



KOMINFO



KOMINFO



DIGITAL
TALENT
SCHOLARSHIP



DIGITAL
TALENT
SCHOLARSHIP

DIGITAL TALENT SCHOLARSHIP 2019

TERBUKA
UNTUK
DISABILITAS

BREAK
YOUR
LIMITS !



digitalent.kominfo.go.id



digitalent.kominfo.go.id



DIGITAL
TALENT
SCHOLARSHIP

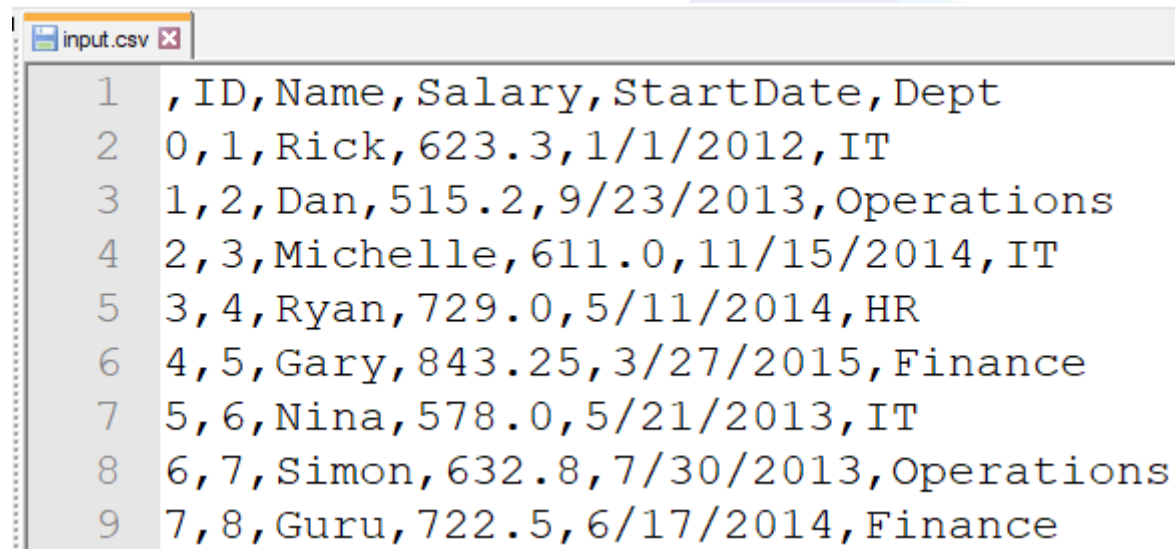
Sesi 12

Python (Input Output)

Big Data Analytics



CSV File



```
input.csv
1 ,ID,Name,Salary,StartDate,Dept
2 0,1,Rick,623.3,1/1/2012,IT
3 1,2,Dan,515.2,9/23/2013,Operations
4 2,3,Michelle,611.0,11/15/2014,IT
5 3,4,Ryan,729.0,5/11/2014,HR
6 4,5,Gary,843.25,3/27/2015,Finance
7 5,6,Nina,578.0,5/21/2013,IT
8 6,7,Simon,632.8,7/30/2013,Operations
9 7,8,Guru,722.5,6/17/2014,Finance
```

- Comma-separated values (CSV) adalah berkas yang menggunakan koma untuk memisahkan nilainya, dapat menyimpan data teks dan angka dalam bentuk *plain text*



DIGITAL
TALENT
SCHOLARSHIP

Python - *Reading a CSV File*

- Fungsi `read_csv` digunakan untuk membaca file CSV sebagai DataFrame

```
▶ import pandas as pd  
data = pd.read_csv('Documents/Data Diyana/input.csv')  
print (data)
```

	id	name	salary	start_date	dept
0	1	Rick	623.30	2012-01-01	IT
1	2	Dan	515.20	2013-09-23	Operations
2	3	Tusar	611.00	2014-11-15	IT
3	4	Ryan	729.00	2014-05-11	HR
4	5	Gary	843.25	2015-03-27	Finance
5	6	Rasmi	578.00	2013-05-21	IT
6	7	Pranab	632.80	2013-07-30	Operations
7	8	Guru	722.50	2014-06-17	Finance



