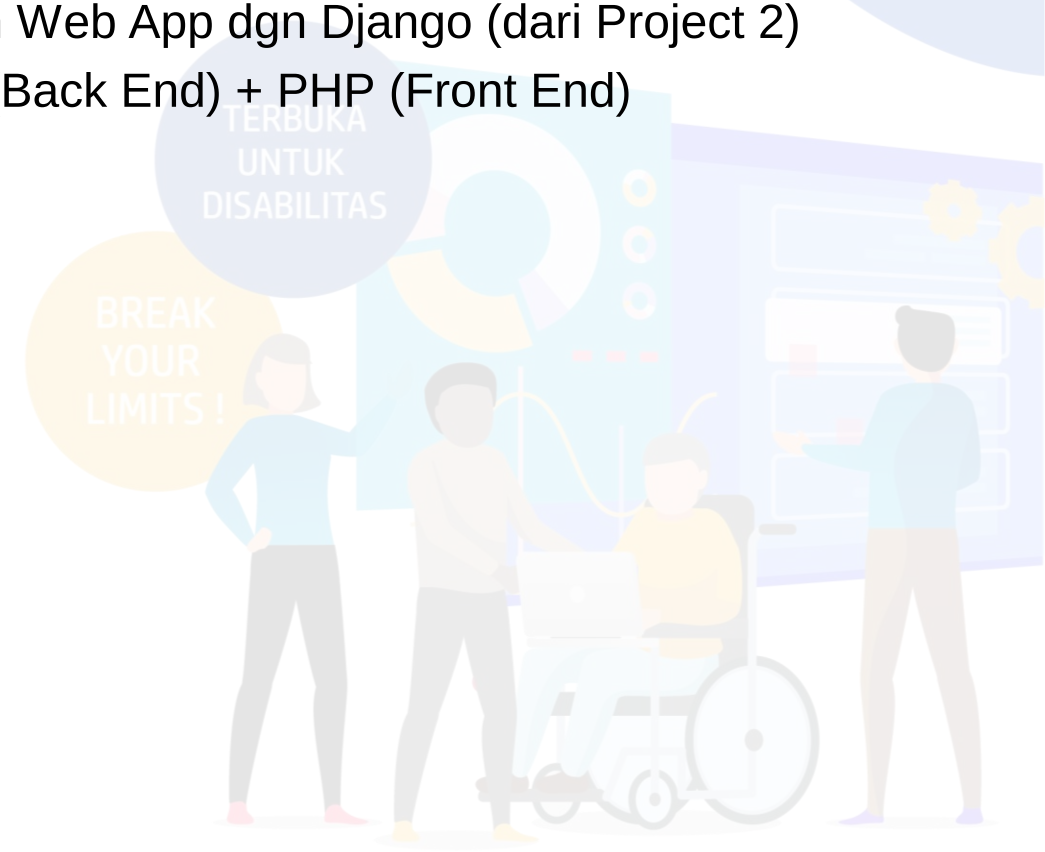
Big Data Analytics

Project 3: Konfigurasi RESTful API untuk Web App & Mobile App + Spark Streaming

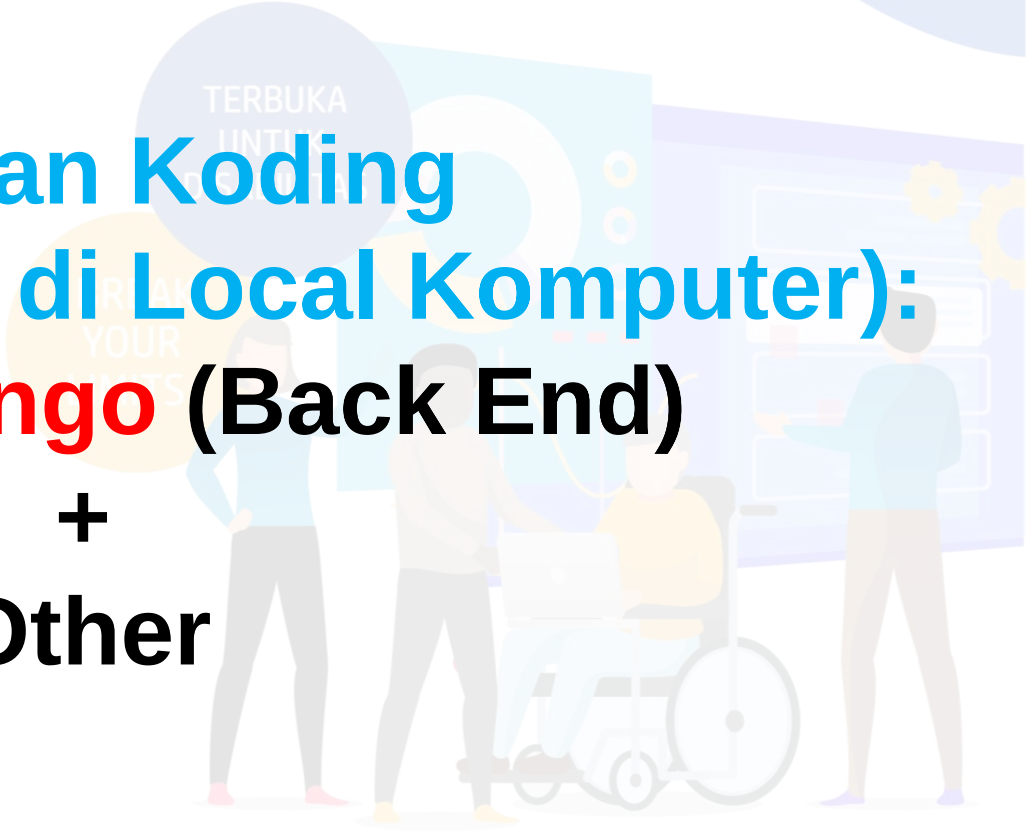
Oleh: Imam Cholissodin | imamcs@ub.ac.id, Putra Pandu Adikara, Sufia Adha Putri
Asisten: Guedho, Sukma, Anshori, Aang dan Gusti
Fakultas Ilmu Komputer (Filkom) Universitas Brawijaya (UB)

