ANALISIS BIG DATA

WEB SCRAPING

DISUSUN OLEH:

FIQQI AHLUDZIKRI (2111010034)

DOSEN:

Dr. M. SAID HASIBUAN

FAKULTAS ILMU KOMPUTER

PROGRAM STUDI TEKNIK INFORMATIKA

INSTITUT INFORMATIKA DAN BISNIS DARMAJAYA

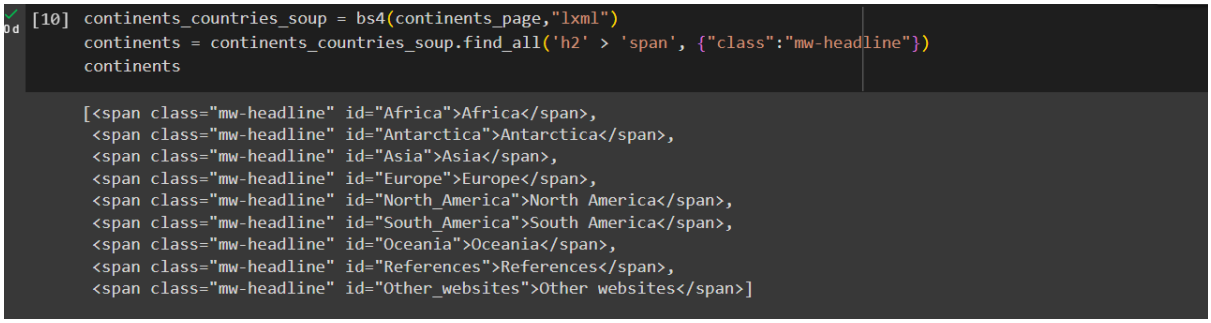
BANDAR LAMPUNG

TAHUN AJARAN 2023/2024

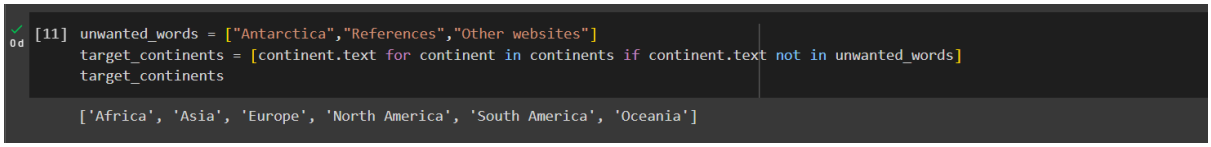
MENCARI TINGKAT KEBAHAGIAAN DI SETIAP NEGARA

Keterangan: Import beberapa library, requests untuk akses website yang akan diambil data, pandas untuk mengolah data, seaborn dan maplotlib untuk visualisasi data, dan bautifulsoup untuk memudahkan proses scraping.

Requests.get : untuk mengambil data dari website wikipedia



Keterangan: memfilter nama benua dari website denagan beatifulsoup



Keterangan: unwanted_word: untuk menampung kata-kata yang tidak relevan dan variable continents untuk menampung nama-nama benua yang diperlukan.

```
ol_html = continents_countries_soup.find_all('ol')
all_countries = [countries.find_all('li',{"class": None, "id": None}) for countries in ol_html]
all_countries
```