Python - Reading Specific Rows

- Fungsi `read_csv` dari Pandas juga dapat digunakan untuk membaca beberapa baris spesifik untuk kolom yang diberikan
- Hasil dari fungsi `read_csv` menggunakan kode yang ditunjukkan disamping untuk 5 baris pertama untuk kolom bernama gaji

```
▶ import pandas as pd
data = pd.read_csv('Documents/Data Diyana/input.csv')

# Slice the result for first 5 rows
print (data[0:5]['salary'])
```

0	623.30
1	515.20
2	611.00
3	729.00
4	843.25

Name: salary, dtype: float64



Python - Reading Specific Rows

- Fungsi `read_csv` juga dapat digunakan untuk membaca beberapa baris spesifik untuk kolom yang diberikan.
- Hasil dari fungsi `read_csv` menggunakan kode yang ditunjukkan disamping untuk 5 baris pertama untuk kolom bernama gaji

```
▶ import pandas as pd
data = pd.read_csv('Documents/Data Diyana/input.csv')

# Slice the result for first 5 rows
print (data[0:5]['salary'])
```

0	623.30
1	515.20
2	611.00
3	729.00
4	843.25

Name: salary, dtype: float64



Python - Reading Specific Columns

- Fungsi `read_csv` juga dapat digunakan untuk membaca beberapa kolom tertentu
- Menggunakan metode multi-axes indexing yang disebut `.loc()` untuk tujuan ini
- Memilih untuk menampilkan kolom gaji dan nama untuk semua baris

```
▶ import pandas as pd
data = pd.read_csv('Documents/Data Diyana/input.csv')

# Use the multi-axes indexing function
print (data.loc[:,['salary', 'name']])
```

	salary	name
0	623.30	Rick
1	515.20	Dan
2	611.00	Tusar
3	729.00	Ryan
4	843.25	Gary
5	578.00	Rasmi
6	632.80	Pranab
7	722.50	Guru



Python - Reading Specific Columns & Rows

- Fungsi `read_csv` juga dapat digunakan untuk membaca beberapa kolom dan baris tertentu
- Menggunakan metode multi-axes indexing yang disebut `.loc()` untuk tujuan ini
- Memilih untuk menampilkan kolom gaji dan nama untuk beberapa baris

TERBUKA
UNTUK

```
import pandas as pd
data = pd.read_csv('Documents/Data Diyana/input.csv')

# Use the multi-axes indexing function
print (data.loc[[1,3,5],['salary','name']])
```

	salary	name
1	515.2	Dan
3	729.0	Ryan
5	578.0	Rasmi



Python - Reading Specific Columns & Rows

- Fungsi `read_csv` juga dapat digunakan untuk membaca beberapa kolom dan baris tertentu
- Menggunakan metode multi-axes indexing yang disebut `.loc()` untuk tujuan ini
- Memilih untuk menampilkan kolom gaji dan nama untuk beberapa baris

TERBUKA
UNTUK

```
▶ import pandas as pd
data = pd.read_csv('Documents/Data Diyana/input.csv')

# Use the multi-axes indexing funtion
print (data.loc[2:6,['salary','name']])
```

	salary	name
2	611.00	Tusar
3	729.00	Ryan
4	843.25	Gary
5	578.00	Rasmi
6	632.80	Pranab

Python - *Processing JSON Data*

- File JSON menyimpan data sebagai teks dalam format yang dapat dibaca manusia
- JSON adalah singkatan dari JavaScript Object Notation
- Pandas dapat membaca file JSON menggunakan fungsi `read_json`

Python - *Processing JSON Data*

- Copy file di samping ke Notepad kemudian simpan dengan nama **input.json**

```
{  "ID":  
  ["1", "2", "3", "4", "5", "6", "7", "8" ],  
  "Name":  
  ["Rick", "Dan", "Michelle", "Ryan", "Ga  
ry", "Nina", "Simon", "Guru" ],  
  "Salary":  
  ["623.3", "515.2", "611", "729", "843.2  
5", "578", "632.8", "722.5" ],  
  "StartDate":  
  [ "1/1/2012", "9/23/2013", "11/15/201  
4", "5/11/2014", "3/27/2015", "5/21/20  
13", "7/30/2013", "6/17/2014"],  
  "Dept":  
  [ "IT", "Operations", "IT", "HR", "Fina  
nce", "IT", "Operations", "Finance"]}
```