Pokok Pembahasan

- Mobile App dgn Ionic/Flutter dan Web App dgn Django (dari Project 2)
- Contoh Project: Python Django (Back End) + PHP (Front End)

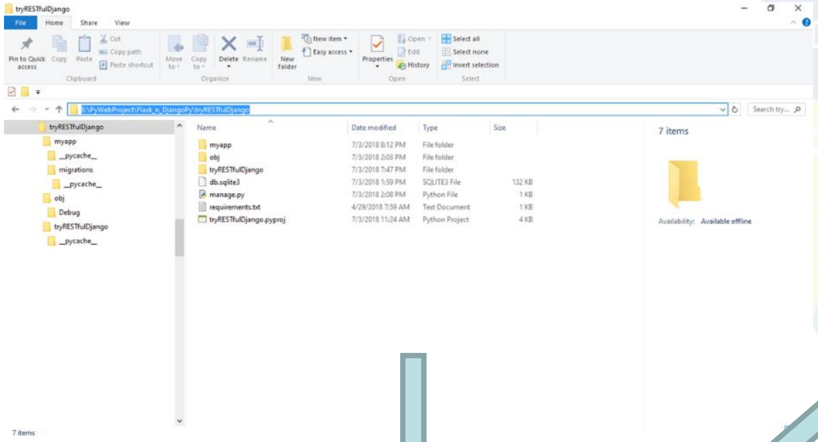


Tahapan Koding
(Dari Contoh Fix di Local Komputer):
Python Django (Back End)
+
Other



Python Django (Web)

- Tahapan Koding RESRful di Python Django (Web):
  - Buka cmd dari folder project “tryRESTfulDjango”, pada path bagian yang diblok warna biru ketik “cmd” tanpa quote, lalu tekan enter, ketik “python manage.py runserver”

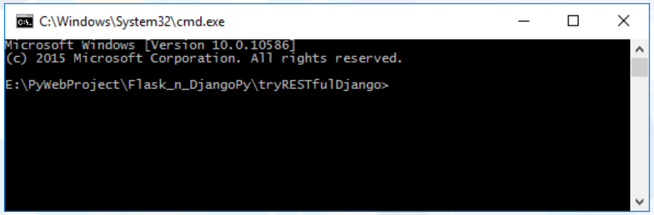


E:\PyWebProject\Flask_n_DjangoPy\tryRESTfulDjango

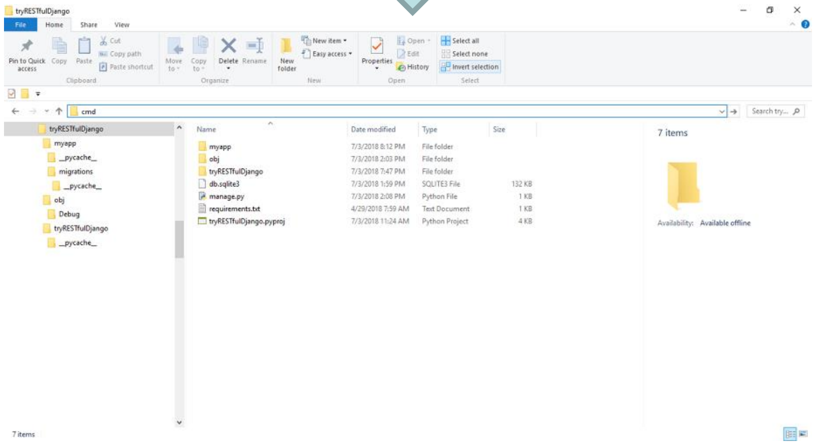
```
tryRESTfulDjango
  myapp
    __pycache__
    migrations
      __pycache__
  obj
    Debug
  tryRESTfulDjango
    __pycache__
```

Name	Date modified	Type	Size
myapp	7/3/2018 8:12 PM	File folder	
obj	7/3/2018 2:03 PM	File folder	
tryRESTfulDjango	7/3/2018 7:47 PM	File folder	
db.sqlite3	7/3/2018 1:59 PM	SQLITE3 File	132 KB
manage.py	7/3/2018 2:08 PM	Python File	1 KB
requirements.txt	4/29/2018 7:59 AM	Text Document	1 KB
tryRESTfulDjango.pyproj	7/3/2018 11:24 AM	Python Project	4 KB

7 items



```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.10586]
(c) 2015 Microsoft Corporation. All rights reserved.

E:\PyWebProject\Flask_n_DjangoPy\tryRESTfulDjango>
```

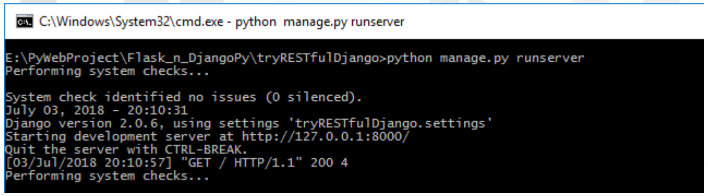


cmd

```
tryRESTfulDjango
  myapp
    __pycache__
    migrations
      __pycache__
  obj
    Debug
  tryRESTfulDjango
    __pycache__
```

