

Pembangunan Visualisasi Data Menggunakan Bahasa Pemrograman Python dengan Dataset Bidang Keuangan

Laporan

Diajukan Sebagai Syarat Menyelesaikan Tugas Besar Mata Kuliah Analisis Visualisasi Data pada Departemen Sistem Informasi

**Dosen Pengampu :
Ricky Akbar, M.Kom
Rahmatika Pratama Santi, M.T**



Kelompok 7

- | | |
|-------------------------------|---------------------|
| 1. Nadia Nur Saida | (2111521007) |
| 2. Al - Amin | (2111521017) |
| 3. Rafiqatul Ulya | (2111522003) |
| 4. Diqzi Aprinaldi | (2111522026) |
| 5. Ghina Fitri Hidayah | (2111523015) |

**Departemen Sistem Informasi
Fakultas Teknologi Informasi
Universitas Andalas
2023**

KATA PENGANTAR

Alhamdulillahirabbil Alamin, Puji syukur kehadiran Allah SWT. Atas segala rahmat dan hidayah-Nya. Shalawat serta salam senantiasa tercurahkan kepada junjungan Nabi Muhammad SAW yang selalu kita nantikan syafaatnya di akhirat nanti. Puji syukur kami panjatkan kehadiran Allah SWT, karena atas berkat dan rahmat-Nya kami dapat menyelesaikan penyusunan Laporan Tugas Besar Analisis Visualisasi Data ini dengan baik.

Penyusunan laporan tugas besar ini dilakukan untuk menjadi syarat penyelesaian Tugas Besar Mata Kuliah Analisis Visualisasi Data, Departemen Sistem Informasi Universitas Andalas. Dalam penyusunan tugas ini, kami banyak memperoleh bantuan dan masukan dari berbagai pihak, baik secara langsung maupun tidak langsung. Oleh karena itu, dalam kesempatan ini kami mengucapkan terima kasih kepada:

1. Tuhan Yang Maha Esa
2. Ricky Akbar, M. Kom, dan Rahmatika Pratama Santi, M.T selaku dosen pengampu mata kuliah ini
3. Rekan-rekan mahasiswa dan sumber lain yang menjadi referensi penulis dalam menyusun laporan ini.

Kami menyadari bahwa Tugas Besar Analisis Visualisasi Data ini masih belum sempurna, oleh karena itu segala kritik dan saran yang bersifat membangun sangat penulis harapkan. Semoga laporan ini dapat berguna bagi pembaca maupun untuk kami sendiri.

Padang, 14 Mei 2023

Penulis

DAFTAR ISI

KATA PENGANTAR.....	2
DAFTAR ISI.....	3
BAB I.....	4
PENDAHULUAN.....	4
1.1 Latar Belakang	4
1.2 Rumusan Masalah	5
1.3 Batasan Masalah.....	5
1.4 Tujuan	5
BAB II	6
TINJAUAN PUSTAKA	6
2.1 Konsep Dasar AVD	6
2.1.1 Pengertian Analisis Visualisasi Data	6
2.1.2 Manfaat dan Tujuan Analisis Visualisasi Data.....	6
2.1.3 Langkah Langkah Analisis Visualisasi Data	7
2.2 Tools Pengolahan Data	8
2.2.1 Bahasa Pemrograman	8
2.2.2 Library Python	10
2.2.3 Aplikasi Pengolahan Data.....	11
2.3 Dataset	12
2.3.1 Penjelasan Umum Dataset	12
2.3.2 Penjelasan Atribut Dataset.....	12
BAB III.....	15
HASIL ANALISIS	15
3.1 Coddington Program.....	15
3.2 Hasil Eksekusi Diagram dan Penjelasan Program	24
BAB IV	54
PENUTUP.....	54
4.1 Kesimpulan	54
4.2 Saran.....	54
DAFTAR PUSTAKA	55

BAB I

PENDAHULUAN

1.1 Latar Belakang

Data adalah suatu fakta atau observasi mentah yang digunakan untuk memberikan sebuah informasi kepada pihak tertentu. Data mengacu pada penjelasan peristiwa, aktivitas, dan transaksi dasar yang telah direkam, diklasifikasi, dan disimpan tetapi belum diatur untuk menyampaikan arti secara spesifik (R. Kelly Rainer, 2009). Sedangkan informasi adalah suatu data yang diolah sehingga dapat dijadikan dasar untuk mengambil sebuah keputusan yang tepat (George H. Bodnar, 2004).