DIGITAL
TALENT
SCHOLARSHIP

Python - *Reading a JSONFile*

- Fungsi `read_json` digunakan untuk membaca konten file JSON sebagai DataFrame

```
▶ import pandas as pd  
  
data = pd.read_json('Documents/Data Diyana/input.json')  
print (data)
```

	ID	Name	Salary	StartDate	Dept
0	1	Rick	623.30	1/1/2012	IT
1	2	Dan	515.20	9/23/2013	Operations
2	3	Michelle	611.00	11/15/2014	IT
3	4	Ryan	729.00	5/11/2014	HR
4	5	Gary	843.25	3/27/2015	Finance
5	6	Nina	578.00	5/21/2013	IT
6	7	Simon	632.80	7/30/2013	Operations
7	8	Guru	722.50	6/17/2014	Finance



Python - Reading Specific Columns and Rows

- Fungsi `read_json` juga dapat digunakan untuk membaca beberapa kolom dan baris tertentu
- Menggunakan metode multi-axes indexing yang disebut `.loc()` untuk tujuan ini
- Memilih untuk menampilkan kolom gaji dan nama untuk beberapa baris

TERBUKA

```
import pandas as pd
data = pd.read_json('Documents/Data Diyana/input.json')

# Use the multi-axes indexing function
print (data.loc[[1,3,5],['Salary', 'Name']])
```

	Salary	Name
1	515.2	Dan
3	729.0	Ryan
5	578.0	Nina





DIGITAL
TALENT
SCHOLARSHIP

Python - Reading a JSON File

- Kita juga dapat menggunakan fungsi `to_json` untuk membaca konten file JSON

TERBUKA

```
import pandas as pd
data = pd.read_json('Documents/Data Diyana/input.json')

print(data.to_json(orient='records', lines=True))
```

```
{"ID":1,"Name":"Rick","Salary":623.3,"StartDate":"1\1\2012","Dept":"IT"}
{"ID":2,"Name":"Dan","Salary":515.2,"StartDate":"9\23\2013","Dept":"Operations"}
{"ID":3,"Name":"Michelle","Salary":611.0,"StartDate":"11\15\2014","Dept":"IT"}
{"ID":4,"Name":"Ryan","Salary":729.0,"StartDate":"5\11\2014","Dept":"HR"}
{"ID":5,"Name":"Gary","Salary":843.25,"StartDate":"3\27\2015","Dept":"Finance"}
{"ID":6,"Name":"Nina","Salary":578.0,"StartDate":"5\21\2013","Dept":"IT"}
{"ID":7,"Name":"Simon","Salary":632.8,"StartDate":"7\30\2013","Dept":"Operations"}
{"ID":8,"Name":"Guru","Salary":722.5,"StartDate":"6\17\2014","Dept":"Finance"}
```



Python - *Processing Excel Data*

- Salin data file di bawah ini ke dalam Excel anda untuk di **sheet 1**, simpan dalam **input.xlsx**

id	name	salary	start_date	dept
1	Rick	623.3	01-01-12	IT
2	Dan	515.2	23-09-13	Operations
3	Tusar	611	15-11-14	IT
4	Ryan	729	11-05-14	HR
5	Gary	843.25	27-03-15	Finance
6	Rasmi	578	21-05-13	it
7	Pranab	632.8	30-07-13	Operations
8	Guru	722.5	17-06-14	Finance

Python - *Processing Excel Data*

- Salin data file di bawah ini ke dalam Excel Anda untuk di **sheet 2** (berada dalam Excel yang sama dengan data yang sebelumnya pada sheet 1) simpan dalam **input.xlsx**

id	name	zipcode
1	Rick	301224
2	Dan	341255
3	Tusar	297704
4	Ryan	216650
5	Gary	438700
6	Rasmi	665100
7	Pranab	341211
8	Guru	347480