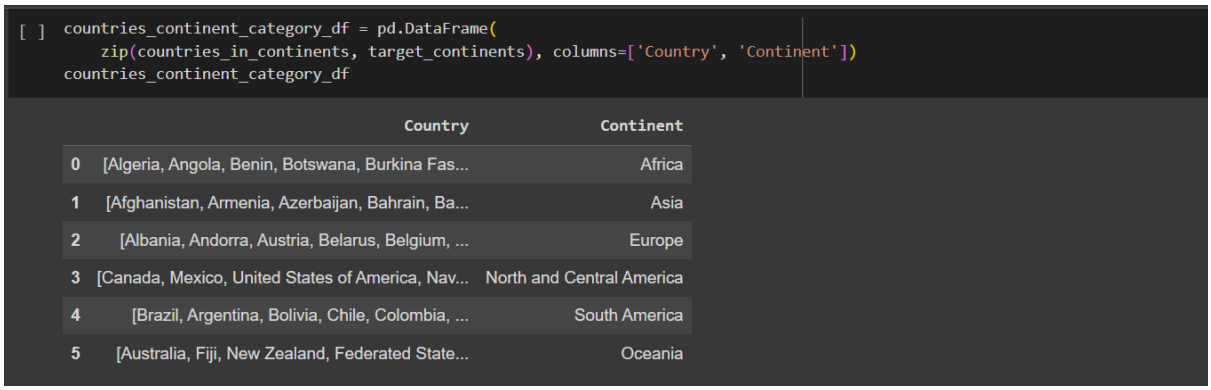
```
[[<li><a href="/wiki/Algeria" title="Algeria">Algeria</a> - <a href="/wiki/Algiers" title="Algiers">Algiers</a></li>,
  <li><a href="/wiki/Angola" title="Angola">Angola</a> - <a href="/wiki/Luanda" title="Luanda">Luanda</a></li>,
  <li><a href="/wiki/Benin" title="Benin">Benin</a> - <a class="mw-redirect" href="/wiki/Porto_Novo" title="Porto Novo">Porto Novo</a>,
 <a href="/wiki/Cotonou" title="Cotonou">Cotonou</a></li>,
  <li><a href="/wiki/Botswana" title="Botswana">Botswana</a> - <a href="/wiki/Gaborone" title="Gaborone">Gaborone</a></li>,
  <li><a href="/wiki/Burkina_Faso" title="Burkina Faso">Burkina Faso</a> - <a href="/wiki/Ouagadougou"
 title="Ouagadougou">Ouagadougou</a></li>,
  <li><a href="/wiki/Burundi" title="Burundi">Burundi</a> - <a href="/wiki/Gitega" title="Gitega">Gitega</a></li>,
  <li><a href="/wiki/Cameroon" title="Cameroon">Cameroon</a> (also spelled Cameroun) - <a href="/wiki/Yaound%C3%A9"
 title="Yaoundé">Yaoundé</a></li>,
  <li><a href="/wiki/Cape_Verde" title="Cape Verde">Cape Verde</a> - <a class="mw-redirect" href="/wiki/Praia,_Cape_Verde" title="Praia,
```

Keterangan: Mengambil nama-nama negara dari halaman website dengan memfilter ol_html dan li

```
countries_in_continents = []
for items in all_countries:
    countries = []
    if items:
        for country in items:
            countries = [country.find('a').text for country in items if country.find('a')]
        countries_in_continents.append(countries)
countries_in_continents
```

```
[['Algeria',
  'Angola',
  'Benin',
  'Botswana',
  'Burkina Faso',
  'Burundi',
  'Cameroon',
  'Cape Verde',
  'Central African Republic',
  'Chad',
  'Comoros',
  'Republic of the Congo',
```

Keterangan: memfilter nama negara dengan membuat loop disetiap item pada variable all_countries, variable countries berisi nama-nama negara di benua yang sama, di dalam items ada daftar element html yang akan difilter. Setelah di filter akan masuk dalam variable countries_in_continents.



```
countries_continent_category_df = pd.DataFrame(
    zip(countries_in_continents, target_continents), columns=['Country', 'Continent'])
countries_continent_category_df
```

	Country	Continent
0	[Algeria, Angola, Benin, Botswana, Burkina Fas...	Africa
1	[Afghanistan, Armenia, Azerbaijan, Bahrain, Ba...	Asia
2	[Albania, Andorra, Austria, Belarus, Belgium, ...	Europe
3	[Canada, Mexico, United States of America, Nav...	North and Central America
4	[Brazil, Argentina, Bolivia, Chile, Colombia, ...	South America
5	[Australia, Fiji, New Zealand, Federated State...	Oceania

Keterangan: menampilkan data dengan print data frame.

```
countries_continent_category_df = countries_continent_category_df.explode(
    'Country').reset_index(drop=True)
countries_continent_category_df
```

	Country	Continent
0	Algeria	Africa
1	Angola	Africa
2	Benin	Africa
3	Botswana	Africa
4	Burkina Faso	Africa
...	...	...
197	Samoa	Oceania
198	Solomon Islands	Oceania
199	Tonga	Oceania
200	Tuvalu	Oceania
201	Vanuatu	Oceania