Dengan dibutuhkannya informasi dari data yang dapat digunakan dalam kegiatan bisnis maka perlu dilakukan eksplorasi pada data untuk dapat diketahui informasinya. Ketika data menjadi terlalu sulit untuk dipahami, visualisasi data adalah jawaban untuk menyederhanakan data kompleks menjadi format grafis sehingga lebih mudah untuk dipahami. Visualisasi Data telah menjadi bagian tak terpisahkan dari dunia bisnis dan bagian yang semakin meningkat dalam mengelola kehidupan sehari-hari. (CIO Review Magazine, 2018). Visualisasi data adalah istilah umum yang menggambarkan setiap upaya untuk membantu orang memahami signifikansi data dengan menempatkan data dalam konteks visual. Visualisasi data menerjemahkan yang kompleks, bervolume tinggi, atau numerik menjadi representasi visual yang lebih mudah diproses. Alat visualisasi data meningkatkan dan mengotomatiskan proses komunikasi visual untuk mendapatkan akurasi dan detail.

Bahasa pemrograman Python sebagai salah satu pemrograman terbaik untuk menangani visualisasi data karena memiliki dukungan komunitas yang aktif dan memiliki banyak pustaka bantu terutama dalam hal komputasi cerdas. Bahasa pemrograman mampu menangani data dalam jumlah besar dengan performa yang baik

Penelitian ini dilakukan terhadap dataset keuangan sebuah platform belajar online yaitu Udemy. Udemy adalah platform online yang menyediakan kursus video interaktif dan tutorial dalam berbagai bidang. Platform ini

dioperasikan oleh Udemy, Inc., sebuah perusahaan teknologi berbasis di San Francisco, California. Kursus di Udemy mencakup berbagai topik, mulai dari teknologi, bisnis, hingga desain dan bahasa asing. Kursus dapat dipelajari secara online, atau diunduh untuk ditonton nanti. Udemy juga menyediakan alat bantu untuk membangun kursus dan melacak kemajuan pelajar. Di seluruh dunia, lebih dari 40 juta orang telah belajar dari 52.000 instruktur dengan total 130.000 kursus yang disampaikan dalam 60 bahasa lebih. Dengan banyaknya pengguna platform tersebut maka penulis akan mencoba membangun visualisasi dari dataset keuangan Udemy ini agar lebih mudah untuk dipahami.

1.2 Rumusan Masalah

Berdasarkan latar belakang penelitian yang disampaikan diatas, maka dapat diidentifikasi permasalahannya adalah bagaimana cara membangun visualisasi dari sebuah dataset keuangan platform belajar Udemy menggunakan bahasa pemrograman python?

1.3 Batasan Masalah

Agar luaran tugas besar ini tidak terlalu jauh nantinya dan tidak menyimpang dari rumusan masalah diatas, maka perlu adanya pembatasan masalah yang ditinjau. Batasan masalahnya adalah sebagai berikut:

1. Peneliti menggunakan sampel data keuangan untuk merepresentasikan penelitian ini.
2. menggunakan pustaka yang dimiliki python untuk menampilkan visualisasi data.

1.4 Tujuan

Tujuan penelitian ini adalah membuat visualisasi dataset keuangan dari sebuah platform belajar yaitu Udemy, sehingga data yang ditampilkan lebih mudah dipahami dan mendapatkan sebuah informasi.

BAB II

TINJAUAN PUSTAKA

2.1 Konsep Dasar AVD

2.1.1 Pengertian Analisis Visualisasi Data

Analisis Visualisasi Data merupakan salah satu mata kuliah departemen sistem informasi. Mata kuliah ini mempelajari teknik visualisasi data dengan berbagai tools. Nantinya diharapkan dapat memvisualisasi data mentah yang diberikan agar mudah dipahami oleh orang banyak.

