

# **LAPORAN TUGAS BESAR**

## **PENGENALAN KOMPUTASI KU1102**

### **Analisis Data**

Diajukan untuk memenuhi Tugas Besar 2 Mata Kuliah Pengenalan Komputasi yang diampu oleh:

Elvayandri, S.Si., M.T.



Institut Teknologi Bandung

Sekolah Teknik Elektro dan Informatika

Jalan Ganesa no. 10, Lb. Siliwangi, Kecamatan Coblong, Kota Bandung, Jawa Barat 40132

## **Anggota Kelompok 5**



**DEWANTORO TRIATMOJO**  
19622152

**BERTO TOGATOROP**  
19622192



**RANDY VERDIAN**  
19622202

**AIRA ARDISTYA A**  
16522062

## DAFTAR ISI

<b>DAFTAR ISI.....</b>	<b>i</b>
<b>PRAKATA.....</b>	<b>ii</b>
<b>PENDAHULUAN.....</b>	<b>iii</b>
<b>Deskripsi Data dan File.....</b>	<b>1</b>
<b>Deskripsi Karakteristik Data.....</b>	<b>3</b>
<b>Data Preprocessing dan Data Cleansing.....</b>	<b>13</b>
<b>Statistik.....</b>	<b>25</b>
<b>Visualisasi.....</b>	<b>35</b>
<b>Korelasi.....</b>	<b>45</b>
<b>Kesimpulan.....</b>	<b>50</b>
<b>Daftar Pustaka.....</b>	<b>51</b>
<b>Pembagian Tugas.....</b>	<b>52</b>

## **PRAKATA**

Puji dan syukur kami panjatkan kehadirat Tuhan Yang Maha Esa, yang telah melimpahkan rahmat dan karunianya sehingga laporan Tugas Besar Pengenalan Komputasi ini dapat tersusun hingga tuntas. Laporan Tugas Besar Pengenalan Komputasi yang telah kami susun berisi tentang program Data Analisis mengenai QS Ranking Universitas Dunia.

Kami juga mengucapkan terimakasih yang sebanyak-banyaknya kepada seluruh pihak yang senantiasa mendukung serta membantu selama proses penggeraan Tugas Besar Komputasi khususnya kepada Ibu Elvayandri, S.Si., M.T. selaku dosen mata kuliah Pengenalan Komputasi yang telah memberikan banyak ilmu pengetahuan dan bimbingan kepada kami hingga selesaiya tugas besar ini.

Dalam penyusunan laporan Tugas Besar ini, kami menyadari bahwa laporan dan program yang telah kami buat masih jauh dari kata sempurna baik segi bahasa, penyusunan, maupun penulisannya. Oleh karena itu, kritik dan saran sangat kami nantikan guna menjadi acuan agar kami bisa menjadi lebih baik lagi di masa mendatang.

Semoga laporan Tugas Besar Pengenalan Komputasi ini bisa menambah wawasan dan bermanfaat untuk kelompok kami , teman-teman, dan para pembaca.

Bandung, 30 November 2022

Kelompok 5 Pengenalan Komputasi Kelas 16

## PENDAHULUAN

Universitas adalah suatu institusi pendidikan tinggi dan penelitian, yang memberikan gelar akademis dalam berbagai bidang. Sebuah universitas menyediakan pendidikan sarjana dan pascasarjana. Universitas merupakan tempat untuk para mahasiswa untuk mempersiapkan diri di dunia kerja nanti.

Calon mahasiswa di seluruh dunia pasti ingin kuliah di universitas yang memadai dan mendukung perkembangannya. Apalagi, jika mereka berkuliahan di universitas top dunia yang merupakan suatu kebanggan dan keuntungan bagi diri mereka sendiri. Untuk mengetahui universitas yang cocok dengannya, mahasiswa dapat melihat berdasarkan peringkat universitas yang dituju.

*Quacquarelli Symonds* (QS) adalah salah satu lembaga pemeringkatan yang mengeluarkan publikasi tahunan peringkat universitas yang bernama *QS World University Ranking*. Pemeringkatan ini bertujuan untuk membantu calon siswa mengidentifikasi universitas terkemuka dunia di bidang pilihan mereka sebagai tanggapan atas tingginya permintaan untuk perbandingan tingkat mata pelajaran.

Pada Tugas Besar 2 Pengenalan Komputasi ini, kami ingin menganalisis dan memvisualisasikan data dari QS Ranking Universitas Top 400 besar dari tahun 2017 hingga 2022. Dengan demikian, kita bisa mendapatkan pengetahuan atau insight dari data ranking universitas dunia.

## Deskripsi Data dan File

- Data yang kami pilih adalah data QS Ranking Universitas Dunia dari tahun 2017 sampai tahun 2022. Kami ambil data hanya untuk 400 besar dunia (setelah dibersihkan)
- Pengetahuan yang ingin kami ketahui dari data yang kami ambil adalah
  1. Di benua mana yang universitas top 400nya paling banyak?
  2. Di negara mana yang universitas top 400nya paling banyak?
  3. Di benua mana yang paling disukai oleh pelajar internasional?
  4. Di negara mana yang paling disukai oleh pelajar internasional?
  5. Apakah universitas top 400 kebanyakan umum atau swasta?
  6. Bagaimana hubungan jumlah pelajar internasional terhadap ranking universitas?
  7. Bagaimana hubungan jumlah staff/pengajar terhadap jumlah murid international di suatu universitas?
  8. Bagaimana hubungan jumlah staff/pengajar terhadap ranking suatu universitas?
- Untuk meloading data dan mengetahui berbagai informasi terkait data, kami menggunakan library Pandas Python. Lampiran notebook yang berisi analisis kita pada [link berikut](#)

```
import pandas as pd
import matplotlib.pyplot as plt
# Import Data dari CSV
# Data frame kotor yang tidak diubah (untuk karakteristik data & menunjukkan kekotoran di data cleansing)
df_uncleaned = pd.read_csv('qs-world-university-rankings-2017-to-2022-V2.csv')
# Data frame kotor yang akan diubah program menjadi bersih (untuk demonstrasi pembersihan)
df_chg_uncleaned = pd.read_csv('qs-world-university-rankings-2017-to-2022-V2.csv')
# Data frame data bersih
df = pd.read_csv('qs_ranking_400_cleaned.csv')
```

- Deskripsi:
  1. Data QS Ranking Universitas Dunia dari tahun 2017 sampai tahun 2022 kami ambil dari dataset kaggle pada [link berikut](#).
  2. Dari sumber, format data ini adalah comma seperated value (CSV). Ukuran file dari data ini sebesar 1,84 MB.

3. Data kotor memiliki dimensi sebesar 6482 baris, 15 kolom, dan jumlah elemen data sebesar 97230.
4. Data bersih memiliki dimensi sebesar 2390 baris, 13 kolom, dan jumlah elemen data sebesar 31070.

```
▶ # Cara Mengetahui
# Data kotor
print(f"Jumlah baris data kotor: {len(df_uncleaned)}")
print(f"Jumlah kolom data kotor: {len(df_uncleaned.columns)}")
print(f"Jumlah elemen data kotor: {df_uncleaned.size}")

# Data bersih
print(f"Jumlah baris data bersih: {len(df)}")
print(f"Jumlah kolom data bersih: {len(df.columns)}")
print(f"Jumlah elemen data bersih: {df.size}")

● Jumlah baris data kotor: 6482
Jumlah kolom data kotor: 15
Jumlah elemen data kotor: 97230
Jumlah baris data bersih: 2390
Jumlah kolom data bersih: 13
Jumlah elemen data bersih: 31070
```

## Deskripsi Karakteristik Data

Pada data ini terdapat 15 atribut yaitu university, year, rank\_display, score, link, country, city, region, logo, type, research\_output, student\_faculty\_ratio, international\_students, size, dan faculty\_count.

### 1. Atribut university

- Atribut ini berisi nama dari universitas di ranking dunia
- Jenis atribut ini adalah kategorical yang nominal
- Atribut ini bernilai nama universitas seperti "Massachusetts Institute of Technology (MIT)", "Stanford University", dll

```
[ ] # Nilai dari atribut
print(f"Contoh nilai atribut: {df_uncleaned['university'].unique()}")  
  
Contoh nilai atribut: ['Massachusetts Institute of Technology (MIT)' 'Stanford University'  
'Harvard University' ... 'Université de Tunis'  
'Université de Tunis El Manar' 'Zagazig University']
```

- Persentase data kosong adalah 0%

```
[ ] # Data kosong
print(f"Persentase data kosong: {df_uncleaned['university'].isnull().sum()*100/len(df_uncleaned['university']):.2f}%")  
  
Persentase data kosong: 0.00%
```

### 2. Atribut year

- Atribut ini berisi tahun data tersebut dicatat
- Jenis atribut ini adalah numerical yang diskrit
- Atribut ini bernilai 2017, 2018, 2019, 2020, 2021, dan 2022.

```
[ ] # Nilai dari atribut
print(f"Contoh nilai atribut: {df_uncleaned['year'].unique()}")  
  
Contoh nilai atribut: [2017 2018 2019 2020 2021 2022]
```

- Atribut ini memiliki range data 2017 sampai dengan 2022

```
[7] # Range
print(f"Range: {df_uncleaned['year'].min()}-{df_uncleaned['year'].max()}")

Range: 2017-2022
```

- Persentase data kosong adalah 0.00%

```
[ ] # Data kosong
print(f"Persentase data kosong: {df_uncleaned['year'].isnull().sum()*100/len(df_uncleaned['year']):.2f}%")

Persentase data kosong: 0.00%
```

### 3. Atribut rank\_display

- Atribut ini berisi ranking dari universitas
- Jenis atribut ini adalah numerical yang diskrit
- Atribut ini bernilai 1, 2,..., 1201.

```
[ ] # Nilai dari atribut
print(f"Contoh nilai atribut: {df_uncleaned['rank_display'].unique()}")

Contoh nilai atribut: [1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
17 18 19 20 21 22 23 24 26 27 28 29 30 31
32 33 34 35 36 37 39 40 41 42 43 44 45 46
49 51 53 55 56 57 58 59 60 61 62 63 65 66
67 68 71 72 73 74 75 77 78 79 80 81 82 83
84 85 87 88 89 90 91 92 93 94 95 97 98 102
104 106 108 109 110 111 112 113 115 117 118 119]
```

- Atribut ini memiliki range dari 1 sampai 1201

```
[8] # Range
cek_range_rank = df_uncleaned.loc[(df_uncleaned['rank_display'].isna() == False) & (df_uncleaned['rank_display'].str.isnumeric() == True)][['rank_display']].apply(lambda x: int(x))

Range data: 1-1201
```

- Persentase data kosong adalah 1.05%

```
[ ] # Data kosong
print(f"Persentase data kosong: {df_uncleaned['rank_display'].isnull().sum()*100/len(df_uncleaned['rank_display']):.2f}%")
Persentase data kosong: 1.05%
```

#### 4. Atribut score

- Atribut ini berisi nilai/penilaian universitas
- Jenis atribut ini adalah numerical yang kontinu

```
▶ # Nilai dari atribut
print(f"Contoh nilai atribut: {df_uncleaned['score'].unique()}")
Contoh nilai atribut: [100.  98.7  98.3  97.2  96.9  96.8  95.6  94.2  94.1  93.   92.8  91.5
 91.4  91.1  90.9  90.1  89.3  89.   88.9  88.6  87.7  87.3  86.4  86.
 85.7  85.4  85.2  84.8  84.1  84.   83.8  79.9  82.6  82.1  81.8  81.7
 81.3  80.6  79.6  79.5  79.4  78.1  78.   77.9  77.8  77.6  75.7  69.9
 75.4  74.9  74.3  74.2  73.9  72.4  72.2  72.1  71.7  71.6  71.4  70.7
 70.2  69.8  69.7  69.1  68.8  68.7  68.2  68.1  67.9  67.8  67.3  67.2
 67.   66.9  65.8  65.6  65.4  65.2  65.   64.7  64.5  64.2  64.   63.6
 63.1  62.8  62.7  62.4  61.5  62.2  62.1  61.8  61.6  61.4  60.5  60.]
```

- Atribut ini memiliki nilai dari 23.5 sampai dengan 100.0

```
[12] # Range
cek_range_score = df_uncleaned['score']
print(f"Range data: {df_uncleaned['score'].min()}-{df_uncleaned['score'].max()}")
Range data: 23.5-100.0
```

- Persentase data kosong adalah 56.49%

```
[ ] # Data kosong
print(f"Persentase data kosong: {df_uncleaned['score'].isnull().sum()*100/len(df_uncleaned['score']):.2f}%")
Persentase data kosong: 56.49%
```

## 5. Atribut link

- Atribut ini berisi link menuju universitas tersebut pada website [www.topuniversities.com](http://www.topuniversities.com)
- Jenis atribut ini adalah kategorical yang nominal
- Atribut ini memiliki nilai seperti ["https://www.topuniversities.com/universities/massachusetts-institute-technology-mit"](https://www.topuniversities.com/universities/massachusetts-institute-technology-mit), dll

```
[ ] # Nilai dari atribut
print(f"Contoh nilai atribut: {df_uncleaned['link'].unique()}")

Contoh nilai atribut: ['https://www.topuniversities.com/universities/massachusetts-institute-technology-mit'
'https://www.topuniversities.com/universities/stanford-university'
'https://www.topuniversities.com/universities/harvard-university' ...
'https://www.topuniversities.com/universities/universite-de-tunis'
'https://www.topuniversities.com/universities/universite-de-tunis-el-manar'
'https://www.topuniversities.com/universities/zagazig-university']
```

- Persentase data kosong adalah 0.00%

```
[ ] # Data kosong
print(f"Persentase data kosong: {df_uncleaned['link'].isnull().sum()*100/len(df_uncleaned['link']):.2f}%")

Persentase data kosong: 0.00%
```

## 6. Atribut country

- Atribut ini berisi asal negara dari universitas
- Jenis atribut ini adalah kategorical yang nominal
- Atribut ini memiliki nilai seperti "United States", "United Kingdom", "Switzerland", "Singapore", dll

```
[ ] # Nilai dari atribut
print(f"Contoh nilai atribut: {df_uncleaned['country'].unique()}")
```

```
Contoh nilai atribut: ['United States' 'United Kingdom' 'Switzerland' 'Singapore' 'Australia'
'China (Mainland)' 'Hong Kong SAR' 'Canada' 'France' 'Japan'
'South Korea' 'Netherlands' 'Germany' 'Taiwan' 'Denmark' 'Sweden'
'Belgium' 'New Zealand' 'Argentina' 'Finland' 'Ireland' 'Russia' 'Norway'
'Brazil' 'Mexico' 'Malaysia' 'Chile' 'Israel' 'India' 'Austria' 'Spain'
'Italia' 'Saudi Arabia' 'South Africa' 'Lebanon' 'Kazakhstan' 'Thailand'
```