Python - Reading an Excel File

- Fungsi `read_csv` digunakan untuk membaca konten file CSV sebagai DataFrame

```
import pandas as pd  
data = pd.read_excel('Documents/Data Diyana/input.xlsx')  
print (data)
```

	id	name	salary	start_date	dept
0	1	Rick	623.30	2012-01-01	IT
1	2	Dan	515.20	2013-09-23	Operations
2	3	Tusar	611.00	2014-11-15	IT
3	4	Ryan	729.00	2014-05-11	HR
4	5	Gary	843.25	2015-03-27	Finance
5	6	Rasmi	578.00	2013-05-21	it
6	7	Pranab	632.80	2013-07-30	Operations
7	8	Guru	722.50	2014-06-17	Finance



Python - Reading Specific Columns and Rows

- Mirip dengan apa yang telah kita lihat di bab sebelumnya untuk membaca file CSV, fungsi `read_excel` juga dapat digunakan untuk membaca beberapa kolom dan baris tertentu
- Menggunakan metode multi-axes indexing yang disebut `.loc()` untuk tujuan ini
- Memilih untuk menampilkan kolom gaji dan nama untuk beberapa baris

```
▶ import pandas as pd
data = pd.read_excel('Documents/Data Diyana/input.xlsx')

# Use the multi-axes indexing funtion
print (data.loc[[1,3,5],['salary','name']])
```

	salary	name
1	515.2	Dan
3	729.0	Ryan
5	578.0	Rasmi



Python - Reading Multiple Excel Sheets

- Beberapa *sheet* dengan format data yang berbeda juga dapat dibaca dengan menggunakan fungsi `read_excel` dengan bantuan kelas *wrapper* bernama `ExcelFile`
- Ini akan membaca beberapa *sheet* ke dalam memori hanya sekali
- Dalam contoh ini kita membaca `sheet1` dan `sheet2` menjadi dua frame data dan mencetaknya secara individual

```
import pandas as pd
with pd.ExcelFile('Documents/Data Diyana/input.xlsx') as xls:
    df1 = pd.read_excel(xls, 'Sheet1')
    df2 = pd.read_excel(xls, 'Sheet2')

print("****Result Sheet 1****")
print(df1[0:5]['salary'])
print("")
print("****Result Sheet 2****")
print(df2[0:5]['zipcode'])
```

****Result Sheet 1****

0	623.30
1	515.20
2	611.00
3	729.00
4	843.25

Name: salary, dtype: float64

****Result Sheet 2****

0	301224
1	341255
2	297704
3	216650
4	438700

Name: zipcode, dtype: int64

Python - *Reading HTML File*

- Dalam contoh di bawah ini kita membuat permintaan ke URL untuk di-load
- Kemudian gunakan parameter parser html untuk membaca seluruh file HTML
- Selanjutnya, kita mencetak beberapa baris pertama dari halaman HTML

```
import urllib2
from bs4 import BeautifulSoup

# Fetch the html file
response = urllib2.urlopen('http://tutorialspoint.com/python/python_overview.htm')
html_doc = response.read()

# Parse the html file
soup = BeautifulSoup(html_doc, 'html.parser')

# Format the parsed html file
strhtml = soup.prettify()

# Print the first few characters
print (strhtml[:225])
```

Hasil eksekusi

```
<!DOCTYPE html>
<!--[if IE 8]><html class="ie ie8"> <![endif]-->
<!--[if IE 9]><html class="ie ie9"> <![endif]-->
<!--[if gt IE 9]><!-->
<html>
<!--<![endif]-->
<head>
<!-- Basic -->
<meta charset="utf-8"/>
<title>
```

Python - *Extracting Tag Value*

- Kita dapat mengekstraksi nilai tag dari instance pertama tag menggunakan kode berikut

```
import urllib2
from bs4 import BeautifulSoup

response = urllib2.urlopen('http://tutorialspoint.com/python/python_overview.htm')
html_doc = response.read()

soup = BeautifulSoup(html_doc, 'html.parser')

print (soup.title)
print(soup.title.string)
print(soup.a.string)
print(soup.b.string)
```

Hasil eksekusi

```
Python Overview
None
Python is Interpreted
```

Python - *Extracting All Tags*

```
import urllib2
from bs4 import BeautifulSoup

response = urllib2.urlopen('http://tutorialspoint.com/python/python_overview.htm')
html_doc = response.read()
soup = BeautifulSoup(html_doc, 'html.parser')

for x in soup.find_all('b'): print(x.string)
```