Pengertian dari visualisasi data sendiri ialah tampilan berupa grafis atau visual informasi dan data (Tableau). Visualisasi data mengacu pada teknik yang dipakai kepada mengkomunikasi data atau informasi dengan menciptakannya sebagai objek visual (misalnya : titik, garis, batang) dalam grafik

2.1.2 Manfaat dan Tujuan Analisis Visualisasi Data

Analisis Visualisasi Data sangat diperlukan implementasinya dalam kehidupan sehari - hari, sehingga manfaat dari visualisasi data adalah :

- Memperlihatkan tren
- Menampilkan konteks dari sebuah informasi
- Mempersingkat waktu baca
- Memperkuat narasi
- Memperjelas rangkaian cerita secara kronologis (waktu ke waktu)
- Memberikan perspektif pada sebuah cerita
- Menjelaskan sebuah proses (tahap per tahap)
- Menstimulasi imajinasi pembaca

Sedangkan tujuan dari adanya visualisasi data :

- a. Untuk mengkomunikasikan informasi secara jelas dan efisien kepada pengguna lewat grafik informasi yang dipilih, seperti tabel dan grafik
- b. Visualisasi yang efektif membantu pengguna dalam menganalisa dan penalaran tentang data dan bukti
- c. Dapat melaksanakan pekerjaan dengan sifat analisis tertentu. Seperti melaksanakan perbandingan atau memahami sebab akibat dari

perancangan grafik (visualisasi data sangat berguna dalam hal *comparasion*).

- d. Membantu eksplorasi data. Dalam konteks statistika dan data science, proses analisis data menggunakan grafik merupakan proses yang vital; Proses ini masuk ke dalam tahap Exploratory Data Analysis (EDA); Visualisasi data yang efektif merupakan langkah awal dalam sebuah analisis data yang baik.
- e. Metode Analisis Data. Konsep ini memunculkan istilah visual analytics; Visual analytics atau analisis visual merupakan ilmu untuk melakukan penalaran analitik (*reasoning analytics*) dengan didukung oleh antarmuka visual yang interaktif; memungkinkan pembuat keputusan menggunakan fleksibilitas, kreativitas, dan pengetahuan latar belakang mereka.

2.1.3 Langkah Langkah Analisis Visualisasi Data

Saat melakukan Visualisasi Data tentunya ada beberapa tahapan yang perlu diperhatikan saat kita hendak melakukan visualisasi data adalah sebagai berikut :

1. Menyiapkan data: Data harus disiapkan terlebih dahulu, termasuk membersihkan data dari nilai kosong atau outlier, melakukan normalisasi atau standarisasi, dan melakukan pemrosesan data lainnya sesuai kebutuhan.
2. Memilih teknik visualisasi: Berdasarkan jenis data dan tujuan analisis, memilih teknik visualisasi yang tepat, seperti scatter plot, line chart, bar chart, heatmap, atau teknik visualisasi lainnya.
3. Membuat visualisasi: Setelah teknik visualisasi dipilih, visualisasi dapat dibuat dengan menggunakan perangkat lunak atau bahasa pemrograman tertentu, seperti Python, R, atau Tableau.
4. Menganalisis hasil visualisasi: Setelah visualisasi selesai dibuat, hasil visualisasi dapat dianalisis untuk mendapatkan insight atau informasi baru yang berguna. Hal ini dapat dilakukan dengan melihat pola atau trend dari visualisasi, atau dengan melakukan analisis lebih lanjut menggunakan teknik analisis data lainnya.

5. Melakukan iterasi: Proses analisis visualisasi data adalah sebuah proses iteratif, yang berarti bahwa dapat dilakukan beberapa kali untuk mendapatkan insight yang lebih dalam atau memperbaiki visualisasi yang kurang efektif.