- Persentase data kosong adalah 0.0%

```
[ ] # Data kosong
print(f"Persentase data kosong: {df_uncleaned['country'].isnull().sum()*100/len(df_uncleaned['country']):.2f}%")
```

```
Persentase data kosong: 0.00%
```

## 7. Atribut city

- Atribut ini berisi asal kota dari universitas
- Jenis atribut ini adalah kategorical yang nominal
- Atribut ini memiliki nilai seperti "Cambridge", "Stanford", "Pasadena", "Oxford", "London", "Chicago", dll

```
[ ] # Nilai dari atribut
print(f"Contoh nilai atribut: {df_uncleaned['city'].unique()}")
```

```
Contoh nilai atribut: ['Cambridge' 'Stanford' 'Pasadena' 'Oxford' 'London' 'Zürich' 'Chicago'
'Princeton' 'Singapore' 'Lausanne' 'New Haven' 'Ithaca' 'Baltimore'
'Philadelphia' 'Edinburgh' 'New York City' 'Canberra' 'Ann Arbor'
'Beijing' 'Durham' 'Evanston' 'Hong Kong' 'Berkeley' 'Manchester'
'Montreal' 'Los Angeles' 'Toronto' 'Paris' 'Tokyo' 'Seoul' 'Kyoto'
'San Diego' 'Bristol' 'Parkville' 'Shanghai' 'Vancouver' 'Sydney'
```

- Persentase data kosong adalah 2.75%

```
[ ] # Data kosong
print(f"Persentase data kosong: {df_uncleaned['city'].isnull().sum()*100/len(df_uncleaned['city']):.2f}%")
```

```
Persentase data kosong: 2.75%
```

## 8. Atribut region

- Atribut ini berisi asal benua dari universitas
- Jenis atribut ini adalah kategorical yang nominal
- Atribut ini memiliki nilai seperti "North America", "Europe", "Asia", "Latin America", "Africa", dan "Oceania"

```
[ ] # Nilai dari atribut
print(f"Contoh nilai atribut: {df_uncleaned['region'].unique()}")  
Contoh nilai atribut: ['North America' 'Europe' 'Asia' 'Oceania' 'Latin America' 'Africa']
```

- Persentase data kosong adalah 0.00%

```
[ ] # Data kosong
print(f"Persentase data kosong: {df_uncleaned['region'].isnull().sum()*100/len(df_uncleaned['region']):.2f}%")  
Persentase data kosong: 0.00%
```

## 9. Atribut logo

- Atribut ini berisi link logo dari universitas pada website [www.topuniversities.com](http://www.topuniversities.com)
- Jenis atribut ini adalah kategorical yang nominal
- Atribut ini memiliki nilai seperti ["https://www.topuniversities.com/sites/default/files/massachusetts-institute-of-technology-mit\\_410\\_small.jpg"](https://www.topuniversities.com/sites/default/files/massachusetts-institute-of-technology-mit_410_small.jpg), dll

```
[ ] # Nilai dari atribut
print(f"Contoh nilai atribut: {df_uncleaned['logo'].unique()}")  
Contoh nilai atribut: ['https://www.topuniversities.com/sites/default/files/massachusetts-institute-of-technology-mit_410_small.jpg'
 'https://www.topuniversities.com/sites/default/files/stanford-university_573_small.jpg'
 'https://www.topuniversities.com/sites/default/files/harvard-university_253_small.jpg'
 ...
 'https://www.topuniversities.com/sites/default/files/universit-de-tunis_592560cf2aeae70239af5470_small.jpg'
 'https://www.topuniversities.com/sites/default/files/universit-de-tunis-el-manar_592560cf2aeae70239af5472_small.jpg'
 'https://www.topuniversities.com/sites/default/files/zagazig-university_592560cf2aeae70239af4f43_small.jpg']
```

- Persentase data kosong adalah 0.00%

```
[ ] # Data kosong
print(f"Persentase data kosong: {df_uncleaned['logo'].isnull().sum()*100/len(df_uncleaned['logo']):.2f}%")
Persentase data kosong: 0.00%
```

## 10. Atribut type

- Atribut ini berisi tipe public/private dari universitas
- Jenis atribut ini adalah kategorical yang binary
- Atribut ini memiliki nilai "Private" atau "Public"

```
[ ] # Nilai dari atribut
print(f"Contoh nilai atribut: {df_uncleaned['type'].unique()}")
# note: nan disini artinya not a number atau kosong.

Contoh nilai atribut: ['Private' 'Public' nan]
```

- Persentase data kosong adalah 0.19%

```
[ ] # Data kosong
print(f"Persentase data kosong: {df_uncleaned['type'].isnull().sum()*100/len(df_uncleaned['type']):.2f}%")
Persentase data kosong: 0.19%
```

## 11. Atribut research\_output

- Atribut ini berisi tingkat keaktifan riset dari universitas
- Jenis atribut ini adalah kategorical yang ordinal
- Atribut ini memiliki nilai 'Very High', 'Very high', 'High', 'Medium', 'Low'

```
[ ] # Nilai dari atribut
print(f"Contoh nilai atribut: {df_uncleaned['research_output'].unique()}")
# note: nan disini artinya not a number atau kosong.

Contoh nilai atribut: ['Very High' 'Very high' 'High' 'Medium' 'Low' nan]
```

- Persentase data kosong adalah 0.03%

```
[ ] # Data kosong
print(f"Percentase data kosong: {df_uncleaned['research_output'].isnull().sum()*100/len(df_uncleaned['research_output']):.2f}%")
Percentase data kosong: 0.03%
```

## 12. Atribut student\_faculty\_ratio

- Atribut ini berisi ratio dari jumlah mahasiswa per jumlah staff atau pengajar
- Jenis atribut ini adalah numerical yang diskrit

```
[ ] # Nilai dari atribut
print(f"Contoh nilai atribut: {df_uncleaned['student_faculty_ratio'].unique()}")
Contoh nilai atribut: [ 4.  3.  5.  2.  7.  6.  8.  9.  11.  10.  19.  13.  14.  12.  15.  17.  18.  16.
 25.  21.  nan 22.  24.  23.  20.  32.  29.  26.  35.  27.  28.  44.  31.  42.  30.  41.
 33.  45.  38.  40.  58.  1.  67.  36.]
```

- Atribut ini memiliki range dari 1.0 sampai dengan 67.0

```
[13] # Range
print(f"Range: {df_uncleaned['student_faculty_ratio'].min()}-{df_uncleaned['student_faculty_ratio'].max()}")
Range: 1.0-67.0
```

- Persentase data kosong adalah 1.16%

```
[ ] # Data kosong
print(f"Percentase data kosong: {df_uncleaned['student_faculty_ratio'].isnull().sum()*100/len(df_uncleaned['student_faculty_ratio']):.2f}%")
Percentase data kosong: 1.16%
```

### 13. Atribut international\_students

- Atribut ini berisi jumlah mahasiswa internasional yang mengikuti universitas tersebut
- Jenis atribut ini adalah numerical yang diskrit

```
[ ] # Mengoreksi nilai data (jika tidak, data akan dianggap python tidak masuk akal.)
cek_range_is = df_uncleaned['international_students'].apply(lambda x: str(x).replace(',', '')) # Step 2
cek_range_is = cek_range_is.apply(lambda x: str(x).replace('.', '')) # Step 3
cek_range_is = cek_range_is.apply(lambda x: float(x)) # Step 4
cek_range_is = cek_range_is.loc[cek_range_is.isna() == False].apply(lambda x: float(x))

# Nilai dari atribut
print(f"Contoh nilai atribut: {cek_range_is.unique()}")

Contoh nilai atribut: [3730. 3879. 5877. ... 157. 585. 2300.]
```

- Atribut ini memiliki range dari 1.0 sampai dengan 31049.0

```
[16] # Range
print(f"Range: {cek_range_is.min()}-{cek_range_is.max()}")

Range: 1.0-31049.0
```

- Persentase data kosong adalah 2.53%

```
[ ] # Data kosong
print(f"Persentase data kosong: {df_uncleaned['international_students'].isnull().sum()*100/len(df_uncleaned['international_students']):.2f}%")

Persentase data kosong: 2.53%
```

### 14. Atribut size

- Atribut ini berisi ukuran tanah dari universitas
- Jenis atribut ini adalah kategorical yang ordinal
- Atribut ini memiliki nilai seperti "S", "M", "L", "XL"

```
[ ] # Nilai dari atribut
print(f"Contoh nilai atribut: {df_uncleaned['size'].unique()}")
# note: nan disini artinya not a number atau kosong.

Contoh nilai atribut: ['M' 'L' 'S' 'XL' nan]
```

- Persentase data kosong adalah 0.03%

```
[ ] # Data kosong
print(f"Persentase data kosong: {df_uncleaned['size'].isnull().sum()*100/len(df_uncleaned['size']):.2f}%")
Persentase data kosong: 0.03%
```

## 15. Atribut faculty\_count

- Atribut ini berisi jumlah staff fakultas atau akademik dari universitas
- Jenis atribut ini adalah numerical yang diskrit

```
[ ] # Mengoreksi nilai data (jika tidak, data akan dianggap python tidak masuk akal.)
df_cor = df_uncleaned.loc[df_uncleaned['faculty_count'].isnull() == False]['faculty_count']
df_cor = df_cor.apply(lambda x: str(x).replace(',', ''))
df_cor = df_cor.apply(lambda x: int(float(x)))

# Nilai dari atribut
print(f"Contoh nilai atribut: {df_cor.unique()}")
print(df_cor.min())

Contoh nilai atribut: [ 3065  4725  4646  5800   968  6708  7195  2719  8000  2703  1050  4288
 3812  1767  5391  2843  4855  5154  4832  7087  4216  1763  7132  6174
 5673  3652  2944  3701  4889  3876  4487  9881  178  4473  3859  697
 3911  1172  5302  4545  20311  4835  4526  2201  6201  3571  7874  1307
 3219  1389  3513  2634  5166  633  1483  1492  6000  1534  2796  6663
 4386  2041  3127  3314  3852  2748  2887  3914  2939  7293  1267  4100
 3040  1775  3323  3386  1208  2677  3761  2316  3586  743  3463  16103]
```

- Atribut ini memiliki range dari 1 sampai 20311

```
[19] # Range
print(f"Range: {df_cor.min()}-{df_cor.max()}")

Range: 1-20311
```

- Atribut ini memiliki data kosong sebesar 1.20%

```
[21] # Data kosong
print(f"Persentase data kosong: {df_uncleaned['faculty_count'].isnull().sum()*100/len(df_uncleaned['faculty_count']):.2f}%")
Persentase data kosong: 1.20%
```

## Data Preprocessing dan Data Cleansing

Data yang kotor terdapat pada 11 atribut, yaitu link, rank\_display,score, city, type, research\_output,student\_faculty\_ratio, international\_students, size, faculty\_count. Data-data yang kotor kami bersihkan menggunakan library Pandas di Python dengan ketentuan masing-masing atribut kami jelaskan dibawah (Urutan nomor adalah urutan proses cleansing & preprocessing, bukan urutan atribut pada data asli).

### 1. Atribut link

- Deskripsi:

Data pada atribut link tidak akan dipakai untuk analisis sehingga dibuang saja. Persentase preprocessing data pada atribut ini adalah 100%.

```
[ ] # Mengecek data
print(df_uncleaned["link"].unique(), end="\n\n")

# Persentase Data Kotor
print("100% karena dibuang semua")

['https://www.topuniversities.com/universities/massachusetts-institute-technology-mit',
 'https://www.topuniversities.com/universities/stanford-university',
 'https://www.topuniversities.com/universities/harvard-university' ...
 'https://www.topuniversities.com/universities/universite-de-tunis',
 'https://www.topuniversities.com/universities/universite-de-tunis-el-manar',
 'https://www.topuniversities.com/universities/zagazig-university']

100% karena dibuang semua
```

- Cara Preprocessing: Membuang kolom link
- Solusi Pemrograman: Membuang kolom link pada dataframe menggunakan method .drop()

```
[ ] # Cleaning atribut link
df_chg_uncleaned.drop('link', axis=1, inplace=True)
```

## 2. Atribut logo

- Deskripsi:

Data pada atribut logo tidak akan dipakai untuk analisis sehingga dibuang saja. Persentase preprocessing data pada atribut ini adalah 100%.

```
[ ] # Mengecek kekotoran data
print(df_uncleaned["logo"].unique(), end="\n\n")

# Persentase Data Kotor
print("100% karena dibuang semua")

['https://www.topuniversities.com/sites/default/files/massachusetts-institute-of-technology-mit_410_small.jpg'
 'https://www.topuniversities.com/sites/default/files/stanford-university_573_small.jpg'
 'https://www.topuniversities.com/sites/default/files/harvard-university_253_small.jpg'
 ...
 'https://www.topuniversities.com/sites/default/files/universit-de-tunis_592560cf2aeae70239af5470_small.jpg'
 'https://www.topuniversities.com/sites/default/files/universit-de-tunis-el-manar_592560cf2aeae70239af5472_small.jpg'
 'https://www.topuniversities.com/sites/default/files/zagazig-university_592560cf2aeae70239af4f43_small.jpg']

100% karena dibuang semua
```

- Cara Preprocessing: Membuang kolom logo
- Solusi Pemrograman: Membuang kolom logo pada dataframe menggunakan method .drop()

```
[ ] # Cleaning atribut logo
df_chg_uncleaned.drop('logo', axis=1, inplace=True)
```

## 3. Atribut rank\_display

- Deskripsi:

Kekotoran data pada atribut rank\_display berupa beberapa data yang kosong dan data bukan bernilai integer (namun data bernilai string seperti "700-800"). Persentase kekotoran data pada atribut ini adalah 48.24%.

```
[ ] # Mengecek kekotoran data
# Format tidak benar
print(df_uncleaned.loc[(pd.to_numeric(df_uncleaned['rank_display'], errors='coerce').isna()) & (df_uncleaned['rank_display'].isnull() == False)]['rank_display'], end="\n\n")
# Data kosong (NaN)
print(df_uncleaned.loc[df_uncleaned['rank_display'].isnull()]['rank_display'], end="\n\n")

# Persentase data kotor (kosong + salah format)
rank_kosong = (df_uncleaned['rank_display'].isnull() == True).sum()
rank_format = ((pd.to_numeric(df_uncleaned['rank_display'], errors='coerce').isna()) & (df_uncleaned['rank_display'].isnull() == False)).sum()
print(f'Persentase data kotor adalah {(rank_kosong+rank_format)*100/len(df_uncleaned["rank_display"]):.2f}%')

400      401-410
401      401-410
402      401-410
403      401-410
404      401-410
...
6379    1001-1200
6380    1001-1200
6381    1001-1200
6382    1001-1200
6383    1001-1200
Name: rank_display, Length: 3059, dtype: object

914    NaN
915    NaN
916    NaN
917    NaN
918    NaN
...
5152   NaN
5154   NaN
5169   NaN
5177   NaN
5179   NaN
Name: rank_display, Length: 68, dtype: object

Persentase data kotor adalah 48.24%
```

- Cara Pembersihan:

Data dibersihkan dengan cara menghapus data yang kosong dan juga menghapus data universitas yang tidak 400 besar

- Solusi Pemrograman:

1. Membuang data yang kosong pada atribut rank\_display
2. Karena datanya range, maka akan ada karakter "-". Kita ganti dengan angka "0"
3. Kita ambil data universitas 400 tertinggi (untuk menghindari nilai range). Data range tadi akan terfilter karena angkanya yang lebih dari 400 setelah diganti.
4. Lalu reset index dataframe

```
[ ] # Cleaning atribut rank_display
df_chg_uncleaned = df_chg_uncleaned.dropna(subset=['rank_display']) # Step 1
df_chg_uncleaned['rank_display'] = df_chg_uncleaned['rank_display'].apply(lambda x: int(str(x).replace('-', '0'))) # Step 2
df_chg_uncleaned = df_chg_uncleaned.loc[df_chg_uncleaned["rank_display"] <= 400] # Step 3
df_chg_uncleaned.reset_index(drop=True, inplace=True) # Step 4
```

#### 4. Atribut score

- Deskripsi:

Kekotoran data Pada atribut score berupa data yang kosong. Persentase kekotoran data pada atribut ini adalah 56.49%.9

```
[ ] # Mengecek data data kosong (NaN)
print(df_uncleaned.loc[df_uncleaned['score'].isnull()]['score'], end="\n\n")

# Persentase Kekotoran
print(f"Persentase data kosong: {df_uncleaned['score'].isnull().sum()*100/len(df_uncleaned['score']):.2f}%")

375    NaN
400    NaN
401    NaN
402    NaN
403    NaN
...
6477    NaN
6478    NaN
6479    NaN
6480    NaN
6481    NaN
Name: score, Length: 3662, dtype: float64

Persentase data kosong: 56.49%
```

- Pembersihan:

Data yang kotor ini dibersihkan dengan cara mengisi data dengan ketentuan jika ada data universitas lain dengan ranking yang sama, maka score data yang hilang akan sama dengan score universitas tersebut dan jika data yang kotor terletak di antara dua ranking yang berbeda, maka score data yang kotor adalah rata-rata dari dua data terdekatnya (atas dan bawahnya).