Name	Date modified	Type	Size
myapp	7/3/2018 8:12 PM	File folder	
obj	7/3/2018 2:03 PM	File folder	
tryRESTfulDjango	7/3/2018 7:47 PM	File folder	
db.sqlite3	7/3/2018 1:59 PM	SQLITE3 File	132 KB
manage.py	7/3/2018 2:08 PM	Python File	1 KB
requirements.txt	4/29/2018 7:59 AM	Text Document	1 KB
tryRESTfulDjango.pyproj	7/3/2018 11:24 AM	Python Project	4 KB

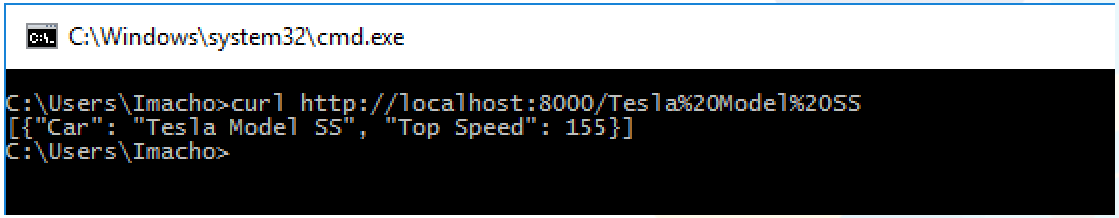
7 items



```
C:\Windows\System32\cmd.exe - python  manage.py runserver

E:\PyWebProject\Flask_n_DjangoPy\tryRESTfulDjango>python manage.py runserver
Performing system checks...

System check identified no issues (0 silenced).
July 03, 2018 - 20:10:31
Django version 2.0.6, using settings 'tryRESTfulDjango.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
[03/Jul/2018 20:10:57] "GET / HTTP/1.1" 200 4
Performing system checks...
```

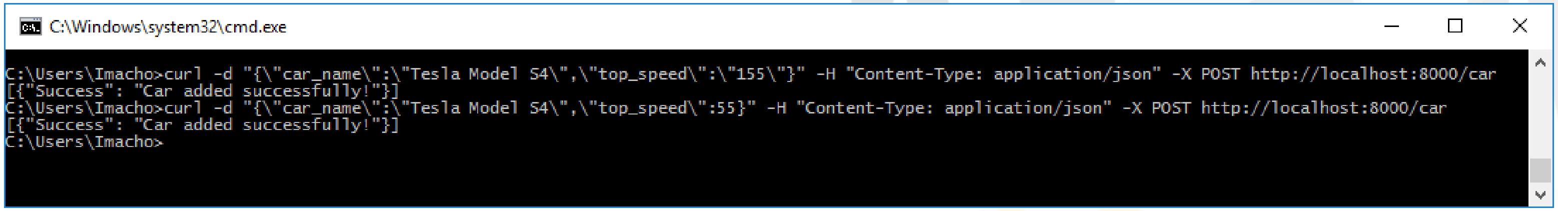
Python Django (Web)

- Tahapan Koding RESRful di Python Django (Web):
  - Coba ketik “curl http://localhost:8000/Tesla%20Model%20SS” terlihat bahwa django GET akses property mobil dengan nama “Tesla Model SS” dengan mengembalikan dua property



```
C:\Users\Imacho>curl http://localhost:8000/Tesla%20Model%20SS
[{"Car": "Tesla Model SS", "Top Speed": 155}]
C:\Users\Imacho>
```