2.2 Tools Pengolahan Data

2.2.1 Bahasa Pemrograman

Guido Van Rossum, seorang programmer berkebangsaan Belanda, menciptakan Python. Awalnya di tahun 1989, ia memulainya di Centrum Wiskunde & Informatica (CWI) sebagai proyek hobi agar tetap sibuk selama waktu Natal. Nama untuk bahasa ini terinspirasi dari acara TV BBC, Monty Python's Flying Circus karena Guido Van Rossum adalah penggemar acara tersebut. Pada tahun 1991, ia mempublikasikan versi pertama kode python (versi 0.9.0). Kemudian pada tahun 1994, dirilisnya python 1.0 dengan fungsi baru untuk mempermudah proses pencantuman data (pemetaan, pemfilteran dan pengurangan). Pada 16 Oktober 2000, dirilisnya python 2.0 dengan fitur baru seperti dukungan untuk karakter Unicode dan me-loop daftar secara lebih singkat. Terakhir pada 3 Desember 2008, dirilisnya python 3.0 yang mencakup isi fitur seperti fungsi cetak dan lebih banyak dukungan untuk mengatasi masalah.

Bahasa Pemrograman yang digunakan dalam pengolahan dan visualisasi dataset pada pembuatan laporan ini ialah bahasa pemrograman Python. Python merupakan salah satu bahasa pemrograman yang populer yang digunakan dalam pengolahan data. Namun, Python juga banyak digunakan dalam pembuatan aplikasi web, pengembangan perangkat lunak, ilmu data, dan Machine Learning. Keunggulan menggunakan bahasa pemrograman Python karena efisien dan mudah untuk dipelajari serta dapat dijaankan di berbagai platform. Perangkat lunak Python dapat diunduh secara gratis, terintegrasi baik dengan semua tipe sistem, dan meningkatkan kecepatan pengembangan.

Ada beberapa yang menjadi alasan mengapa bahasa pemrograman Python menjadi bahasa pemrograman yang cocok untuk pengolahan data, diantaranya :

a. Mudah di pelajari

Bahasa pemrograman python memiliki syntax yang sangat mudah dipelajari dan dipahami, sehingga sangat cocok untuk pemula yang ingin mempelajari pengolahan data.

b. Memiliki banyak library untuk pengolahan data

Python memiliki banyak library seperti NumPy, Pandas, dan Matplotlib yang dirancang khusus untuk pengolahan data. Library-library ini menyediakan fungsi-fungsi yang sangat bermanfaat untuk melakukan analisis data, visualisasi, manipulasi, dan pengolahan data lainnya.

c. Dukungan untuk analisis statistik

Python juga menyediakan library yang lengkap untuk analisis statistik seperti SciPy, Statsmodels, dan Scikit-learn. Library-library ini memudahkan pengguna dalam melakukan analisis data dan model prediksi.

d. Kemampuan untuk bekerja dengan data mentah

Python mampu bekerja dengan data mentah yang bersifat terstruktur maupun tidak terstruktur. Python juga mampu mengolah data yang besar dan kompleks dengan mudah.

e. Cocok untuk machine learning

Python juga menjadi bahasa pemrograman yang populer untuk machine learning dan deep learning. Library seperti Tensorflow, Keras, dan PyTorch membuat Python menjadi pilihan yang tepat untuk pengolahan data yang memerlukan teknik machine learning.

Jadi, bahasa pemrograman Python merupakan bahasa pemrograman yang sangat cocok digunakan dalam pengolahan data. Python memiliki library dan tools yang lengkap untuk mengolah data dan analisis statistik data, serta mudah digunakan dan dipelajari pemula.

2.2.2 Library Python

Library python adalah sekumpulan kode yang sering digunakan yang dapat disertakan dalam pemograman python. Library ini merupakan kode yang sudah dibuat dan tersedia secara terbuka. Fungsi dari library python adalah untuk membantu dalam mengembangkan program atau aplikasi python agar mempercepat proses pengembangan karena tidak perlu lagi membuat fungsi atau metode dari awal untuk menyelesaikan suatu masalah. Library python menyediakan sekumpulan fungsi yang siap digunakan dan dapat membantu pengguna dalam menyelesaikan masalah atau tugas tertentu.