- Solusi Pemrograman:

1. Mencari data score yang kosong
2. Looping pada atribut tersebut lalu mengisi data dengan ketentuan diatas

```
[ ] # Cleaning atribut score
score_empty = pd.isna(df_chg_uncleaned['score']).tolist() # Step 1
for i in range(len(df_chg_uncleaned)): # Step 2
    if score_empty[i] == True:
        if df_chg_uncleaned.loc[i-1, "rank_display"] == df_chg_uncleaned.loc[i, "rank_display"]:
            df_chg_uncleaned.at[i, 'score'] = df_chg_uncleaned.loc[i-1, "score"]
        elif df_chg_uncleaned.loc[i+1, "rank_display"] == df_chg_uncleaned.loc[i, "rank_display"]:
            df_chg_uncleaned.at[i, 'score'] = df_chg_uncleaned.loc[i+1, "score"]
        else:
            df_chg_uncleaned.at[i, 'score'] = (df_chg_uncleaned.loc[i-1, "score"]+df_chg_uncleaned.loc[i+1, "score"])/2
```

## 5. Atribut city

- Deskripsi:

Kekotoran data Pada atribut score berupa data yang kosong. Terdapat 54 data yang kotor dari 2400 data. Persentase kekotoran data pada atribut ini adalah 2.75%.

```
[ ] # Pengecekan Data kosong (NaN)
print(df_uncleaned.loc[df_uncleaned['city'].isnull()]['city'], end="\n\n")

# Persentase Kekotoran
print(f"Persentase data kosong: {df_uncleaned['city'].isnull().sum()*100/len(df_uncleaned['city']):.2f}%")
```

116	NaN
194	NaN
241	NaN
265	NaN
344	NaN
...	
6405	NaN
6408	NaN
6422	NaN
6427	NaN
6431	NaN

Name: city, Length: 178, dtype: object

Persentase data kosong: 2.75%

- Pembersihan:

Data yang kotor ini dibersihkan dengan cara mengisi data dengan mencari kota dari universitas yang hilang datanya pada google.

- Solusi Pemrograman:

1. Mencari data yang kosong pada atribut city
2. Looping pada atribut tersebut lalu mengisi data.

```
[ ] # Cleaning atribut city
city_empty = pd.isna(df_chg_uncleaned['city']).tolist() # Step 1
for i in range(len(df_chg_uncleaned)): # Step 2
    if city_empty[i] == True:
        if df_chg_uncleaned.loc[i, "university"] == "Université PSL" or df_chg_uncleaned.loc[i, "university"] == "Sorbonne University":
            df_chg_uncleaned.at[i, "city"] = "Paris"
        elif df_chg_uncleaned.loc[i, "university"] == "Université Paris-Saclay":
            df_chg_uncleaned.at[i, "city"] = "Gif-sur-Yvette"
        elif df_chg_uncleaned.loc[i, "university"] == "Aarhus University":
            df_chg_uncleaned.at[i, "city"] = "Aarhus"
        elif df_chg_uncleaned.loc[i, "university"] == "Queen's University Belfast":
            df_chg_uncleaned.at[i, "city"] = "Belfast"
        elif df_chg_uncleaned.loc[i, "university"] == "Ulsan National Institute of Science and Technology (UNIST)":
            df_chg_uncleaned.at[i, "city"] = "Ulsan"
        elif df_chg_uncleaned.loc[i, "university"] == "Kyung Hee University":
            df_chg_uncleaned.at[i, "city"] = "Seoul"
        elif df_chg_uncleaned.loc[i, "university"] == "National Yang Ming Chiao Tung University":
            df_chg_uncleaned.at[i, "city"] = "Taipei City"
        elif df_chg_uncleaned.loc[i, "university"] == "Southern University of Science and Technology":
            df_chg_uncleaned.at[i, "city"] = "Shenzhen"
        elif df_chg_uncleaned.loc[i, "university"] == "University of Macau":
            df_chg_uncleaned.at[i, "city"] = "Taipa"
        elif df_chg_uncleaned.loc[i, "university"] == "Brunel University London":
            df_chg_uncleaned.at[i, "city"] = "Uxbridge"
        elif df_chg_uncleaned.loc[i, "university"] == "Oxford Brookes University":
            df_chg_uncleaned.at[i, "city"] = "Oxford"
        elif df_chg_uncleaned.loc[i, "university"] == "National Research Tomsk Polytechnic University":
            df_chg_uncleaned.at[i, "city"] = "Tomsk"
        elif df_chg_uncleaned.loc[i, "university"] == "UCSI University":
            df_chg_uncleaned.at[i, "city"] = "Kuala Lumpur"
```

## 6. Atribut type

- Deskripsi:

Kekotoran data Pada atribut type berupa data yang kosong. Persentase kekotoran data pada atribut ini adalah 0.19%.

```
[ ] # Pengecekan Data kosong (NaN)
print(df_uncleaned.loc[df_uncleaned['type'].isnull()]['type'], end="\n\n")

# Persentase Kekotoran
print(f"Persentase data kotor: {df_uncleaned['type'].isnull().sum()*100/len(df_uncleaned['type']):.2f}%")
```

1124	NaN
1126	NaN
2076	NaN
2084	NaN
3105	NaN
3131	NaN
3152	NaN
3983	NaN
5037	NaN
5064	NaN
5065	NaN
5076	NaN
	Name: type, dtype: object
	Persentase data kotor: 0.19%

- Pembersihan:

Data yang kotor ini dibersihkan dengan cara mengisi data dengan mencari tipe dari universitas yang hilang datanya public atau private pada google (setelah searching, ternyata yang kosong semuanya Public).

- Solusi Pemrograman:

1. Mencari data yang kosong pada atribut type
2. Looping pada atribut tersebut lalu mengisi data.

```
[ ] # Cleaning atribut type
type_empty = pd.isna(df_chg_uncleaned['type']).tolist() # Step 1
for i in range(len(df_chg_uncleaned)): # Step 2
    if type_empty[i] == True:
        df_chg_uncleaned.at[i, "type"] = "Public"
```

## 7. Atribut research\_output

- Deskripsi:

Kekotoran data Pada atribut score berupa data yang kosong. Persentase kekotoran data pada atribut ini adalah 0.03%.

```
[ ] # Pengecekan Data kosong (NaN)
print(df_uncleaned.loc[df_uncleaned['research_output'].isnull()]['research_output'], end="\n\n")

# Persentase Kekotoran
print(f"Persentase data kotor: {df_uncleaned['research_output'].isnull().sum()*100/len(df_uncleaned['research_output']):.2f}%")
```

	3152	5076
NaN		
Name: research_output, dtype: object		

Persentase data kotor: 0.03%

- Pembersihan:

Data yang kotor ini dibersihkan dengan cara menghapus row dari data yang kotor tersebut.

- Solusi Pemrograman:

Menghapus baris pada data research\_output kosong

```
[ ] # Cleaning research_output  
df_chg_uncleaned = df_chg_uncleaned.dropna(subset=["research_output"])
```

## 8. Atribut student\_faculty\_ratio

- Deskripsi:

Kekotoran data Pada atribut score berupa data yang kosong. Persentase kekotoran data pada atribut ini adalah 1.16%.

```
[ ] # Pengecekan Data kosong (NaN)  
print(df_uncleaned.loc[df_uncleaned['student_faculty_ratio'].isnull()]['student_faculty_ratio'], end="\n\n")  
  
# Persentase Kekotoran  
print(f"Persentase data kotor: {df_uncleaned['student_faculty_ratio'].isnull().sum()*100/len(df_uncleaned['student_faculty_ratio']):.2f}%")  
  
165    NaN  
575    NaN  
915    NaN  
917    NaN  
919    NaN  
     ..  
5139    NaN  
5177    NaN  
5179    NaN  
5900    NaN  
6279    NaN  
Name: student_faculty_ratio, Length: 75, dtype: float64  
  
Persentase data kotor: 1.16%
```

- Pembersihan:

Data yang kotor ini dibersihkan dengan cara menghapus row dari data yang kotor tersebut.

- Solusi Pemrograman:

Menghapus baris pada data student\_faculty\_ratio kosong

```
[ ] # Cleaning student_faculty_ratio  
df_chg_uncleaned = df_chg_uncleaned.dropna(subset=["student_faculty_ratio"])
```

## 9. Atribut size

- Deskripsi:

Kekotoran data Pada atribut score berupa data yang kosong. Persentase kekotoran data pada atribut ini adalah 0.03%.

```
[ ] # Pengecekan Data kosong (NaN)
print(df_uncleaned.loc[df_uncleaned['size'].isnull()]['size'], end="\n\n")

# Persentase Kekotoran
print(f"Persentase data kotor: {df_uncleaned['size'].isnull().sum()*100/len(df_uncleaned['size']):.2f}%")

3152    NaN
5076    NaN
Name: size, dtype: object

Persentase data kotor: 0.03%
```

- Pembersihan:

Data yang kotor ini dibersihkan dengan cara menghapus row dari data yang kotor tersebut.

- Solusi Pemrograman:

Menghapus baris pada data size kosong

```
[ ] # Cleaning size
df_chg_uncleaned = df_chg_uncleaned.dropna(subset=["size"])
```

## 10. Atribut international\_students

- Deskripsi:

Kekotoran data Pada atribut score berupa data yang kosong dan format angka yang salah. Persentase kekotoran data pada atribut ini adalah 37.03%.

```
[22] # Mengecek kekotoran data
# Format tidak benar
print(df_uncleaned.loc[(pd.to_numeric(df_uncleaned['international_students'], errors='coerce').isna()) & (df_uncleaned['international_students'].isnull() == False)]['international_students'], end="\n\n")
print(df_uncleaned.loc[df_uncleaned['international_students'].isnull()]['international_students'], end="\n\n")

# Persentase data kotor (kosong + salah format)
rank_kosong = (df_uncleaned['international_students'].isnull() == True).sum()
rank_format = ((pd.to_numeric(df_uncleaned['international_students'], errors='coerce').isna()) & (df_uncleaned['international_students'].isnull() == False)).sum()
print(f"Persentase data kotor adalah {((rank_kosong+rank_format)*100/len(df_uncleaned['international_students'])):.2f}%")

0      3,730
1      3,879
2      5,877
3      7,925
5      8,442
...
6478    2,441
6476    1,976
6479    1,026
6480    2,394
6481    2,300
Name: international_students, Length: 2236, dtype: object

472     NaN
634     NaN
678     NaN
703     NaN
819     NaN
...
6364    NaN
6441    NaN
6443    NaN
6445    NaN
6469    NaN
Name: international_students, Length: 164, dtype: object

Persentase data kotor adalah 37.03%
```

- Pembersihan:

Data yang kotor ini dibersihkan dengan cara menghapus row dari data yang kotor tersebut.

- Solusi Pemrograman:

1. Menghapus baris pada data international\_students yang kosong
2. Menghapus koma (",") pada angka
3. Menghapus koma (".") pada angka
4. Data jadikan integer

```
[23] # Cleaning atribut international_students
df_chg_uncleaned = df_chg_uncleaned.dropna(subset=['international_students']) # Step 1
df_chg_uncleaned['international_students'] = df_chg_uncleaned['international_students'].apply(lambda x: str(x).replace(',', '')) # Step 2
df_chg_uncleaned['international_students'] = df_chg_uncleaned['international_students'].apply(lambda x: str(x).replace('.', '')) # Step 3
df_chg_uncleaned['international_students'] = df_chg_uncleaned['international_students'].apply(lambda x: int(x)) # Step 4
```

## 11. Atribut faculty\_count

- Deskripsi:

Kekotoran data Pada atribut score berupa data yang kosong dan format angka yang salah.

Persentase kekotoran data pada atribut ini adalah 39.17%.

```
[ ] # Mengecek kekotoran data
# Format tidak benar
print(df_uncleaned.loc[(pd.to_numeric(df_uncleaned['faculty_count'], errors='coerce').isna()) & (df_uncleaned['faculty_count'].isnull() == False)]['faculty_count'], end="\n\n")
# Data kosong (NaN)
print(df_uncleaned.loc[df_uncleaned['faculty_count'].isnull()]['faculty_count'], end="\n\n")

# Persentase data kotor (kosong + salah format)
rank_kosong = (df_uncleaned['faculty_count'].isnull() == True).sum()
rank_format = ((pd.to_numeric(df_uncleaned['faculty_count'], errors='coerce').isna()) & (df_uncleaned['faculty_count'].isnull() == False)).sum()
print(f"Persentase data kotor adalah {(rank_kosong+rank_format)*100/len(df_uncleaned['faculty_count']):.2f}%")


0    3,065
1    4,725
2    4,646
3    5,800
5    6,708
...
6477   1,174
6478   3,564
6479   1,113
6480   1,688
6481   5,871
Name: faculty_count, Length: 2461, dtype: object

213    NaN
404    NaN
771    NaN
890    NaN
917    NaN
...
5179   NaN
5789   NaN
6048   NaN
6235   NaN
6348   NaN
Name: faculty_count, Length: 78, dtype: object

Persentase data kotor adalah 39.17%
```

- Pembersihan:

Data yang kotor ini dibersihkan dengan cara menghapus row dari data yang kotor tersebut.