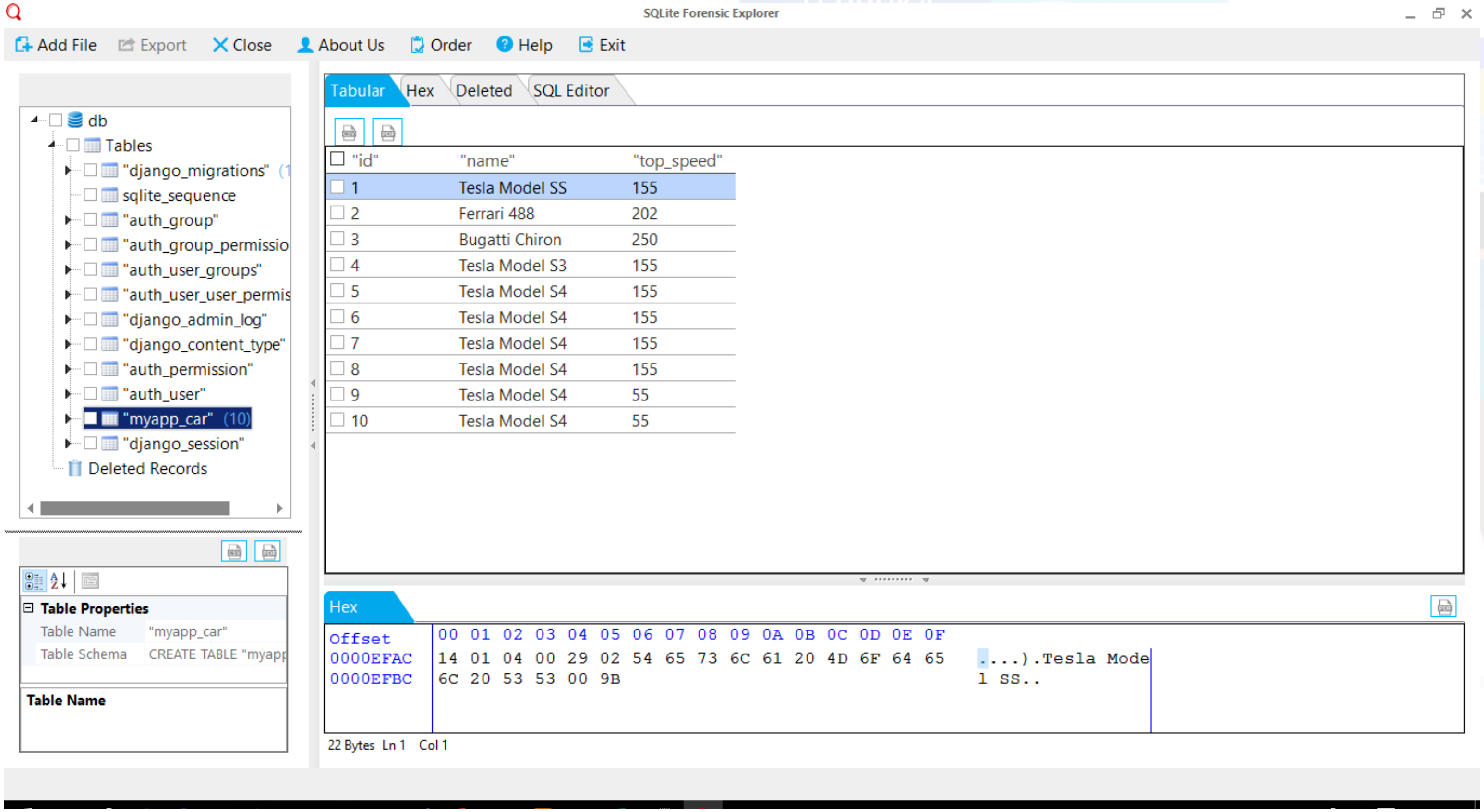
  - Coba ketik “curl -d "{\"car_name\":\"Tesla Model S4\",\"top_speed\":\"155\"}" -H "Content-Type: application/json" -X POST http://localhost:8000/car” atau “curl -d "{\"car_name\":\"Tesla Model S4\",\"top_speed\":55}" -H "Content-Type: application/json" -X POST http://localhost:8000/car” untuk menambah data mobil ke database “db.sqlite3”



```
C:\Users\Imacho>curl -d "{\"car_name\":\"Tesla Model S4\",\"top_speed\":\"155\"}" -H "Content-Type: application/json" -X POST http://localhost:8000/car
[{"Success": "Car added successfully!"}]
C:\Users\Imacho>curl -d "{\"car_name\":\"Tesla Model S4\",\"top_speed\":55}" -H "Content-Type: application/json" -X POST http://localhost:8000/car
[{"Success": "Car added successfully!"}]
C:\Users\Imacho>
```

Python Django (Web)

- Tahapan Koding RESRful di Python Django (Web):
  - Cek dengan Aplikasi sqlite untuk melihat hasil update data mobil:



"id"	"name"	"top_speed"
1	Tesla Model SS	155
2	Ferrari 488	202
3	Bugatti Chiron	250
4	Tesla Model S3	155
5	Tesla Model S4	155
6	Tesla Model S4	155
7	Tesla Model S4	155
8	Tesla Model S4	155
9	Tesla Model S4	55
10	Tesla Model S4	55

Table Properties

Table Name	"myapp_car"
Table Schema	CREATE TABLE "myapp

Table Name

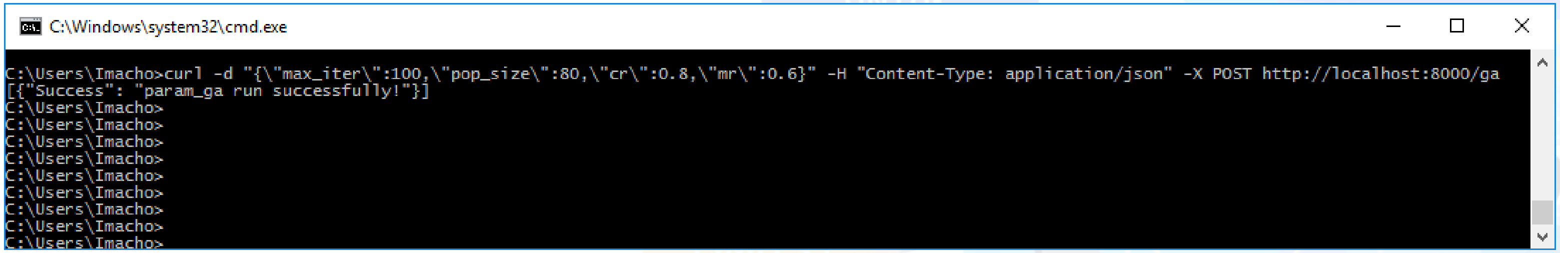
Hex

```
Offset    00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
0000EFAC  14 01 04 00 29 02 54 65 73 6C 61 20 4D 6F 64 65   ....).Tesla Mode
0000EFBC  6C 20 53 53 00 9B                                 l SS..
```

22 Bytes Ln 1 Col 1

Python Django (Web)