Bahasa pemograman python memiliki ribuan bahkan jutaan library yang tersedia. Berikut beberapa contoh library python yang sering digunakan dalam pengolahan dan pembuatan grafik data :

a. Numpy

Library Python yang digunakan untuk komputasi numerik dan ilmiah, termasuk manipulasi array dan matriks, operasi matematika, dan statistik.

b. Pandas

Library Python yang digunakan untuk manipulasi dan analisis data, termasuk mengimpor, memanipulasi, dan menyimpan data dalam berbagai format, serta melakukan analisis statistik dan visualisasi data.

c. Matplotlib

Library Python untuk visualisasi data, termasuk membuat grafik dan plot yang menarik dan informatif.

d. Scikit-learn

Library Python yang digunakan untuk pembelajaran mesin, termasuk algoritma klasifikasi, regresi, dan pengelompokan, serta evaluasi model dan pengolahan data.

e. TensorFlow

Library Python yang digunakan untuk pembelajaran mesin dan deep learning, termasuk membangun dan melatih model neural network.

f. Keras

Library Python yang digunakan untuk pembelajaran mesin dan deep learning, dengan fokus pada membangun dan melatih model neural network yang mudah digunakan dan dipahami.

g. SciPy

Library Python untuk ilmu pengetahuan dan teknik, termasuk algoritma numerik, integrasi, optimasi, dan sinyal.

h. OpenCv

Library Python untuk komputer visi, termasuk deteksi objek, pengenalan wajah, pengolahan citra, dan penglihatan mesin.

2.2.3 Aplikasi Pengolahan Data

Aplikasi pengolahan data merupakan perangkat lunak yang dirancang untuk membantu pengguna dalam mengumpulkan, memproses, menyimpan, dan menganalisis data. Ada banyak jenis aplikasi pengolahan data yang tersedia. Aplikasi yang digunakan dalam pengolahan data pada laporan ini ialah Google Colab.

Google Colab adalah platform komputasi gratis berbasis cloud yang dikembangkan oleh Google. Platform ini dirancang untuk membantu pengguna membuat, menjalankan, dan berbagi notebook Jupyter yang berisi kode Python, hasil eksekusi kode, dan teks deskriptif yang menjelaskan alur kerja dan hasil. Notebook Jupyter adalah lingkungan interaktif untuk pengembangan dan penelitian data yang sangat populer di kalangan komunitas ilmiah dan pengembang. Google Colab menyediakan layanan cloud yang kuat untuk menjalankan notebook Jupyter secara gratis, dengan menggunakan mesin virtual yang disebut sebagai "runtime".

Google Colab menawarkan beberapa fitur, seperti akses ke GPU dan TPU untuk melatih model deep learning dan pemrosesan paralel yang memungkinkan pengguna untuk memproses data secara lebih cepat. Selain itu, Google Colab juga memudahkan pengguna untuk mengakses berbagai library dan framework Python populer seperti NumPy, Pandas, Matplotlib, TensorFlow, dan Keras. Karena Google Colab merupakan layanan cloud,

yang berarti bahwa pengguna harus terkoneksi ke internet yang baik agar dapat mengakses dan menggunakan platform ini.

2.3 Dataset

2.3.1 Penjelasan Umum Dataset

Dataset ini berisi kompilasi semua kursus yang terkait pada development (pengembangan) dari sekitar 13 ribu kursus yang tersedia di situs web Udey. Di bawah kategori development (pengembangan), ada kursus dari Keuangan, Akuntansi, Pembukuan, Compliance (penyesuaian), Cryptocurrency, Blockchain, Ekonomi, Investasi & Perdagangan, Pajak dan banyak lagi yang masing-masing memiliki banyak kursus di bawah domainnya.

Link Dataset : <https://www.kaggle.com/datasets/jilkothari/finance-accounting-courses-udemy-13k-course>

2.3.2 Penjelasan Atribut Dataset

Pada data yang dianalisa terdapat 17 kolom serta terdapat dua kolom yang ditambahkan melalui kodingan, sehingga terdapat 19 buah kolom. Kolom tambahan berguna untuk memperjelas data yang telah ada. Berikut 19 Atribut pada dataset Udey:

NO	Nama_Kolom	Fungsi
1	Id	Id dari course yang ada
2	Title	judul dari akademi yang ada di Udey
3	url	url dari course
4	Is_paid	Berisikan boolean berdasarkan keterangan coursanya berbayar atau tidak
5	Num_subscribers	Jumlah orang yang berlangganan