- Solusi Pemrograman:

1. Menghapus baris pada data faculty\_count yang kosong
2. Menghapus koma (",") pada angka
3. Menghapus koma (".") pada angka
4. Data jadikan integer

```
[ ] # Cleaning atribut faculty_count
df_chg_uncleaned = df_chg_uncleaned.dropna(subset=['faculty_count']) # Step 1
df_chg_uncleaned['faculty_count'] = df_chg_uncleaned['faculty_count'].apply(lambda x: str(x).replace(',', '')) # Step 2
df_chg_uncleaned['faculty_count'] = df_chg_uncleaned['faculty_count'].apply(lambda x: str(x).replace('.', '')) # Step 3
df_chg_uncleaned['faculty_count'] = df_chg_uncleaned['faculty_count'].apply(lambda x: int(x)) # Step 4
```

## Convert data

- Setelah data dibersihkan, convert data menjadi file format CSV.

```
[ ] df_uncleaned.to_csv("qs_ranking_400_cleaned.csv", index=False)
```

# Statistik

## 1. Sample Data

### 1.a. Sepuluh Persen Data Pertama

```
[ ] # Sepuluh Persen data pertama  
df[:240]
```

	university	year	rank_display	score	country	city	region	type	research_output	student_faculty_ratio	international_students	size	faculty_count
0	Massachusetts Institute of Technology (MIT)	2017	1	100.0	United States	Cambridge	North America	Private	Very High	4.0	3730	M	3065
1	Stanford University	2017	2	98.7	United States	Stanford	North America	Private	Very High	3.0	3879	L	4725
2	Harvard University	2017	3	98.3	United States	Cambridge	North America	Private	Very High	5.0	5877	L	4646
3	University of Cambridge	2017	4	97.2	United Kingdom	Cambridge	Europe	Public	Very high	4.0	7925	L	5800
4	California Institute of Technology (Caltech)	2017	5	96.9	United States	Pasadena	North America	Private	Very High	2.0	692	S	968
...	...	...	...	...	...	...	...	...	...	...	...	...	...
235	Tufts University	2017	238	42.2	United States	Medford	North America	Private	Very High	8.0	1798	M	1459
236	Complutense University of Madrid	2017	239	42.1	Spain	Madrid	Europe	Public	Very High	13.0	7295	XL	4778
237	University of Leicester	2017	239	42.1	United Kingdom	Leicester	Europe	Public	Very High	10.0	4919	L	1518

Dapat dilihat bahwa data sudah terurut berdasarkan rangking untuk setiap tahunnya.

### 1.b. Data ITB pada universitas top 400 rentang 2017-2022

```
[ ] # Data ITB pada universitas top 400 rentang 2017-2022  
df.loc[df['university'] == 'Bandung Institute of Technology (ITB)']
```

	university	year	rank_display	score	country	city	region	type	research_output	student_faculty_ratio	international_students	size	faculty_count
726	Bandung Institute of Technology (ITB)	2018	331	35.3	Indonesia	Bandung	Asia	Public	Very High	9.0	613	L	2228
1151	Bandung Institute of Technology (ITB)	2019	359	30.4	Indonesia	Bandung	Asia	Public	Very High	9.0	613	L	2228
1518	Bandung Institute of Technology (ITB)	2020	331	32.3	Indonesia	Bandung	Asia	Public	Very High	9.0	613	L	2228
1904	Bandung Institute of Technology (ITB)	2021	313	33.3	Indonesia	Bandung	Asia	Public	Very High	9.0	613	L	2228
2292	Bandung Institute of Technology (ITB)	2022	303	34.2	Indonesia	Bandung	Asia	Public	Very High	9.0	613	L	2228

Dapat dilihat bahwa ITB tidak masuk 400 besar pada tahun 2017 dan untuk tahun 2018-2022, ranking ITB berada pada rentang 303-351.

### 1.c. Urutan data dari skor tertinggi

```
[ ] # Urutan data dari skor tertinggi  
df.sort_values(['score'], ascending=False)
```

	university	year	rank_display	score	country	city	region	type	research_output	student_faculty_ratio	international_students	size	faculty_count
0	Massachusetts Institute of Technology (MIT)	2017	1	100.0	United States	Cambridge	North America	Private	Very High	4.0	3730	M	3065
1989	Massachusetts Institute of Technology (MIT)	2022	1	100.0	United States	Cambridge	North America	Private	Very High	4.0	3730	M	3065
796	Massachusetts Institute of Technology (MIT)	2019	1	100.0	United States	Cambridge	North America	Private	Very High	4.0	3730	M	3065
1193	Massachusetts Institute of Technology (MIT)	2020	1	100.0	United States	Cambridge	North America	Private	Very High	4.0	3730	M	3065
1592	Massachusetts Institute of Technology (MIT)	2021	1	100.0	United States	Cambridge	North America	Private	Very High	4.0	3730	M	3065
...	...	...	...	...	...	...	...	...	...	...	...	...	...
1987	Université de Strasbourg	2021	398	28.1	France	Strasbourg	Europe	Public	Very High	21.0	9111	XL	2544
1986	La Trobe University	2021	398	28.1	Australia	Melbourne	Oceania	Public	Very High	20.0	9456	L	1252
2387	Indian Institute of Technology Roorkee (IITR)	2022	400	28.0	India	Roorkee	Asia	Public	Very High	17.0	202	M	499

Dapat dilihat bahwa MIT selalu mendapat skor sempurna, yaitu 100. Selain itu, skor dan ranking universitas rangking tinggi cenderung tidak berubah drastis.

### 1.d. Data Universitas di Asia tahun 2022 yang masuk top 400 Dunia yang diurutkan berdasarkan ranking

```
[ ] # Data Universitas di Asia tahun 2022 yang masuk top 400 Dunia yang diurutkan berdasarkan rankingnya  
df.loc[(df['region'] == 'Asia') & (df['year'] == 2022)].sort_values(['rank_display'], ascending=1)
```

	university	year	rank_display	score	country	city	region	type	research_output	student_faculty_ratio	international_students	size	faculty_count
1999	National University of Singapore (NUS)	2022	11	93.9	Singapore	Singapore	Asia	Public	Very High	7.0	7551	XL	4288
2000	Nanyang Technological University, Singapore (NTU)	2022	12	90.8	Singapore	Singapore	Asia	Public	Very High	6.0	6091	L	3812
2005	Tsinghua University	2022	17	89.0	China (Mainland)	Beijing	Asia	Public	Very High	6.0	5420	XL	6174
2006	Peking University	2022	18	88.8	China (Mainland)	Beijing	Asia	Public	Very High	6.0	5436	XL	5302
2010	The University of Hong Kong	2022	22	86.3	Hong Kong SAR	Hong Kong	Asia	Public	Very High	7.0	8311	L	2944
...	...	...	...	...	...	...	...	...	...	...	...	...	...
2372	Beihang University (former BUAA)	2022	383	29.1	China (Mainland)	Beijing	Asia	Public	Very High	12.0	1575	XL	2529
2374	Kobe University	2022	386	28.9	Japan	Kobe City	Asia	Public	Very High	8.0	1216	L	1986
2383	Indian Institute of Technology Guwahati (IITG)	2022	395	28.3	India	Guwahati	Asia	Public	Very High	17.0	42	M	410
2386	University of the Philippines	2022	399	28.1	Philippines	Quezon City	Asia	Public	High	9.0	292	XL	4480
2387	Indian Institute of Technology Roorkee (IITR)	2022	400	28.0	India	Roorkee	Asia	Public	Very High	17.0	202	M	499

Dapat dilihat bahwa untuk tahun 2022, National University of Singapore (NUS) merupakan universitas top 1 di Asia berdasarkan QS ranking. Dari sample juga dapat diketahui bahwa ada 98 universitas Asia yang masuk top 400 global pada tahun yang sama.

### 1.e. Universitas dengan research output high

# Universitas top 400 dengan research_output 'High'. df.loc[df['research_output'] == 'High'].sort_values(['score'], ascending=False)													
	university	year	rank_display	score	country	city	region	type	research_output	student_faculty_ratio	international_students	size	faculty_count
1745	Tecnológico de Monterrey	2021	155	49.9	Mexico	Monterrey	Latin America	Private	High	8.0	3514	XL	5894
1349	Tecnológico de Monterrey	2020	158	48.5	Mexico	Monterrey	Latin America	Private	High	8.0	3514	XL	5894
2151	Tecnológico de Monterrey	2022	161	48.2	Mexico	Monterrey	Latin America	Private	High	8.0	3514	XL	5894
595	Tecnológico de Monterrey	2018	199	48.0	Mexico	Monterrey	Latin America	Private	High	8.0	3514	XL	5894
1756	Al-Farabi Kazakh National University	2021	165	46.9	Kazakhstan	Almaty	Asia	Public	High	6.0	3054	L	4047
...	...	...	...	...	...	...	...	...	...	...	...	...	...
1985	University of the Philippines	2021	396	28.2	Philippines	Quezon City	Asia	Public	High	9.0	292	XL	4480
1589	Universidad Austral	2020	400	28.2	Argentina	Pilar	Latin America	Private	High	7.0	334	S	663
2386	University of the Philippines	2022	399	28.1	Philippines	Quezon City	Asia	Public	High	9.0	292	XL	4480
2388	Universidad Austral	2022	400	28.0	Argentina	Pilar	Latin America	Private	High	7.0	334	S	663
2389	Vilnius University	2022	400	28.0	Lithuania	Vilnius	Europe	Public	High	7.0	1290	L	2461
121 rows × 13 columns													

Didapat bahwa Universitas dengan research output high dengan score tertinggi adalah Tecnológico de Monterrey dengan score 49.9 yang diraih pada tahun 2021. Selain itu, juga diketahui hanya ada 121 kali kemunculan universitas dengan research output high untuk universitas top 400 tiap tahun untuk periode 2017-2022.

## 2. Statistik Data

Dari statistik yang dibagi untuk kategori masing-masing, Pembaca dapat mendapat informasi berikut :

- **Count :** Count merepresentasikan total data sampel yang dianalisis untuk tiap kategori. Dapat dilihat untuk gabungan semua tahun, terdapat 2390 data. Pembaca juga dapat melihat banyak data untuk tiap tahunnya.
- **Mean :** Mean menunjukkan rata-rata untuk tiap pembagian

- **Std** : Std menunjukkan standar deviasi yang dapat menggambarkan seberapa besar variasi data untuk menentukan kedekatan sebaran data yang ada di dalam sampel. Jika nilai standar deviasi lebih besar dari nilai mean berarti nilai mean merupakan representasi yang buruk dari keseluruhan data.
- **Min** : Min menunjukkan data terkecil pada kategori yang dipilih.
- **Max** : Max menunjukkan data terbesar pada kategori yang dipilih
- **Persentil** : Persentil menunjukkan persebaran data sesuai persen yang ditampilkan. Pada data ini, pembaca dapat melihat persentil 10%, 25%, 50%, 75%, 90%. Dapat dilihat bahwa nilai persentil akan naik seiring naiknya persentase persentil yang dicar

## 2.a. Statistik kuantitatif gabungan 2017-2022

df[['score', 'student_faculty_ratio', 'international_students', 'faculty_count']].describe(percentiles = [.10,.25,.50,.75,.90])				
	score	student_faculty_ratio	international_students	faculty_count
count	2390.000000	2390.000000	2390.000000	2390.000000
mean	49.875021	10.314226	4882.625941	2713.611715
std	18.094296	4.854217	4287.277867	2076.498573
min	28.000000	2.000000	5.000000	150.000000
10%	31.190000	5.000000	980.900000	881.600000
25%	35.025000	7.000000	2110.000000	1339.000000
50%	44.700000	9.000000	3879.000000	2316.000000
75%	61.475000	13.000000	6403.000000	3623.000000
90%	79.220000	17.000000	9177.000000	4832.000000
max	100.000000	35.000000	31049.000000	20311.000000

Tabel menunjukkan statistik dari atribut skor, rasio jumlah mahasiswa per jumlah staf, jumlah mahasiswa internasional, dan jumlah fakultas dari gabungan tahun 2017–2022. Untuk setiap atribut, nilai standar deviasinya lebih kecil daripada rata-rata yang berarti data tidak menyimpang jauh.

## 2.b. Statistik score Tiap Tahunnya

df.groupby('year')['score'].describe(percentiles = [.10,.25,.50,.75,.90])											
	count	mean	std	min	10%	25%	50%	75%	90%	max	
year											
2017	398.0	51.555276	18.001223	30.0	32.27	36.725	46.45	62.800	79.69	100.0	
2018	398.0	52.058040	17.762125	30.0	33.10	37.625	47.85	64.575	80.46	100.0	
2019	397.0	48.934761	18.143278	28.2	30.36	34.500	43.90	59.800	78.64	100.0	
2020	399.0	48.765414	18.112980	28.2	30.48	34.300	44.00	59.700	78.04	100.0	
2021	397.0	48.484131	17.919353	28.1	30.20	34.400	42.80	58.800	76.74	100.0	
2022	401.0	49.452618	18.403887	28.0	30.90	34.700	44.10	59.600	79.10	100.0	

Tabel menunjukkan statistik dari atribut skor dari tahun 2017–2022. Standar deviasinya memiliki nilai yang rendah yang berarti datanya tidak terlalu menyebar. Atribut ini selalu memiliki nilai maksimum 100 dan nilai minimumnya juga cukup stabil, yaitu berada pada rentang 28.0–30.0. Nilai mean dari atribut ini juga cukup stabil. Nilai persentilnya juga naik progresif sesuai dengan tingkat persennya.

## 2.c. Statistik Rasio Jumlah Mahasiswa per Jumlah Staf Tiap Tahunnya

df.groupby('year')['student_faculty_ratio'].describe(percentiles = [.10,.25,.50,.75,.90])											
	count	mean	std	min	10%	25%	50%	75%	90%	max	
year											
2017	398.0	10.502513	4.735710	2.0	5.0	7.0	10.0	13.0	17.0	32.0	
2018	398.0	10.376884	4.825798	2.0	5.0	7.0	9.0	13.0	17.0	32.0	
2019	397.0	10.360202	4.974047	2.0	5.0	7.0	9.0	13.0	17.0	35.0	
2020	399.0	10.230576	4.814339	2.0	5.0	7.0	9.0	12.5	17.0	32.0	
2021	397.0	10.136020	4.821735	2.0	5.0	7.0	9.0	12.0	17.0	32.0	
2022	401.0	10.279302	4.970593	2.0	5.0	7.0	9.0	12.0	17.0	35.0	

Tabel menunjukkan statistik dari atribut Rasio Jumlah Mahasiswa per Jumlah Staf dari tahun 2017–2022. Standar deviasinya juga memiliki nilai yang rendah yang berarti datanya tidak terlalu menyebar. Atribut ini selalu memiliki nilai minimum 2 dan nilai maksimum dari rentang 32–35. Nilai mean dari atribut ini juga cukup stabil dengan range antartahun yang kecil. Persentil dari atribut ini naik progresif sesuai dengan tingkat persennya dan stabil sehingga cukup mirip untuk tiap tahunnya.