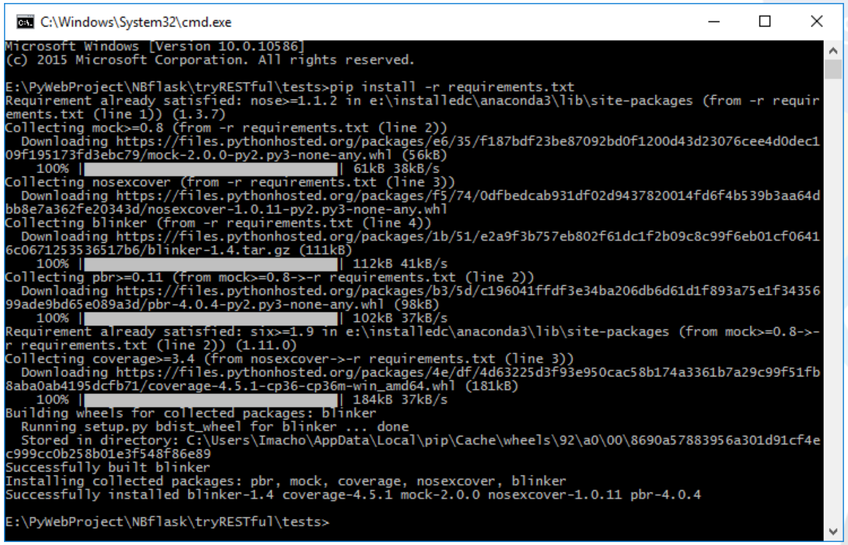
- Tahapan Koding RESRful di Python Django (Web):
  - Coba ketik “curl -d "{\"max_iter\":100,\"pop_size\":80,\"cr\":0.8,\"mr\":0.6}" -H "Content-Type: application/json" -X POST http://localhost:8000/ga” untuk kemudian dilakukan proses algoritma genetika



```
C:\Users\Imacho>curl -d "{\"max_iter\":100,\"pop_size\":80,\"cr\":0.8,\"mr\":0.6}" -H "Content-Type: application/json" -X POST http://localhost:8000/ga
[{"Success": "param_ga run successfully!"}]
C:\Users\Imacho>
C:\Users\Imacho>
C:\Users\Imacho>
C:\Users\Imacho>
C:\Users\Imacho>
C:\Users\Imacho>
C:\Users\Imacho>
C:\Users\Imacho>
C:\Users\Imacho>
```

Tambahan Python

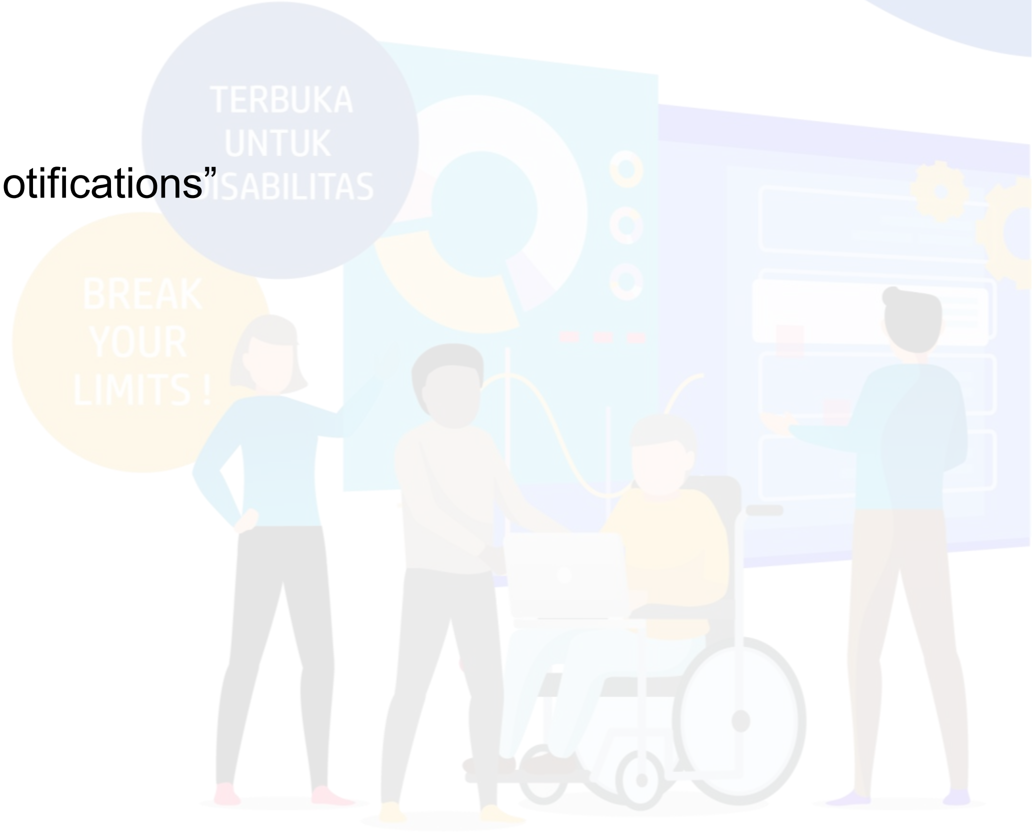
- Tambahan di Python Flask (Web):
  - Cara install dari file “requirements.txt”, dengan “E:\PyWebProject\NBflask\tryRESTful\tests>pip install -r requirements.txt”



```
Microsoft Windows [Version 10.0.10586]
(c) 2015 Microsoft Corporation. All rights reserved.

E:\PyWebProject\NBflask\tryRESTful\tests>pip install -r requirements.txt
Requirement already satisfied: nose>=1.1.2 in e:\installedc\anaconda3\lib\site-packages (from -r requir
ements.txt (line 1)) (1.3.7)
Collecting mock>=0.8 (from -r requirements.txt (line 2))
  Downloading https://files.pythonhosted.org/packages/e6/35/f187bdf23be87092bd0f1200d43d23076cee4d0dec1
09f195173fd3ebc79/mock-2.0.0-py2.py3-none-any.whl (56kB)
    100% |████████████████████████████████| 61kB 38kB/s
Collecting nosexcover (from -r requirements.txt (line 3))
  Downloading https://files.pythonhosted.org/packages/f5/74/0dfbedcab931df02d9437820014fd6f4b539b3aa64d
bb8e7a362fe20343d/nosexcover-1.0.11-py2.py3-none-any.whl
Collecting blinker (from -r requirements.txt (line 4))
  Downloading https://files.pythonhosted.org/packages/1b/51/e2a9f3b757eb802f61dc1f2b09c8c99f6eb01cf0641
6c0671253536517b6/blinker-1.4.tar.gz (111kB)
    100% |████████████████████████████████| 112kB 41kB/s
Collecting pbr>=0.11 (from mock>=0.8->-r requirements.txt (line 2))
  Downloading https://files.pythonhosted.org/packages/b3/5d/c196041ffdf3e34ba206db6d61d1f893a75e1f34356
99ade9bd65e089a3d/pbr-4.0.4-py2.py3-none-any.whl (98kB)
    100% |████████████████████████████████| 102kB 37kB/s
Requirement already satisfied: six>=1.9 in e:\installedc\anaconda3\lib\site-packages (from mock>=0.8->-
r requirements.txt (line 2)) (1.11.0)
Collecting coverage>=3.4 (from nosexcover->-r requirements.txt (line 3))
  Downloading https://files.pythonhosted.org/packages/4e/df/4d63225d3f93e950cac58b174a3361b7a29c99f51fb
8aba0ab4195dcfb71/coverage-4.5.1-cp36-cp36m-win_amd64.whl (181kB)
    100% |████████████████████████████████| 184kB 37kB/s
Building wheels for collected packages: blinker
  Running setup.py bdist_wheel for blinker ... done
  Stored in directory: C:\Users\Imacho\AppData\Local\pip\Cache\wheels\92\a0\00\8690a57883956a301d91cf4e
c999cc0b258b01e3f548f86e89
Successfully built blinker
Installing collected packages: pbr, mock, coverage, nosexcover, blinker
Successfully installed blinker-1.4 coverage-4.5.1 mock-2.0.0 nosexcover-1.0.11 pbr-4.0.4

E:\PyWebProject\NBflask\tryRESTful\tests>
```