NO	Nama_Kolom	Fungsi
		course ini
6	Avg_rating	Rata-rata rating dari coursenya
7	Avg_rating_recent	Perubahan terbaru dari peringkat rata rata
8	num_reviews	Jumlah orang yang memberikan pendapat terkait kursusnya
9	num_published_lectures	Jumlah pelajaran yang ada pada pada kursus tersebut
10	num_published_practice_test	Jumlah latihan yang ada pada kursus tersebut
11	created	Waktu pembuatan kursus
12	published_time	Waktu publish kursus
13	discounted_price_amount	Diskon yang ditawarkan pada kursus
14	discount_price_currency	Mata uang yang digunakan pada diskon
15	price_detail_amount	Harga awal dari kursus
16	price_detail_currency	Mata uang yang digunakan pada harga awal
17	price_detail_string	Harga awal kursus dalam bentuk string
18	after_discount	Harga setelah dikurangi diskon
19	total	total pemasukan yang didapatkan dari

NO	Nama_Kolom	Fungsi
		kursus

BAB III

HASIL ANALISIS

3.1 Coddigan Program

```
# import Library yang diburuhkan dalam visualisasi data
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns

# read data udemy.csv
df = pd.read_csv('udemy (1).csv')
df = df.dropna(how='any',axis=0)
df

# mengecek tipe data setiap atribut
tipe = df.dtypes
print(tipe)

# menambahkan data baru
# after_discount harga dari setiap judul setelah dikurangi
discount
after_discount = df['price_detail__amount'] -
df['discount_price__amount']
df['after_discount'] = after_discount
# total yaitu jumlah pendapatan yang didapatkan dari setiap
judul
total = df['after_discount'] * df['num_subscribers']
df['total'] = total
df

# top10 title dengan sumber pendapatan terbanyak
sort = df.sort_values(by=["total"], ascending= False)
%matplotlib inline
# ambil sampel df 10 baris
sort_10= sort.head(10)
# sort_10.plot.barh(x='title', y='total',
# title='Cuisine Preference', color='green')
fig = plt.figure(figsize=(11,3))
ax = fig.add_axes([0,0,1,1])
ax.set_xlabel('title')
ax.set_ylabel('total_pendapatan (miliar)')
ax.bar(sort_10['title'],sort_10['total'])
plt.xticks(rotation=90)
plt.title('Top10 Sumber Pendapatan Terbesar')
```

```

plt.show()

# menampilkan korelasi antar entitas menggunakan heatmap
plt.title('Korelasi Matriks Dataset Udemmy')
sns.heatmap(data=df.corr())

# menampilkan diagram batang mengenai banyak kursus
berbayar berdasarkan published practice
plt.figure(figsize=(10,6))
plt.title("Banyaknya Kursus berbayar berdasarkan
published_practice")
plt.xlabel("rating")
plt.ylabel("published_practice")
# data.iloc yang akan ditampilkan
sns.countplot(y=df.iloc[4:,11])
plt.show()

# diagram scatterplot
plt.figure(figsize=(5,5),dpi=200)
sns.scatterplot(data=df, x='rating',
y='num_reviews',hue='is_paid',palette='Set1')
plt.title('Hubungan Rating dengan num_reviews berdasarkan
tipe kursus')

df = list(range(1, len(df)+1))
df = pd.DataFrame({'rating': df , 'is_paid': df })
df.head()

df1 = df[0:6345]
df1.head()

df2 = df[6345:12690]
df2.head()

# grafik line hubungan rating dengan kursus berbayar atau
tidak
fig = plt.figure(figsize=(6,4))
ax = plt.axes()
ax.plot(df1['rating'],df1['is_paid'])

# grafik line hubungan rating dengan kursus berbayar atau
tidak
fig = plt.figure(figsize=(6,4))
ax = plt.axes()
ax.plot(df2['rating'],df2['is_paid'])