202 rows × 2 columns

Keterangan: menggunakan metode explode untuk memisahkan setiap negara dengan baris yang berbeda.

```
countries_score_page = requests.get("https://en.wikipedia.org/wiki/World_Happiness_Report#2020_report")
countries_score_soup = bs4(countries_score_page.content,'lxml')
```

```
countries_score_table = countries_score_soup.find('table', {'class':'wikitable'})
countries_score_table
```

```
<table class="wikitable sortable">
<tbody><tr valign="top">
<th style="width: 10px;">Overall rank
</th>
<th style="width: 250px;">Country or region
</th>
<th><abbr title="Happiness score">Score</abbr>
</th>
<th style="width: 10px;"><abbr title="Explained by: GDP">GDP per capita</abbr>
</th>
<th style="width: 10px;"><abbr title="Explained by: Social support">Social support</abbr>
</th>
<th style="width: 10px;"><abbr title="Explained by: Healthy life expectancy">Healthy life expectancy</abbr>
</th>
<th style="width: 10px;"><abbr title="Explained by: Freedom to make life choices">Freedom to make life choices</abbr>
</th>
<th style="width: 10px;"><abbr title="Explained by: Generosity">Generosity</abbr>
</th>
<th style="width: 10px;"><abbr title="Explained by: Perceptions of corruption">Perceptions of corruption</abbr>
</th></tr>
<tr>
```

Keterangan: Mengambil data tingkat kebahagiaan tiap negara.

```
countries_score_df = pd.read_html(str(countries_score_table))
countries_score_df
```

```
[     Overall rank         Country or region  Score  GDP per capita  \
0               1                   Finland  7.809           1.285
1               2                   Denmark  7.646           1.327
2               3               Switzerland  7.560           1.391
3               4                   Iceland  7.504           1.327
4               5                    Norway  7.488           1.424
..            ...                       ...    ...             ...
148           149  Central African Republic  3.476           0.041
149           150                    Rwanda  3.312           0.343
150           151                  Zimbabwe  3.299           0.426
151           152              South Sudan  2.817           0.289
152           153               Afghanistan  2.567           0.301

      Social support  Healthy life expectancy  Freedom to make life choices  \
0              1.500                    0.961                         0.662
1              1.503                    0.979                         0.665
2              1.472                    1.041                         0.629
3              1.548                    1.001                         0.662
4              1.495                    1.008                         0.670
..               ...                      ...                           ...
148            0.000                    0.000                         0.293
149            0.523                    0.572                         0.604
150            1.048                    0.375                         0.377
151            0.553                    0.209                         0.066
152            0.356                    0.266                         0.000

      Generosity  Perceptions of corruption
0          0.160                      0.478
1          0.243                      0.495
```

Keterangan: Memproses data menggunakan metode pandas read_html.

```
[ ] countries_score_df = countries_score_df[0]
    countries_score_df = countries_score_df.rename(columns={"Country or region":"Country"})
    countries_score_df
```

	Overall rank	Country	Score	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption
0	1	Finland	7.809	1.285	1.500	0.961	0.662	0.160	0.478
1	2	Denmark	7.646	1.327	1.503	0.979	0.665	0.243	0.495
2	3	Switzerland	7.560	1.391	1.472	1.041	0.629	0.269	0.408
3	4	Iceland	7.504	1.327	1.548	1.001	0.662	0.362	0.145
4	5	Norway	7.488	1.424	1.495	1.008	0.670	0.288	0.434
...	...	...	...	...	...	...	...	...	...
148	149	Central African Republic	3.476	0.041	0.000	0.000	0.293	0.254	0.028
149	150	Rwanda	3.312	0.343	0.523	0.572	0.604	0.236	0.486
150	151	Zimbabwe	3.299	0.426	1.048	0.375	0.377	0.151	0.081
151	152	South Sudan	2.817	0.289	0.553	0.209	0.066	0.210	0.111
152	153	Afghanistan	2.567	0.301	0.356	0.266	0.000	0.135	0.001