Tambahan Python

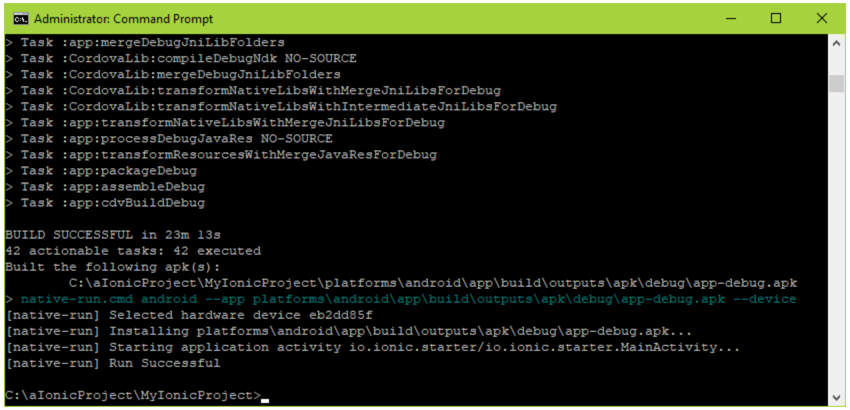
- Tambahan di Python Flask (Web):
  - Install “pip install pyfcm”
  - Install “pip install fcm-django”
  - Install “pip install pusher_push_notifications”



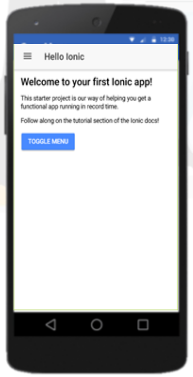
Run Ionic App pada Android Device

- Ketikkan berikut: (tunggu beberapa waktu)

```
C:\aIonicProject\MyIonicProject>npm i -g native-run
C:\Users\Imacho\AppData\Roaming\npm\native-run ->
C:\Users\Imacho\AppData\Roaming\npm\node_modules\native-
run\bin\native-run
+ native-run@0.2.9
added 30 packages from 33 contributors in 16.161s

C:\aIonicProject\MyIonicProject>ionic cordova run android --device
```



```
> Task :app:mergeDebugJniLibFolders
> Task :CordovaLib:compileDebugNdk NO-SOURCE
> Task :CordovaLib:mergeDebugJniLibFolders
> Task :CordovaLib:transformNativeLibsWithMergeJniLibsForDebug
> Task :CordovaLib:transformNativeLibsWithIntermediateJniLibsForDebug
> Task :app:transformNativeLibsWithMergeJniLibsForDebug
> Task :app:processDebugJavaRes NO-SOURCE
> Task :app:transformResourcesWithMergeJavaResForDebug
> Task :app:packageDebug
> Task :app:assembleDebug
> Task :app:cdvBuildDebug

BUILD SUCCESSFUL in 23m 13s
42 actionable tasks: 42 executed
Built the following apk(s):
        C:\aIonicProject\MyIonicProject\platforms\android\app\build\outputs\apk\debug\app-debug.apk
> native-run.cmd android --app platforms\android\app\build\outputs\apk\debug\app-debug.apk --device
[native-run] Selected hardware device eb2dd85f
[native-run] Installing platforms\android\app\build\outputs\apk\debug\app-debug.apk...
[native-run] Starting application activity io.ionic.starter/io.ionic.starter.MainActivity...
[native-run] Run Successful

C:\aIonicProject\MyIonicProject>_
```



Welcome to your first Ionic app!