- Kita dapat mengekstraksi nilai tag dari semua instance tag menggunakan kode berikut

Hasil eksekusi

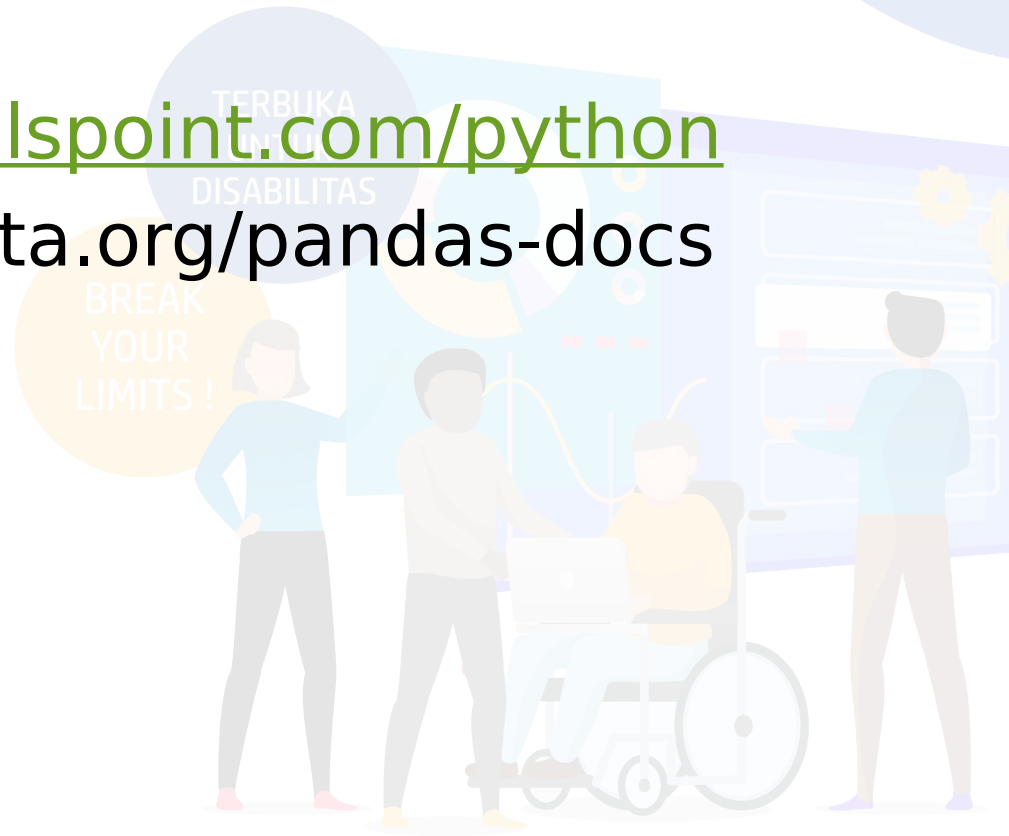
```
Python is Interpreted
Python is Interactive
Python is Object-Oriented
Python is a Beginner's Language
Easy-to-learn
Easy-to-read
Easy-to-maintain
A broad standard library
Interactive Mode
Portable
Extendable
Databases
GUI Programming
Scalable
```



DIGITAL
TALENT
SCHOLARSHIP

Referensi

- <https://www.tutorialspoint.com/python>
- <http://pandas.pydata.org/pandas-docs>








DIGITAL
TALENT
SCHOLARSHIP

IKUTI KAMI



DIGITAL
TALENT
SCHOLARSHIP

-  digitalent.kominfo
-  digitalent.kominfo
-  DTS_kominfo
-  Digital Talent Scholarship 2019

Pusat Pengembangan Profesi dan Sertifikasi
Badan Penelitian dan Pengembangan SDM
Kementerian Komunikasi dan Informatika
Jl. Medan Merdeka Barat No. 9
(Gd. Belakang Lt. 4 - 5)
Jakarta Pusat, 10110



digitalent.kominfo.go.id



digitalent.kominfo.go.id