## 2.d. Jumlah Mahasiswa Internasional Tiap Tahunnya

	df.groupby('year')['international_students'].describe(percentiles = [.10,.25,.50,.75,.90])										
	count	mean	std	min	10%	25%	50%	75%	90%	max	
year											
2017	398.0	4922.811558	4291.885691	5.0	1069.3	2111.25	3889.5	6420.25	9220.8	31049.0	
2018	398.0	4877.361809	4296.513380	5.0	1012.4	2102.75	3873.5	6395.00	9130.8	31049.0	
2019	397.0	4878.302267	4293.348260	5.0	992.2	2110.00	3875.0	6403.00	9137.4	31049.0	
2020	399.0	4858.416040	4287.215487	5.0	892.2	2107.50	3879.0	6387.00	9107.8	31049.0	
2021	397.0	4898.586902	4295.274520	5.0	925.6	2134.00	3883.0	6426.00	9137.4	31049.0	
2022	401.0	4860.533666	4286.097897	5.0	869.0	2115.00	3879.0	6403.00	9107.0	31049.0	

Tabel menunjukkan statistik dari atribut Jumlah Mahasiswa Internasional dari tahun 2017–2022. Standar deviasinya memiliki nilai yang cukup tinggi, tetapi belum melewati rata-rata yang berarti nilainya cukup menyebar, tetapi rata-rata masih dapat mempresentasikan keseluruhan data yang diambil. Atribut ini selalu memiliki nilai minimum 5 dan nilai maksimum 31049. Nilai mean dari atribut ini dapat dikatakan cukup stabil karena nilainya cukup mirip tiap tahunnya. Persentil dari atribut ini naik progresif sesuai dengan tingkat persennya dan cukup mirip untuk tiap tahunnya.

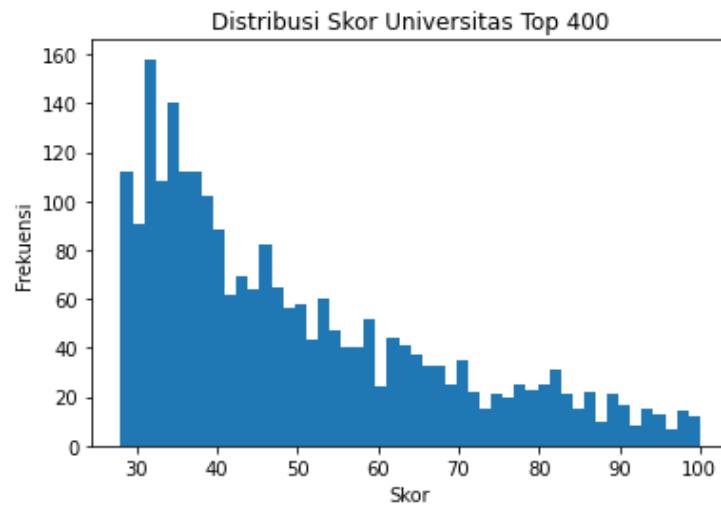
## 2.e. Jumlah Fakultas Tiap Tahunnya

df.groupby('year')[ 'faculty_count'].describe(percentiles = [.10,.25,.50,.75,.90])											
	count	mean	std	min	10%	25%	50%	75%	90%	max	
year											
2017	398.0	2714.979899	2070.825578	178.0	916.2	1376.25	2316.0	3598.25	4794.2	20311.0	
2018	398.0	2686.680905	2084.504187	150.0	854.9	1321.25	2269.5	3582.25	4794.2	20311.0	
2019	397.0	2704.516373	2080.522457	150.0	883.6	1339.00	2295.0	3600.00	4799.6	20311.0	
2020	399.0	2715.370927	2083.397200	150.0	883.8	1336.00	2295.0	3641.00	4832.6	20311.0	
2021	397.0	2749.546599	2079.580124	180.0	891.8	1389.00	2353.0	3658.00	4833.2	20311.0	
2022	401.0	2710.660848	2072.713638	180.0	869.0	1333.00	2325.0	3623.00	4778.0	20311.0	

Tabel menunjukkan statistik dari atribut Jumlah Fakultas Tiap Tahunnya dari tahun 2017–2022. Standar deviasinya memiliki nilai yang cukup dekat dengan rata-rata secara persentase, tetapi belum melewati rata-rata yang berarti nilainya cukup menyebar, tetapi rata-rata masih dapat mempresentasikan keseluruhan data yang diambil. Atribut ini selalu memiliki nilai maksimum 20311 dan nilai minimum dari rentang 150–180. Nilai mean dari atribut ini juga cukup stabil dan nilainya cukup mirip tiap tahunnya. Persentil dari atribut ini juga naik secara progresif dan persentil tahun 2017 cukup unik karena memiliki nilai yang cenderung lebih tinggi.

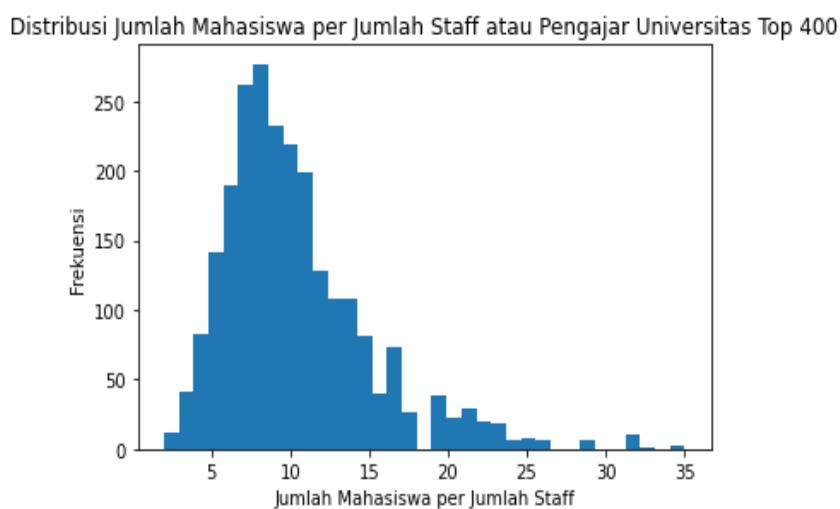
### 3. Distribusi Data

#### 3.a Distribusi Skor Universitas Top 400



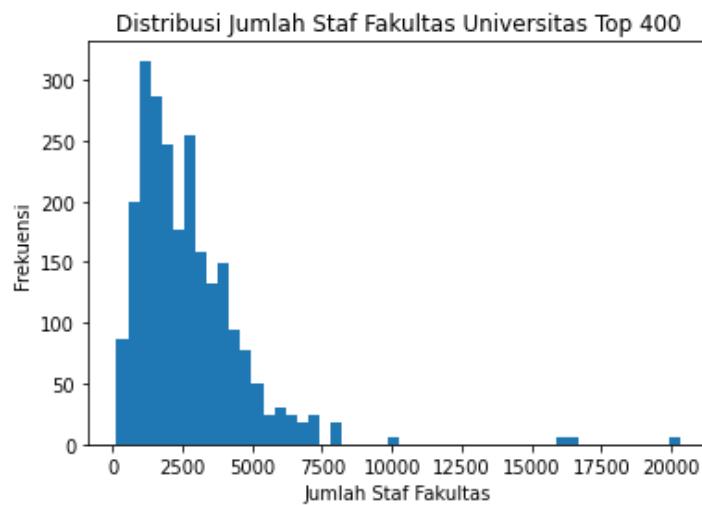
Dari grafik, didapat informasi bahwa histogram persebaran atribut skor adalah *skewed distribution* (distribusi miring) dan kemiringannya ke arah kanan (*right-skewed*). Datanya berkumpul pada rentang 35–37 dan kemudian frekuensinya turun seiring penambahan skor.

#### 3.b Distribusi Rasio Jumlah Mahasiswa per Jumlah Staf Universitas Top 400



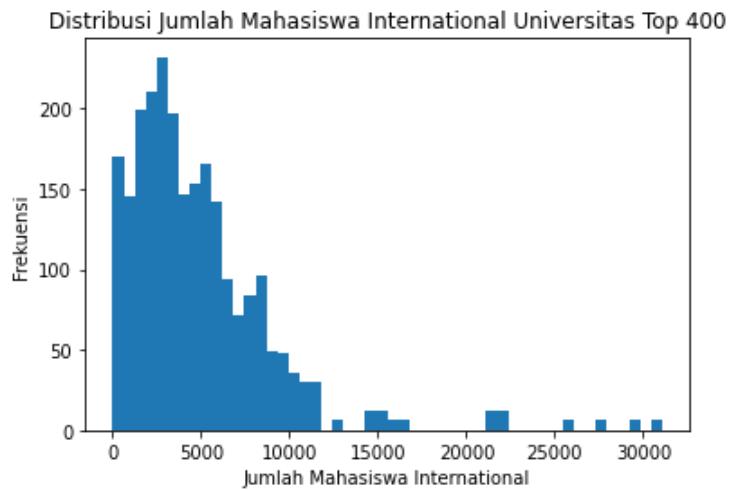
Dari grafik, didapat informasi bahwa atribut Rasio Jumlah Mahasiswa per Jumlah Staf berbentuk *right-skewed* dan memiliki nilai modus yang mendekati 8 dan data berkumpul dari rentang 2 hingga mendekati 17, kemudian ada data juga kumpulan beberapa data di rentang 19–26 dan ada beberapa data yang menyebar di atas 26.

### 3.c Distribusi Jumlah Fakultas Universitas Top 400



Dari grafik, didapat informasi bahwa atribut Jumlah Staf Fakultas juga berbentuk *right-skewed* memiliki nilai modus di antara 0–2500 kemudian turun secara progresif, kecuali untuk nilai data di sekitar 3000, hingga data dengan nilai 7500. Kemudian, ada beberapa data di atas 7500 yang menyebar secara tidak merata.

### 3.d. Distribusi Jumlah Mahasiswa International Universitas Top 400



Dari grafik, didapat informasi bahwa atribut Jumlah Mahasiswa International berbentuk *right-skewed* dan memiliki nilai modus di sekitar 2500. Frekuensi data naik untuk nilai 0–2500 dan turun pada rentang 2500–6000. Kemudian frekuensi data turun drastis dan naik kembali pada data dengan nilai di sekitar 8000 dan turun secara progresif hingga data dengan nilai di sekitar 11000. Kemudian ada beberapa data di atas 11000 yang menyebar secara tidak merata.

## Visualisasi

### 1. Perbandingan Kategori

#### 1.a. Perbandingan Banyak Universitas Top 400 di Setiap Negara di dalam Wilayah Tertentu pada 2022:

```
df['region'] = df['region'].map({'North America':'Amerika Utara',
                                'Europe':'Eropa','Asia':'Asia','Oceania':'Oceania',
                                'Latin America':'Amerika Latin','Africa':'Afrika'})
df[1987:2387].groupby('region')['country'].value_counts().unstack().plot(kind='bar',figsize=(28,14),
                                                                      fontsize=12,stacked=True,legend=True)
plt.title('Banyak Universitas Top 400 di Setiap Wilayah pada 2022',fontsize=16)
plt.xlabel('Wilayah',fontsize=16)
plt.ylabel('Jumlah Universitas',fontsize=16)
plt.legend(title='Wilayah:',loc="upper center",mode='expand', ncol=len(df.columns))
plt.show()
df['region'] = df['region'].map({'Amerika Utara':'North America','Eropa':'Europe','Asia':'Asia',
                                'Oceania':'Oceania','Amerika Latin':'Latin America','Afrika':'Afrika'})
```

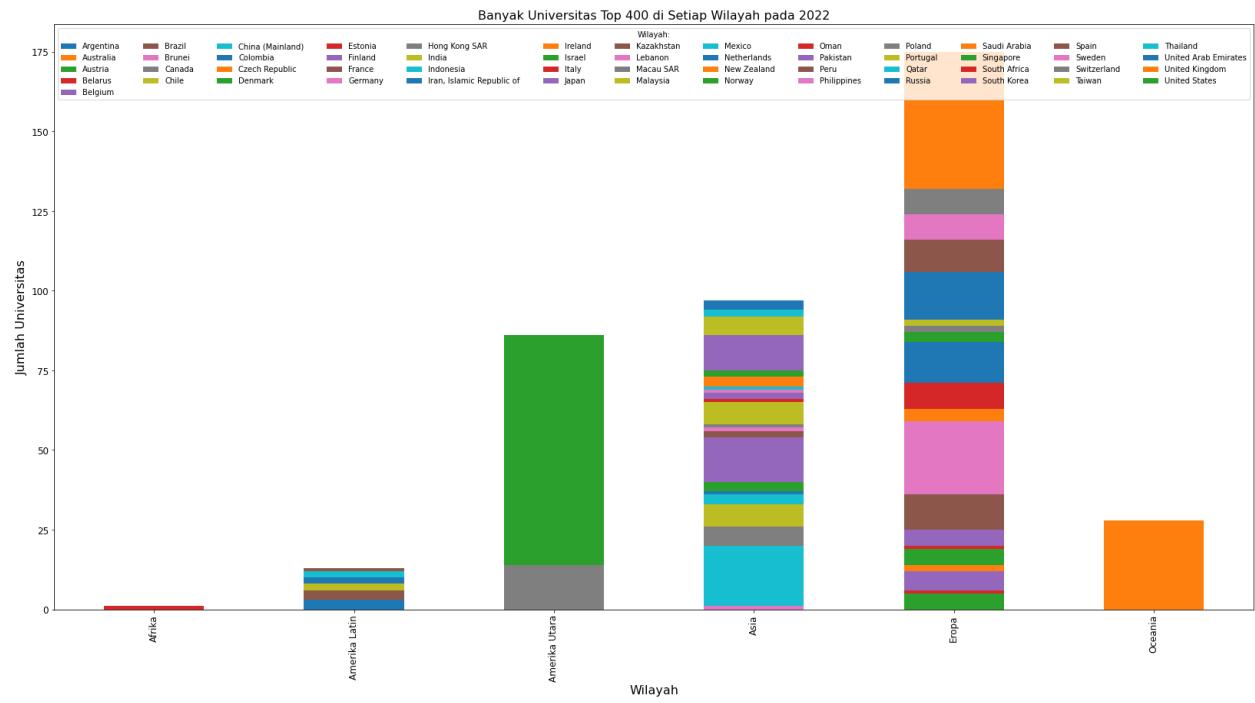


Diagram batang vertikal antara wilayah dan banyak universitas yang termasuk 400 besar dalam ranking QS World. Dari diagram ini kita melihat:

- Eropa memiliki universitas top 400 yang paling banyak dibandingkan wilayah lainnya disusul oleh Asia dan Amerika Utara.
- Oceania dan Amerika Latin memiliki Universitas top 100 yang jauh lebih sedikit dibandingkan wilayah lainnya.
- South Africa merupakan universitas Afrika satu-satunya yang masuk Ranking top 400.
- Amerika Utara dan Oceania hanya memiliki 2 Negara dengan Universitas top 400 yaitu United States dan Canada, dan Australia dan New Zealand.