This starter project is our way of helping you get a functional app running in record time.

Follow along on the tutorial section of the Ionic docs!

Android/iOS Device (di HP Anda)

Instalasi Flutter

- Buka CMD, ketikkan peritah:

Flutter doctor (untuk melihat konfigurasi flutter yang terinstall)

- Konfigurasi pada Visual Studio Code

Start VSC
Install Extension : “flutter”
Buka view – Command palette, lalu
Ketikan perintah “doctor” untuk testing

- Create Flutter Project,

Misal pada drive C anda membuat folder “FlutterProject”, buka CMD dgn ketikkan

```
C:\> mkdir FlutterProject
C:\> cd FlutterProject
C:\FlutterProject> Flutter create <nama app>
C:\FlutterProject> code .
```

→ tekan enter, lalu F5 untuk compile kode flutter

Instalasi Django

- Buka CMD, ketikkan peritah:

```
pip install Django
```

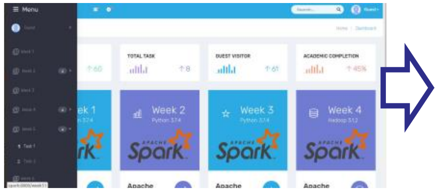
- Copy project InitWebPyUBig101

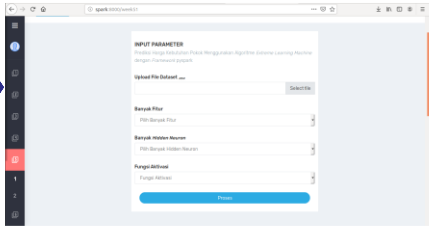
dari link “http://bit.ly/2P6oNDU”
Lalu ekstak pada komputer Anda

- Masuk ke folder hasil ekstrak, lalu untuk run, ketikkan

```
python manage.py runserver
```

Lalu akses week 5, pada task 1 atau 2




spark:8000/week51

INPUT PARAMETER

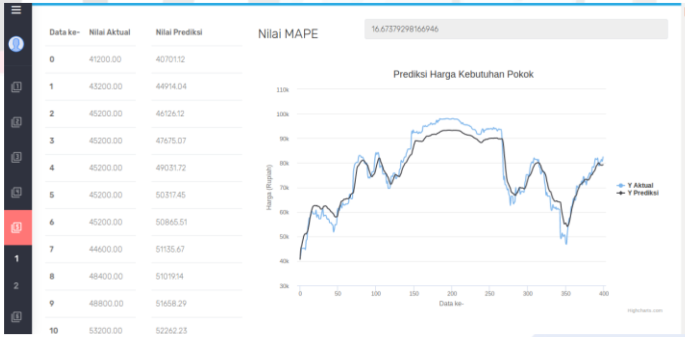
Prediksi Harga Kebutuhan Pokok Menggunakan Algoritme *Extreme Learning Machine* dengan *Framework* pyspark