153 rows × 9 columns

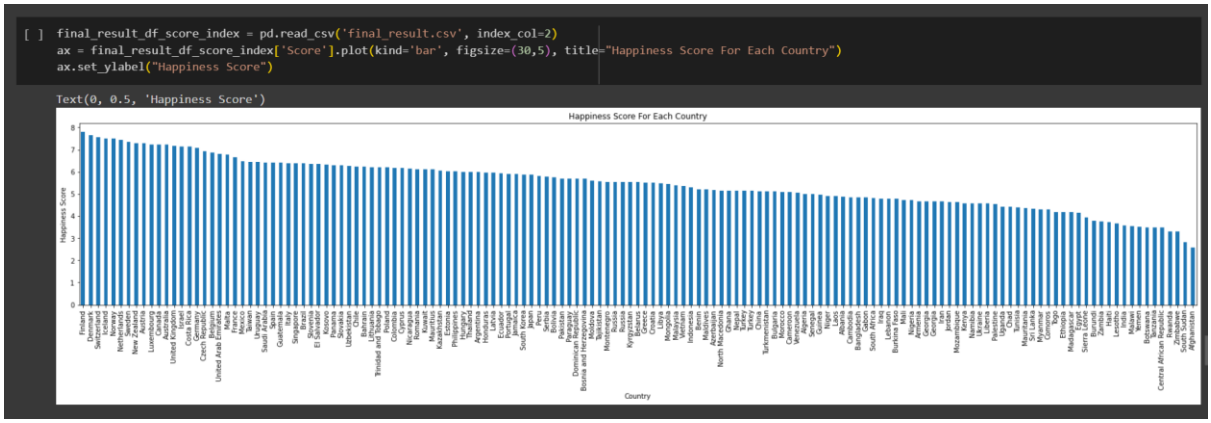
Keterangan: Mengakses data frame dan mengganti nama kolom country or region menjadi country.

```
[ ] merged_df = pd.merge(countries_score_df, countries_continent_category_df, how='inner', on='Country')
    merged_df.to_csv('final_result.csv')
    merged_df
```

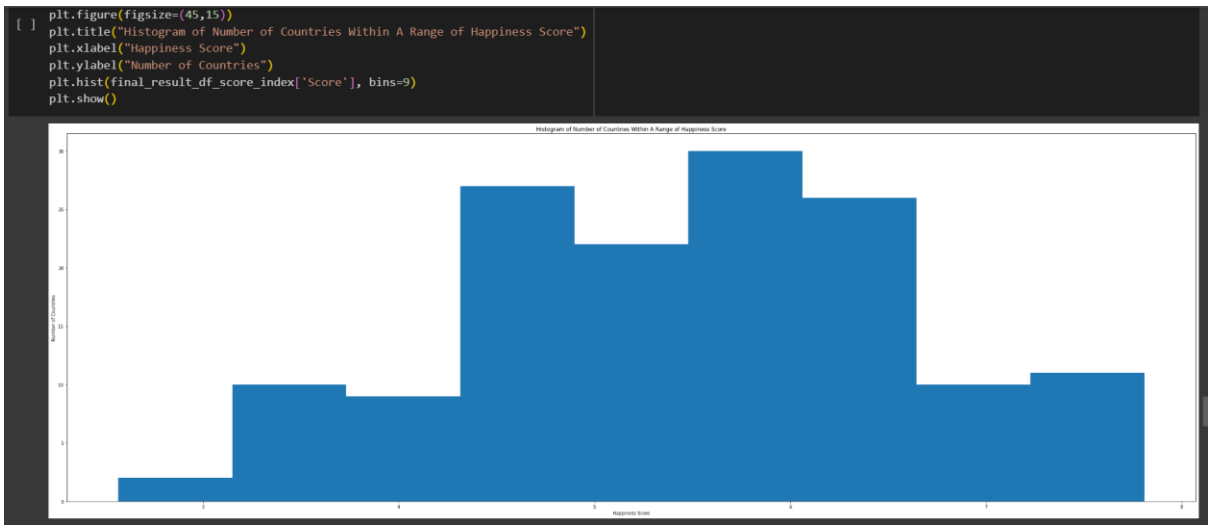
	Overall rank	Country	Score	GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption	Continent
0	1	Finland	7.809	1.285	1.500	0.961	0.662	0.160	0.478	Europe
1	2	Denmark	7.646	1.327	1.503	0.979	0.665	0.243	0.495	Europe
2	3	Switzerland	7.560	1.391	1.472	1.041	0.629	0.269	0.408	Europe
3	4	Iceland	7.504	1.327	1.548	1.001	0.662	0.362	0.145	Europe
4	5	Norway	7.488	1.424	1.495	1.008	0.670	0.288	0.434	Europe
...	...	...	...	...	...	...	...	...	...	...
142	149	Central African Republic	3.476	0.041	0.000	0.000	0.293	0.254	0.028	Africa
143	150	Rwanda	3.312	0.343	0.523	0.572	0.604	0.236	0.486	Africa
144	151	Zimbabwe	3.299	0.426	1.048	0.375	0.377	0.151	0.081	Africa
145	152	South Sudan	2.817	0.289	0.553	0.209	0.066	0.210	0.111	Africa
146	153	Afghanistan	2.567	0.301	0.356	0.266	0.000	0.135	0.001	Asia