### 1.b. Perbandingan Tipe Universitas Top 400 (Swasta/Umum) pada 2022:

```
df['type'] = df['type'].map({'Private':'Swasta','Public':'Umum'})
df[1987:2387]['type'].value_counts().plot(kind='barh',color=('red','blue'),figsize=(18,6),fontsize=12)
plt.title('Perbandingan Tipe Universitas Top 400 pada 2022',fontsize=16)
plt.xlabel('Frekuensi',fontsize=16)
plt.ylabel('Tipe Universitas',fontsize=16)
plt.show()
df['type'] = df['type'].map({'Swasta':'Private','Umum':'Public'})
```

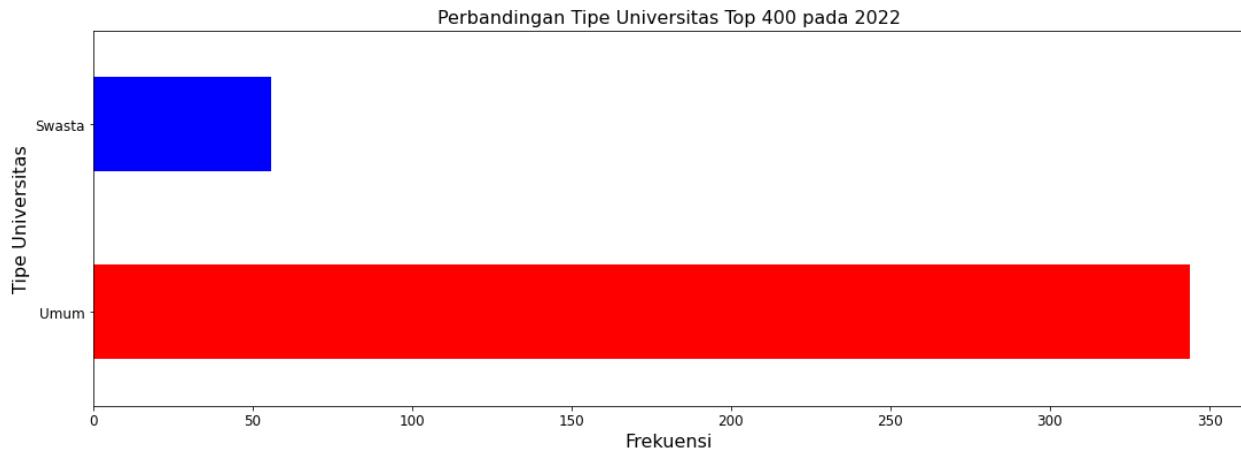


Diagram batang horizontal antara tipe universitas dan jumlahnya. Dari diagram ini kita dapat melihat perbandingan banyak universitas dengan tipe swasta dan umum dengan anggapan berdasarkan observasi bahwa universitas terbaik umumnya dimiliki oleh negara.

### 1.c. Banyak Murid Internasional untuk Setiap Negara yang Memiliki Universitas Top 400 Pada 2022:

```
df1=df[1987:2387].groupby('country')['international_students'].sum().reset_index(name='sum')
df1=df1.sort_values(['sum'],ascending=[1])
df1.set_index('country',inplace=True)
df1.plot(kind='barh',figsize=(16,16),color='darkcyan',legend=None)
plt.title('Banyak Murid Internasional untuk Setiap Negara yang Memiliki Universitas Top 400 pada 2022')
plt.xlabel('Negara',fontsize=12)
plt.ylabel('Jumlah Murid Internasional',fontsize=12)
plt.show()
```

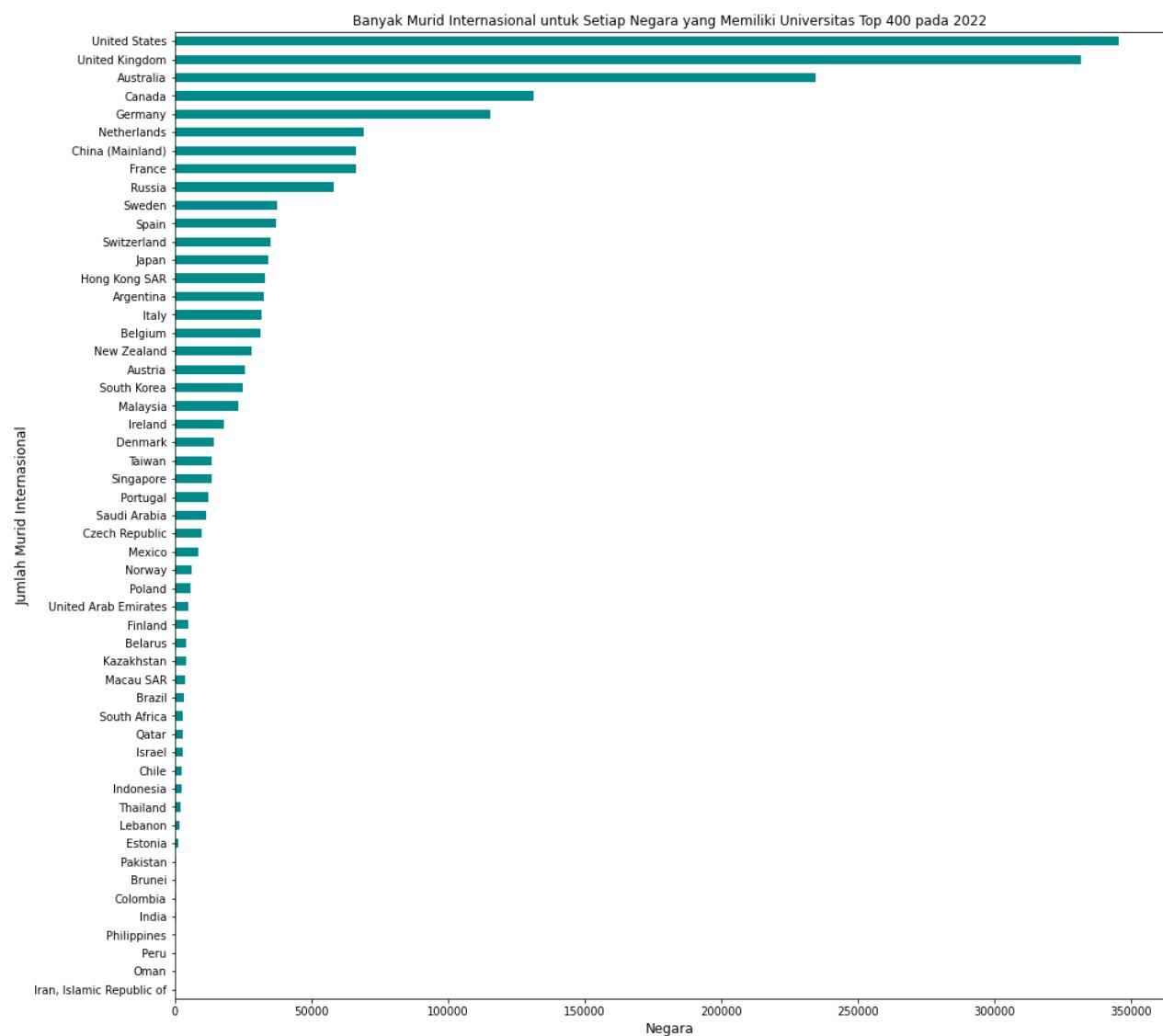


Diagram batang horizontal antara negara dan jumlah murid internasional untuk universitas top 400 QS World. Disini kita dapat melihat banyak murid internasional untuk setiap negara

dengan United States Memiliki Jumlah murid internasional terbanyak dan disusul oleh United Kingdom dan Australia.

## 2. Penampilan Perubahan terhadap Waktu

### 2.a. Perubahan banyak universitas top 400 di setiap wilayahnya terhadap tahun:

```
| df['region'] = df['region'].map({'North America':'Amerika Utara','Europe':'Eropa','Asia':'Asia',  
                                'Oceania':'Oceania','Latin America':'Amerika Latin','Africa':'Afrika'})  
df.groupby('year')[['region']].value_counts().unstack().plot(kind='area',color=('black','gold',  
                                'lightcoral','yellowgreen',  
                                'blueviolet','lightblue'),  
figsize=(18,8),fontsize=12,stacked=True,legend=True)  
plt.title('Banyak Universitas Top 400 di Setiap Wilayah ',fontsize=16)  
plt.xlabel('Wilayah',fontsize=16)  
plt.ylabel('Jumlah Universitas',fontsize=16)  
plt.legend(title='Wilayah:',loc="center right")  
plt.show()  
df['region'] = df['region'].map({'Amerika Utara':'North America','Eropa':'Europe','Asia':'Asia',  
                                'Oceania':'Oceania','Amerika Latin':'Latin America','Africa':'Africa'})
```

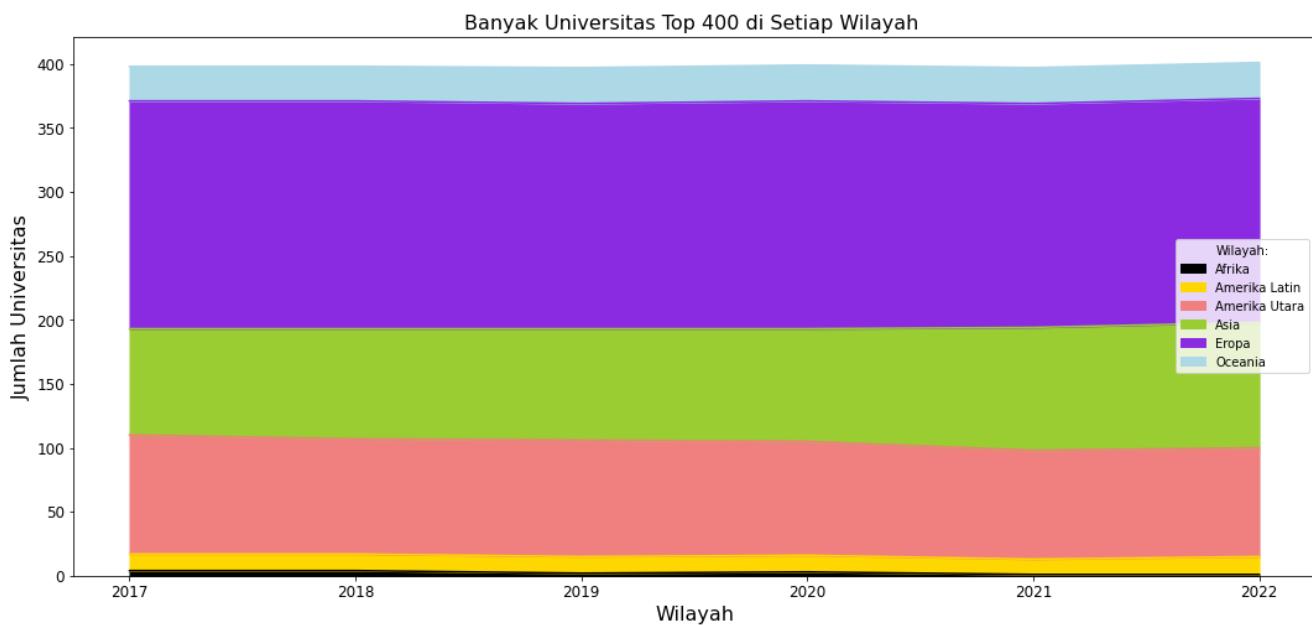


Diagram area perubahan banyak universitas top 400 di setiap wilayah terhadap rentang waktu 2017-2022. Di diagram ini kita dapat melihat dibandingkan wilayah lainnya, Eropa selalu memiliki universitas top 400 yang jauh lebih banyak.

## 2.b. Rata-Rata Jumlah Score Universitas Top 400 setiap tahunnya:

```
df.groupby('year')['score'].mean().plot(kind='line',marker='*',markersize=12,linestyle='-.',figsize=(16,8))
plt.xlabel('Tahun',fontsize=16)
plt.ylabel('Rata-rata Score',fontsize=16)
plt.title('Rata-Rata Jumlah Score Universitas Top 400 setiap tahunnya',fontsize=16)
plt.plot()
```

Diagram garis perubahan score rata-rata universitas top 400 dari tahun 2017 ke tahun 2022.

Dapat dilihat pada tahun 2019 score rata-rata universitas top 400 mengalami penurunan yang cukup signifikan.