Upload File Dataset .csv

Banyak Fitur: Pilih Banyak Fitur

Banyak *Hidden Neuron*: Pilih Banyak Hidden Neuron

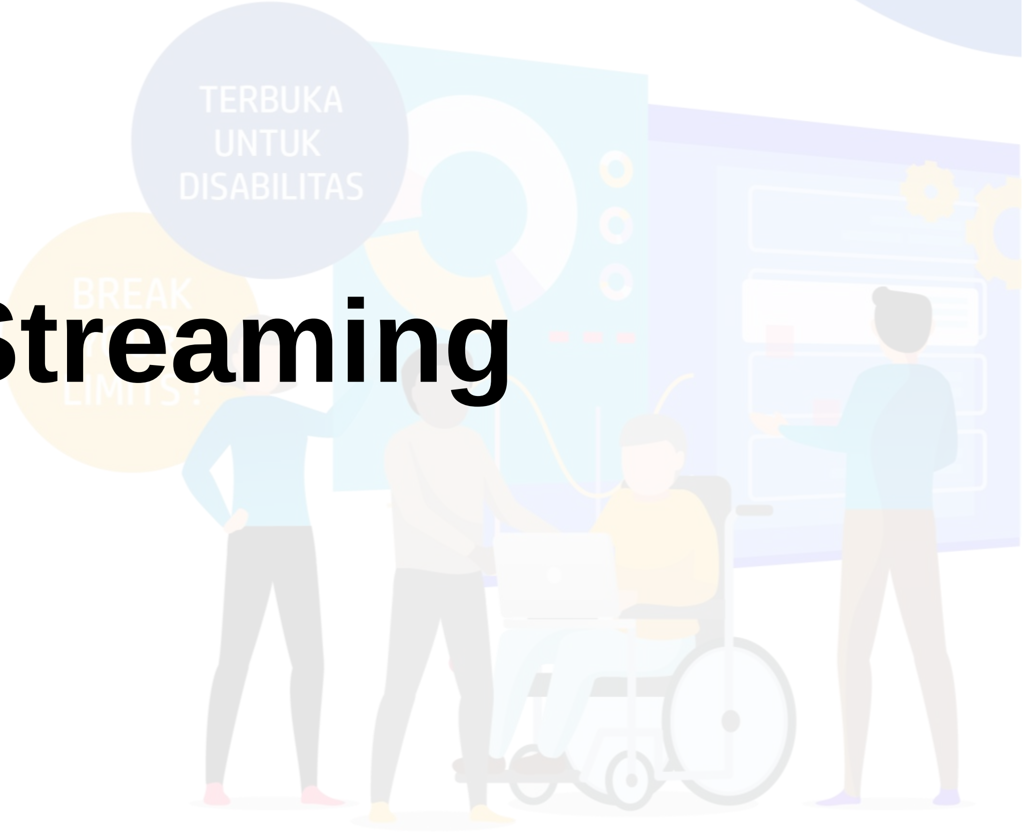
Fungsi Aktivasi: Fungsi Aktivasi



Data ke-	Nilai Aktual	Nilai Prediksi
0	41200.00	40701.12
1	43200.00	44914.04
2	45200.00	46126.12
3	45200.00	47675.07
4	45200.00	49031.72
5	45200.00	50317.45
6	45200.00	50865.51
7	44600.00	51135.67
8	48400.00	51019.14
9	48800.00	51658.29
10	53200.00	52262.23

Nilai MAPE: 16.67379298166946

Spark Streaming



Spark Streaming

- pySparkWordCount streaming:

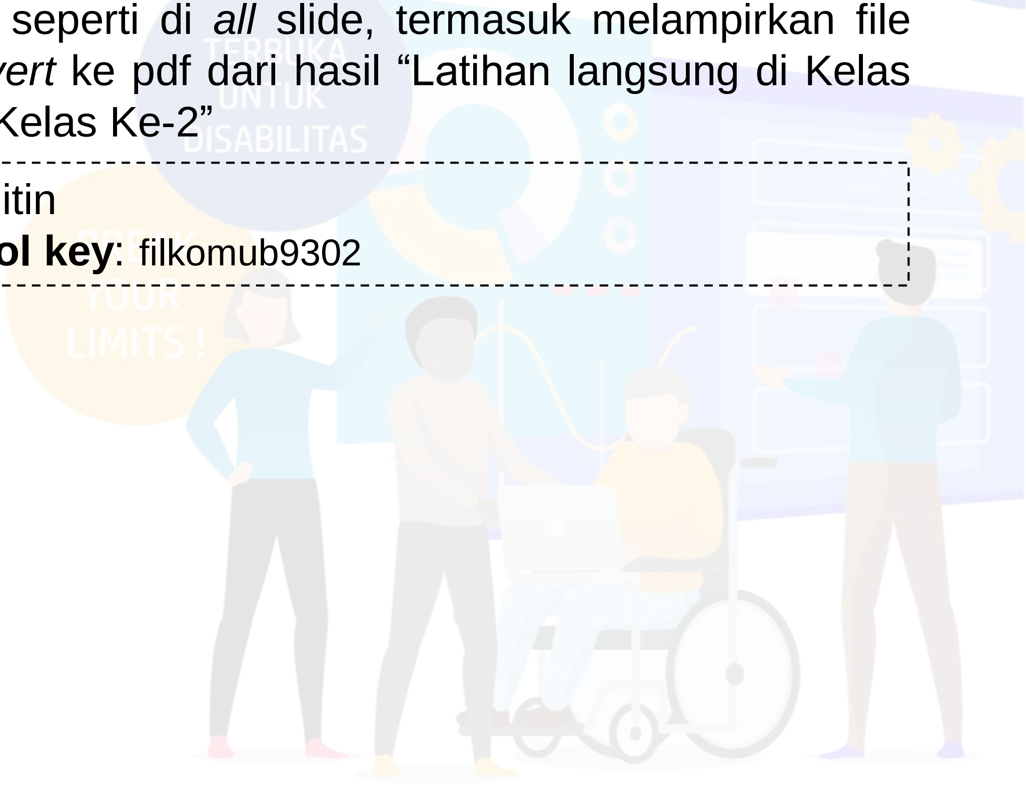
1. Code hdfs_wordcount.py streaming berhasil dijalankan. Link file kode “https://goo.gl/vY6f4E”
   Karena file kode tersebut streaming, maka akan selalu mencari file baru untuk diproses kembali dengan konsep wordcount setiap waktu (misal per detik)

   Pada saat koding dijalankan, untuk melakukan proses wordcount maka masukkan file text sembarang ke alamat /user/hduser/wordcount/inputpada hdfs, misal file terebut adalah “input.txt” dan “input2.txt”, dari link berikut “ https://goo.gl/6d7CWQ”
2. Jalankan pySparkWordCount streaming ke-2. Link file kode “https://goo.gl/cnGuHo”

Tugas Individu

1. Buatlah rangkuman materi di atas dengan cara berikut:
   - ○ Lakukan ulang materi di atas seperti di *all* slide, termasuk melampirkan file *.doc/docx yang sudah di-*convert* ke pdf dari hasil “Latihan langsung di Kelas Ke-1 dan Latihan langsung di Kelas Ke-2”

> Register ke turnitin cek plagiasi diturnitin
> Masukkan **id class**: 21563495 & **enrool key**: filkomub9302



DIGITAL TALENT SCHOLARSHIP 2019

Big Data Analytics

Terimakasih

Oleh: Imam Cholissodin | imamcs@ub.ac.id, Putra Pandu Adikara, Sufia Adha Putri
Asisten: Guedho, Sukma, Anshori, Aang dan Gusti
Fakultas Ilmu Komputer (Filkom) Universitas Brawijaya (UB)