147 rows × 10 columns

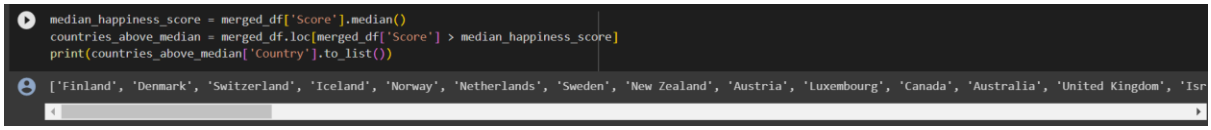
Keterangan: Menggabungkan daftar negara dan benua dengan data frame table menggunaman fungsi merge_df dan simpan data di file final_result.csv



Keterangan: Ambil data dari file final_result.csv dengan pd_read_csv dan membuat barcode sederhana.

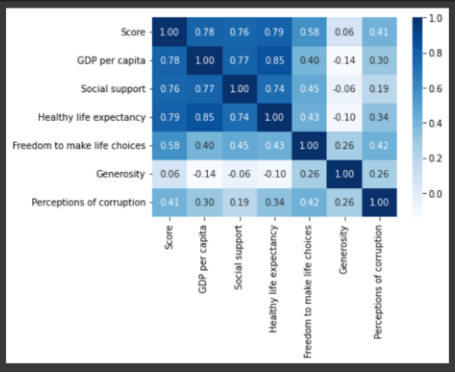


Keterangan: Membuat histogram menggunakan matplot lib.



Keterangan: Mencari negara dengan score lebih tinggi dari nilai median.

```
[ ] heatmap_df = merged_df.drop(['Overall rank','Country','Continent'], axis=1)
    ax = sns.heatmap(heatmap_df.corr(), annot=True, fmt='.2f', cmap='Blues')
```



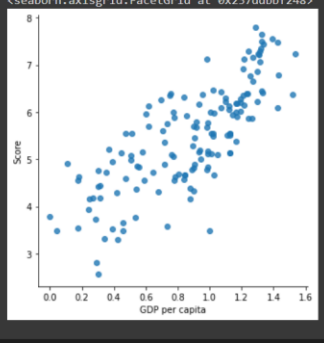
Keterangan: Mencari korelasi antara score kebahagiaan dengan faktor-faktor lain dengan heatmap.

Hasil: Semakin pekat warna biru maka semakin semakin besar korelasi antara faktor-faktor tersebut.
Hipotesa: Tingkat kebahagiaan seseorang atau sebuah negara punya korelasi yang relatif tinggi dengan faktor GDP per capita, Social support dan Healthty life expectancy.

Apakah kaya = Bahagia?

```
[ ] sns.lmplot(x='GDP per capita', y='Score', data=merged_df, fit_reg=False)
```

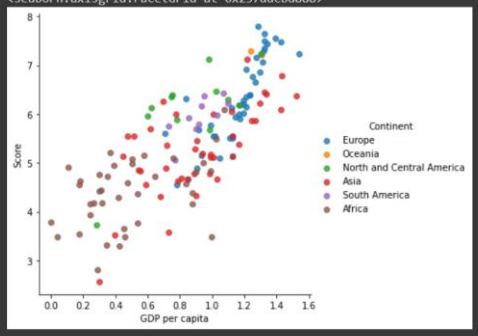
```
<seaborn.axisgrid.FacetGrid at 0x237ddbbf248>
```



Keterangan: Membuat scaterplot negara dengan GDP per capita sangat tinggi tetapi score kebahagiaan lebih rendah daripada GDP per capita rendah.
Kesimpulan: Uang atau aset yang lebih banyak belum tentu bisa membuat orang lebih bahagia.

```
sns.lmplot(x='GDP per capita', y='Score', data=merged_df, fit_reg=False, hue='Continent')
```

```
<seaborn.axisgrid.FacetGrid at 0x237ddcbd888>
```



Keterangan: Membedakan warna titik berdasarkan benua.