## 3. Penampilan Hierarki dan Hubungan Keseluruhan-Bagian

### 3.a. Total Murid Internasional pada Setiap Wilayah di tahun 2022:

```
fracs=df[1987:2387].groupby('region')['international_students'].sum()
total=sum(fracs)
wilayah1=['Afrika','Asia','Eropa','Amerika Latin','Amerika Utara','Oceania']

#3.a.1. Persentase
df[1987:2387].groupby('region')['international_students'].sum().plot(kind='pie',figsize=(16,16),fontsize=14,autopct='%.2f%%',
                                                               colors=('black','yellowgreen','blueviolet','gold','lightcoral','lightblue'),
                                                               explode=(0.1,0.025,0.05,0,0,0.025),shadow=True,startangle=90,labels='( ',' ',',',',',',',',')')
plt.xlabel('Wilayah',fontsize=16)
plt.ylabel('Persentase Total Murid Internasional',fontsize=16)
plt.title('Persentase Total Murid Internasional pada Setiap Wilayah di tahun 2022',fontsize=16)
plt.legend(title='Wilayah:',labels=wilayah1)
plt.show()

#3.a.2. Jumlah
df[1987:2387].groupby('region')['international_students'].sum().plot(kind='pie',figsize=(16,16),fontsize=14,autopct=lambda p: '{:.0f}'.format(p * total / 100),
                                                               colors=('black','yellowgreen','blueviolet','gold','lightcoral','lightblue'),
                                                               explode=(0.1,0.025,0.05,0,0,0.025),shadow=True,startangle=90,labels='( ',' ',',',',',',',',')')
plt.xlabel('Wilayah',fontsize=16)
plt.ylabel('Jumlah Murid Internasional',fontsize=16)
plt.title('Total Murid Internasional pada Setiap Wilayah di tahun 2022',fontsize=16)
plt.legend(title='Wilayah:',labels=wilayah1)
plt.show()
```

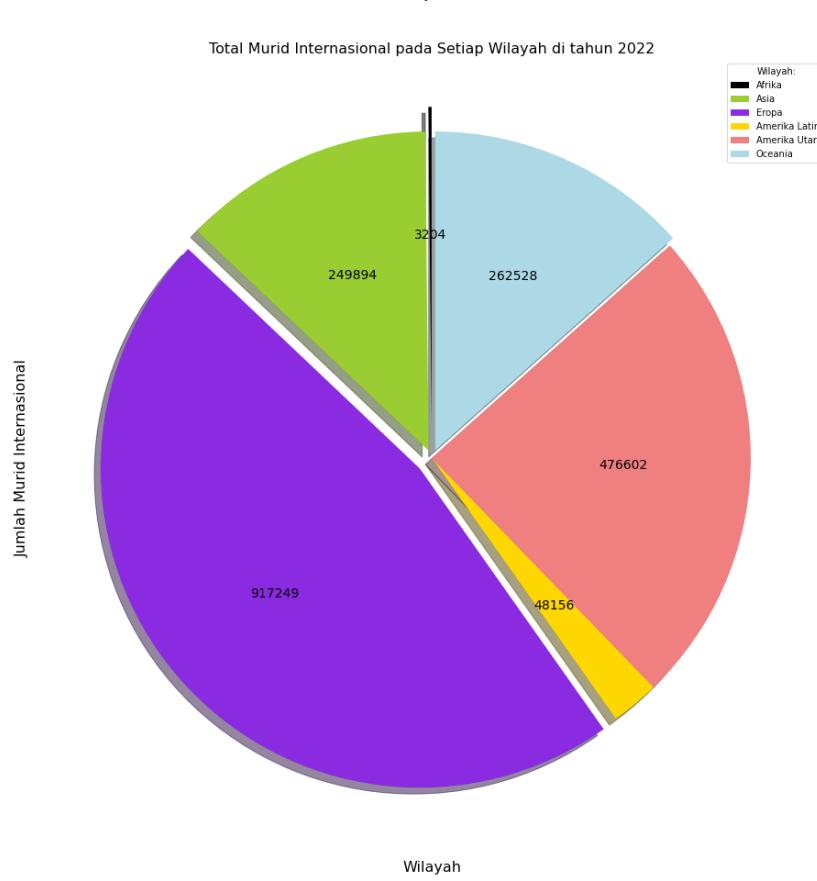
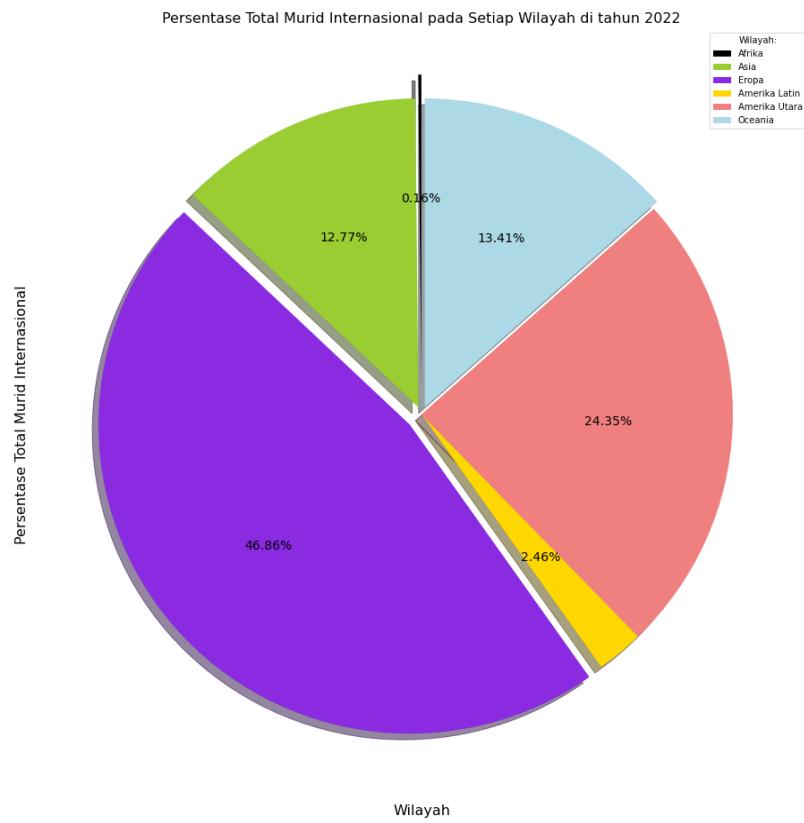
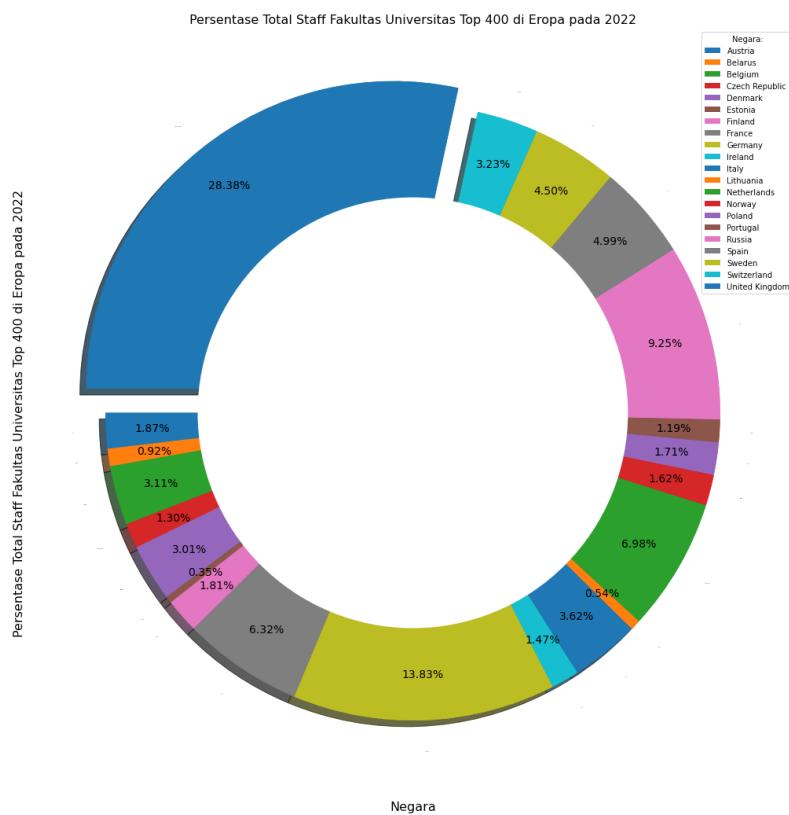


Diagram lingkaran pie yang memperlihatkan proporsi murid internasional setiap wilayah. Dengan ini dapat dinyatakan Eropa memiliki hampir setengah dari total murid internasional universitas top 400.

### **3.b. Total Staff Fakultas Universitas Top 400 di Eropa pada 2022:**



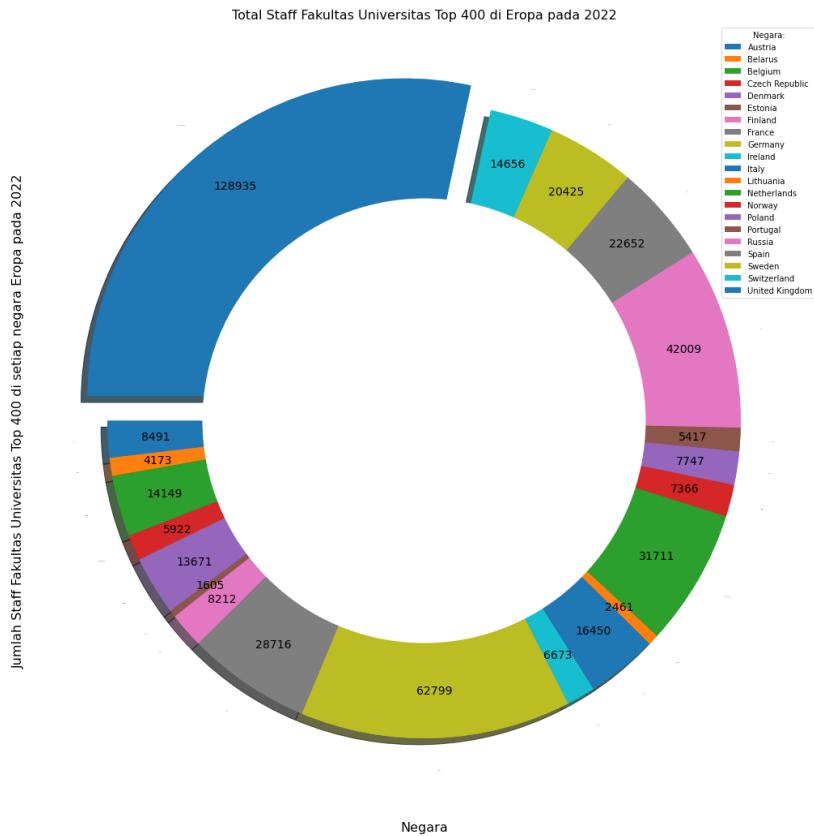
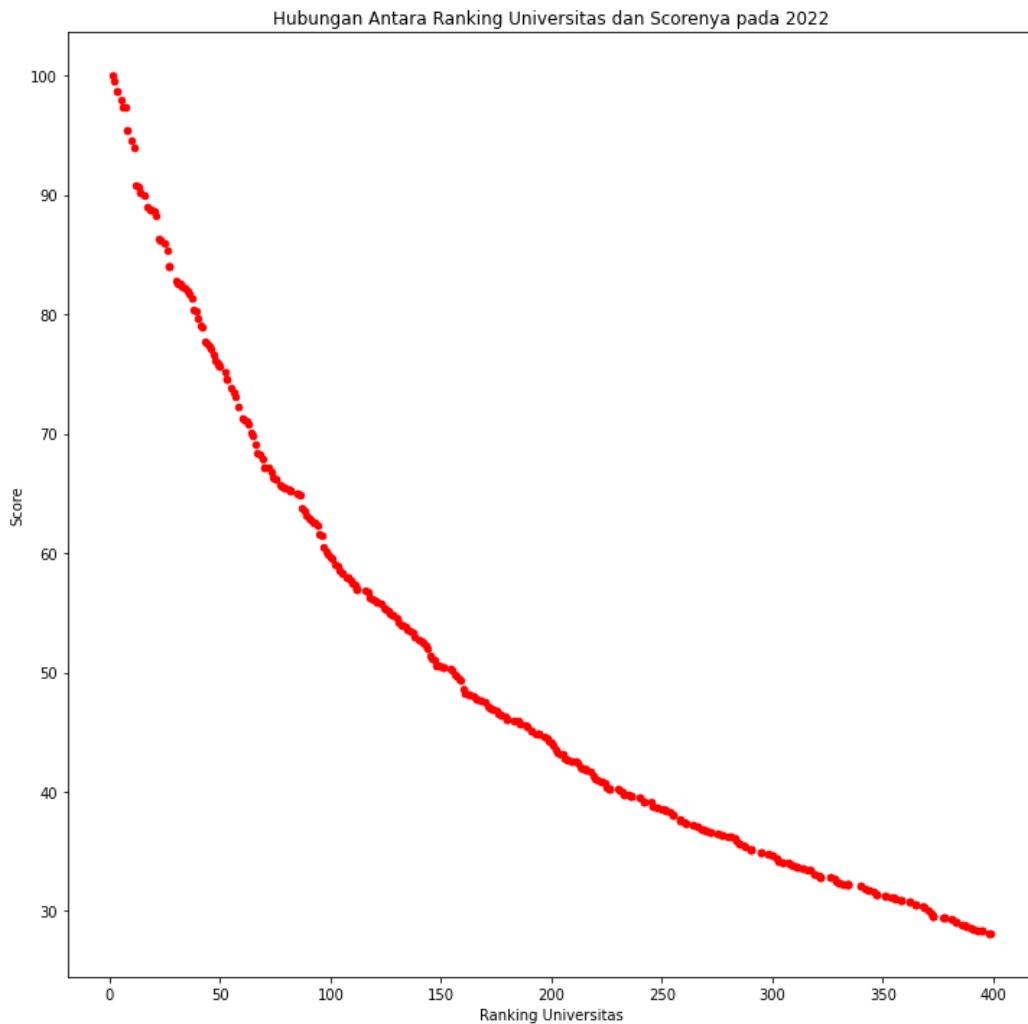


Diagram lingkaran donat yang memperlihatkan proporsi staf fakultas universitas top 400 di eropa pada 2022 untuk setiap negara. Terlihat negara United Kingdom memiliki jumlah staf fakultas terbanyak untuk universitas top 400 di Eropa pada 2022 dengan hampir 1/3 dari staff dimilikinya.

#### 4. Plotting relationships

##### 4.a. Hubungan Antara Ranking Universitas dan Skornya pada 2022:

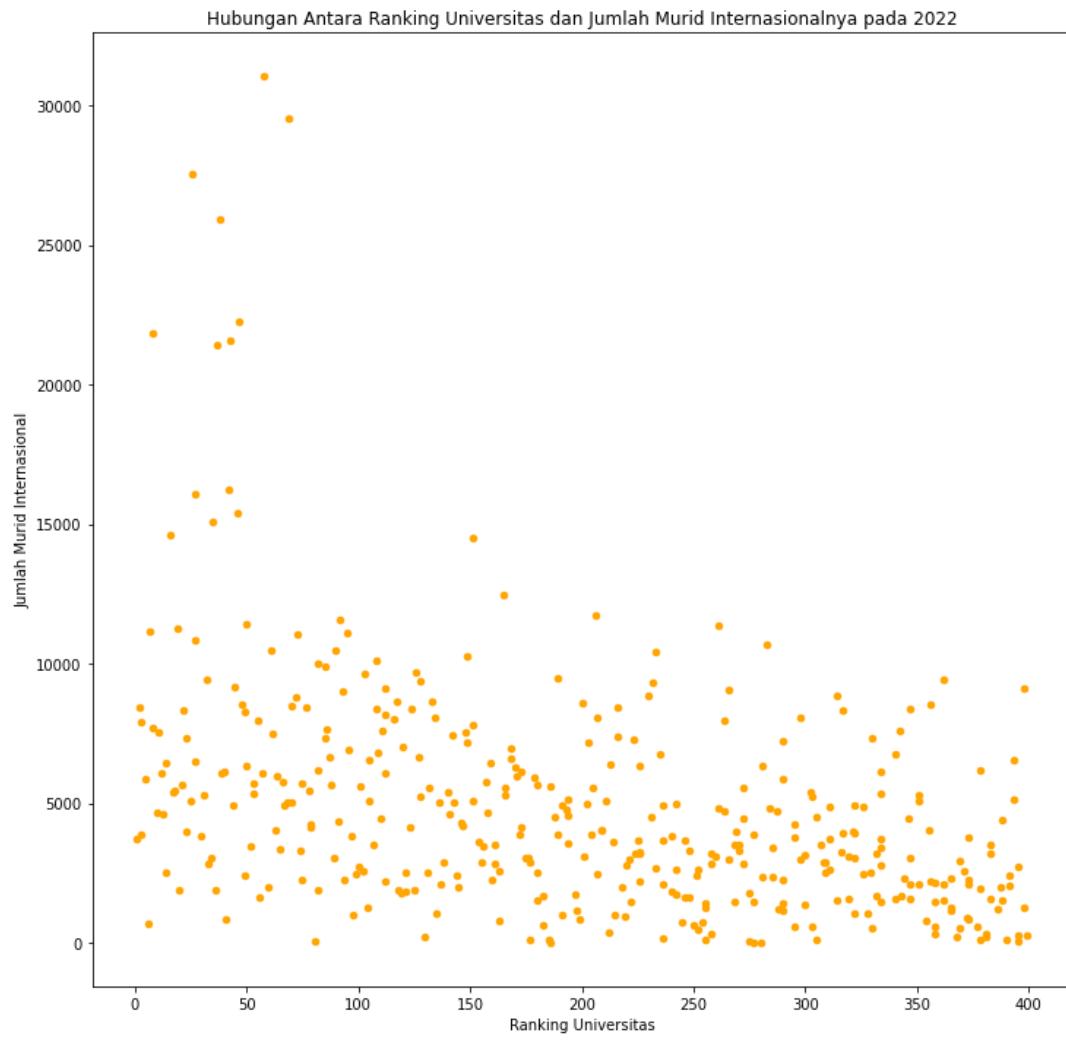
```
df[1987:2387].plot(kind='scatter',x='rank_display',y='score',figsize=(12,12),color='red')
plt.xlabel('Ranking Universitas')
plt.ylabel('Score')
plt.title('Hubungan Antara Ranking Universitas dan Scorenya pada 2022')
plt.show()
corr4=df[1987:2387]['rank_display'].corr(df['score'])
print(corr4)
```



Scatter plot antara ranking universitas dan skornya. Dari sini kita dapat melihat bahwa semakin tinggi skor sangat cenderung untuk ranking semakin besar (indeks mengecil).

#### 4.b. Hubungan Antara Ranking Universitas dan Jumlah Murid Internasionalnya pada 2022:

```
df[1987:2387].plot(kind='scatter',x='rank_display',y='international_students',figsize=(12,12),color='orange')
plt.xlabel('Ranking Universitas')
plt.ylabel('Jumlah Murid Internasional')
plt.title('Hubungan Antara Ranking Universitas dan Jumlah Murid Internasionalnya pada 2022')
plt.show()
corr4=df[1987:2387]['rank_display'].corr(df['international_students'])
print(corr4)
```



Scatter plot antara ranking universitas dan jumlah murid internasionalnya. Dari sini terlihat semakin tinggi ranking (semakin kecil angkanya) agak cenderung untuk semakin banyak murid internasionalnya.

# Korelasi

## 1. Nilai Korelasi

Untuk menentukan korelasi digunakan metode pearson, metode pearson adalah suatu ukuran untuk menentukan kekuatan hubungan dari dua data. Jika nilai korelasi dalam interval [-1,0), maka korelasi **berbanding terbalik**. Jika nilai korelasi sama dengan 0, maka **tidak ada korelasi**. Jika nilai korelasi dalam interval (0,1], maka korelasi **berbanding lurus**. Kemudian, apabila nilai korelasi mendekati 1 atau -1, maka **korelasi kuat** dan apabila nilai korelasi mendekati 0, maka **korelasi rendah**.

```
df.corr(method="pearson")
```

	year	rank_display	score	student_faculty_ratio	international_students	faculty_count
year	1.000000	-0.001465	-0.057670	-0.019758	-0.003053	0.004164
rank_display	-0.001465	1.000000	-0.944690	0.273907	-0.423241	-0.378660
score	-0.057670	-0.944690	1.000000	-0.316148	0.430244	0.405168
student_faculty_ratio	-0.019758	0.273907	-0.316148	1.000000	0.157065	-0.288218
international_students	-0.003053	-0.423241	0.430244	0.157065	1.000000	0.432507
faculty_count	0.004164	-0.378660	0.405168	-0.288218	0.432507	1.000000

## 2. Korelasi Berbanding Lurus

Korelasi dari dua variabel dari data ini yang berbanding lurus, diantaranya yaitu :

- rank\_display dengan student\_faculty\_ratio
- score dengan faculty\_count
- student\_faculty\_ratio dengan international\_students
- international\_students dengan score
- faculty\_count dengan international\_students

## 3. Korelasi Berbanding Terbalik

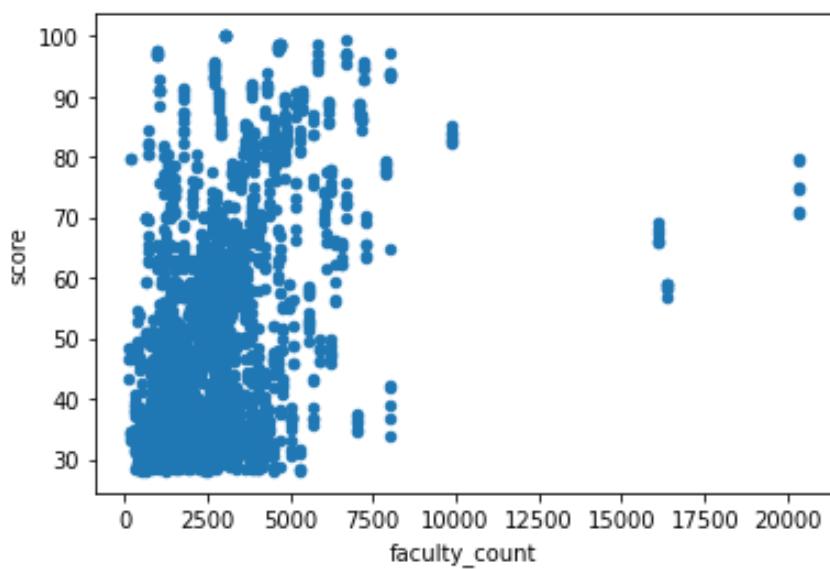
Korelasi dari dua variabel dari data ini yang berbanding terbalik, diantaranya yaitu :

- rank\_display dengan score
- score dengan student\_faculty\_ratio

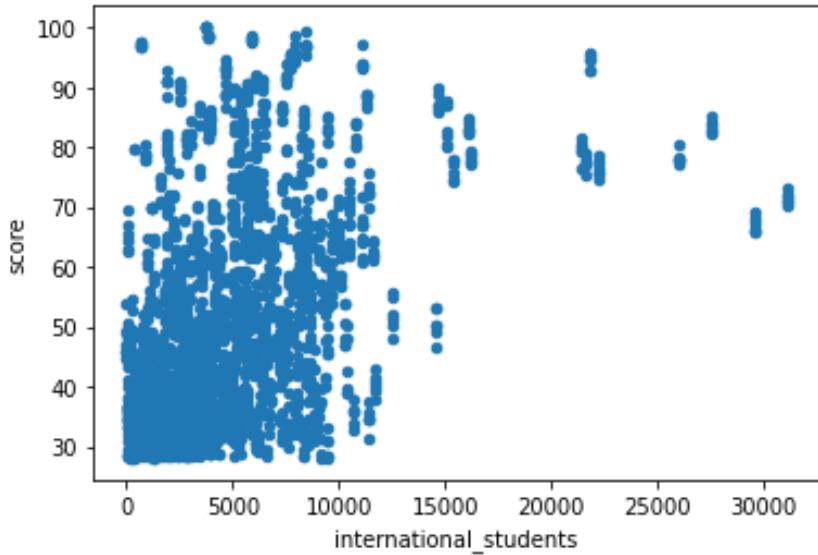
- student\_faculty\_ratio dengan faculty\_count
- international\_students dengan rank\_display
- faculty\_count dengan rank\_display

#### 4. Grafik Korelasi

```
df.plot(kind="scatter", x="faculty_count", y="score")
df_korelasi1 = df["faculty_count"].corr(df["score"])
print(f"Nilai korelasi = {df_korelasi1}")
```

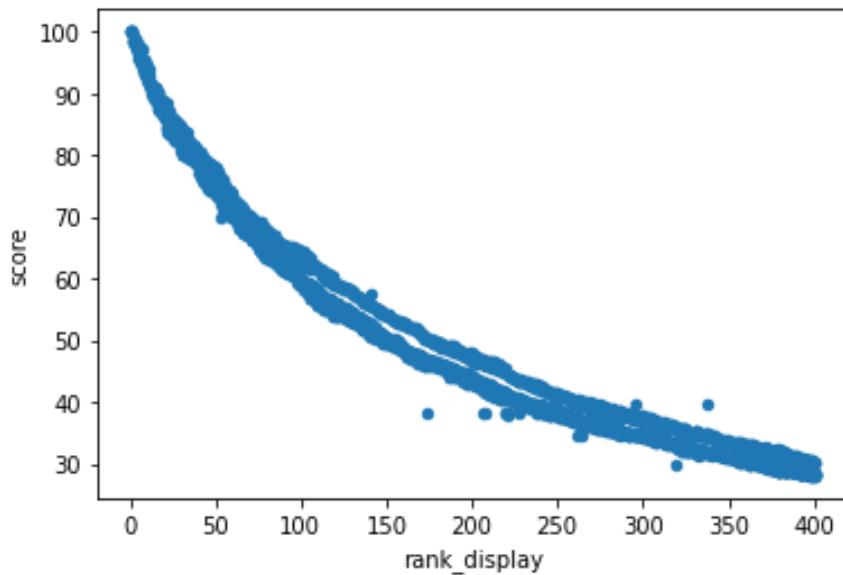


```
df.plot(kind="scatter", x="international_students", y="score")
df_korelasi2 = df["international_students"].corr(df["score"])
print(f"Nilai korelasi = {df_korelasi2}")
```

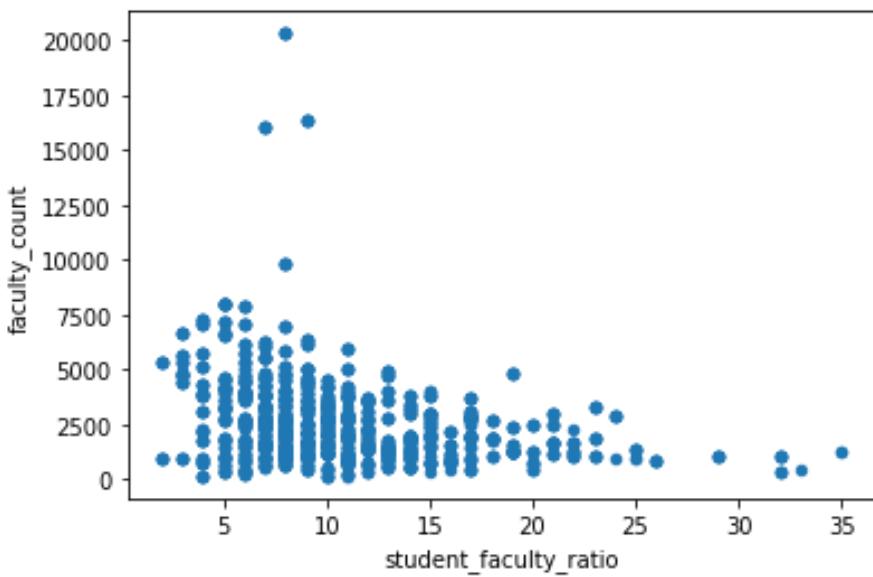


Dari dua *scatter plot* di atas dapat digambarkan bahwa faculty\_count dengan score dan international\_students dengan score yang berbanding lurus. Sehingga, semakin banyak fakultas pada suatu universitas, maka semakin tinggi nilai yang diperoleh. Begitu pula, jika semakin banyak murid internasional pada universitas tersebut, maka semakin tinggi nilai yang diperoleh. Walaupun nilai korelasi nya hanya 0.4051678 dan 0.4302439 yang menandakan korelasi atau hubungan yang lemah.

```
df.plot(kind="scatter", x="student_faculty_ratio", y="faculty_count")
df_korelasi3 = df["student_faculty_ratio"].corr(df["faculty_count"])
print(f"Nilai korelasi = {df_korelasi3}")
```



```
df.plot(kind="scatter", x="rank_display", y="score")
df_korelasi4 = df["score"].corr(df["rank_display"])
print(f"Nilai korelasi = {df_korelasi4}")
```



Dari dua *scatter plot* di atas yaitu korelasi antara student\_faculty\_ratio dengan faculty\_count dan rank\_display dengan score adalah berbanding terbalik. Sehingga, semakin tinggi rasio jumlah mahasiswa per jumlah pengajar, maka semakin kecil jumlah staff/pengajar fakultas pada universitas tersebut. Namun, mereka memiliki nilai korelasi sebesar -0.2882177 yang menandakan hubungan yang lemah. Kemudian, semakin rendah rank yang diperoleh, maka semakin kecil nilai yang diperoleh dengan nilai korelasi sebesar -0.9446902 yang menandakan hubungan yang kuat.

## **Kesimpulan**

Dari analisis data kami, dari rumusan masalah yang dicetuskan dapat disimpulkan bahwa

1. Benua Eropa memiliki jumlah universitas terbanyak yang masuk Top 400 Universitas.
2. United States memiliki jumlah universitas terbanyak yang masuk Top 400 Universitas.
3. Benua yang paling disukai atau memiliki jumlah mahasiswa internasional terbanyak adalah benua Eropa.
4. Negara yang paling disukai atau memiliki jumlah mahasiswa internasional terbanyak adalah United States.
5. Universitas pada Top 400 kebanyakan bertipe umum dibanding swasta.
6. Semakin banyak murid internasional pada universitas tersebut, maka semakin tinggi nilai yang diperoleh yang mengakibatkan semakin tinggi/atas ranking universitas tersebut. Namun, memiliki hubungan yang lemah.
7. Semakin banyak jumlah staff/pengajar di suatu universitas, maka semakin tinggi/atas ranking yang diperoleh universitas tersebut. Namun, memiliki hubungan yang lemah.
8. Tidak ada korelasi dari jumlah staff/pengajar terhadap ranking suatu universitas

## **Daftar Pustaka**

<https://www.kaggle.com/datasets/padhmam/qs-world-university-rankings-2017-2022>, diakses pada 25 November 2022

<https://pandas.pydata.org/docs/>, diakses pada 25 November 2022

<https://matplotlib.org/stable/index.html>, diakses pada 25 November 2022

## **Pembagian Tugas**

Dewantoro Triatmojo (19622152): Membuat Deskripsi Data dan File, Deskripsi Karakteristik Data, Data Preprocessing dan Data Cleansing.

Berto Ricardo Togatorop (19622192): Membuat Statistika.

Randy Verdian (19622202): Membuat Korelasi.

Aira Ardistya A (16522062): Membuat Visualisasi.