```



```

# gabungan antara 2 tabel diatas
ax.set_title('Hubungan rating dengan is_paid')
ax.set_ylabel('is_paid')
ax.set_xlabel('rating')
fig = plt.figure(figsize=(6,4))
ax = plt.axes()
ax.plot(df1['rating'],df1['is_paid'],'co')
ax.plot(df2['rating'],df2['is_paid'],'m^')

#diagram pie presentase subscriber dengan kategori rating
%matplotlib inline
plt.title('Presentase subscriber dengan kategori
discount_price')
# ambil sampel df 10 baris
sort_10= sort.head(10)
df=(sort_10['discount_price_amount']).value_counts().plot.
pie(autopct='%1.1f%%',labels=None,legend=True)
ax.bar(sort_10['title'],sort_10['num_subscribers'])
plt.tight_layout()

# sortir 5title dan rating dengan diskon sedikit
x=df.sort_values(by='discount_price__amount')['title'].tail
(5)
y=df.sort_values(by='discount_price__amount')['rating'].tai
l(5)
plt.barh(x,y)

from sklearn.preprocessing import LabelEncoder
encode=LabelEncoder()
df['is_paid']=encode.fit_transform(df['is_paid'].values)

plt.title('Published Practice dengan Diskon yang didapat')
sns.boxplot(x="num_published_practice_tests", y=
"discount_price__amount", data=df)

# Menunjukkan jumlah kelas yang ditawarkan kursus beserta
jumlah subscriber nya
sns.set()
plt.figure(figsize = (16,9))
sns.scatterplot(x=df.loc[:, 'num_subscribers'],
y=df.loc[:, 'num_published_lectures'], palette = 'icefire',
hue=df.is_paid)

# menampilkan rumlah rivIEWS dan diskon kelas yang
diberikan
sns.set()
plt.figure(figsize = (16,9))

```

```

sns.scatterplot(x=df.loc[:, 'discount_price__amount'],
y=df.loc[:, 'num_reviews'], hue=df.is_paid,
palette='viridis')

# menambahkan data discount_amount_usd
df["price_amount_usd"] =
df["price_detail__amount"].apply(lambda x: x*0.014)
df["discount_amount_usd"] =
df["discount_price__amount"].apply(lambda x: x*0.014)
df.head()

# menampilkan data course yang dibayar (true) dan tidak
dibayar (false), berdasarkan jumlah subscriernya
sns.set()
plt.figure(figsize = (16,9))
sns.barplot(x=df.loc[:, 'is_paid'],
y=df.loc[:, 'num_subscribers'], palette = 'magma')

# menampilkan data course yang dibayar (true) dan tidak
dibayar (false), berdasarkan jumlah review
sns.set()
plt.figure(figsize = (16,9))
sns.barplot(x=df.loc[:, 'is_paid'],
y=df.loc[:, 'num_reviews'])

# menambahkan data tahun rilis suatu kursus
df['created_year'] = df['created'].apply(lambda x:
np.int(x[0:4]))
df['published_year'] = df['published_time'].apply(lambda x:
np.int(x[0:4]))
df

# menampilkan jumlah course yang di publish di setiap
tahunnya
sns.countplot(x= "published_year", data=
df).set_title('Jumlah Kursus Publish Per Tahun')

# menampilkan course yang dibayar(true) atau tidak
dibayar(false) dengan tidak menggunakan mata uang dollar
sns.set()
plt.figure(figsize = (16,9))
sns.barplot(x=df.loc[:, 'is_paid'],
y=df.loc[:, 'price_amount_usd'])

# menampilkan jumlah pelanggan berdasarkan pengaruh harga
course
sns.set()

```

```

plt.figure(figsize = (16,9))
sns.scatterplot(x=df.loc[:, 'num_subscribers'],
y=df.loc[:, 'price_amount_usd'], color = 'red')

# persentasi perbandingan jumlah kursus berbayar dan tidak
berbayar
paid_course_data = [len(df[df['is_paid']==True]),
                    len(df[df['is_paid']==False])]
paid_course_labels = ['Paid', 'Not Paid']
plt.title('Persentase pembayaran course Udemy')
plt.pie(paid_course_data, labels = paid_course_labels,
autopct='%.1f%%')
plt.show()

# persentase jenis mata uang yang berlaku pada harga diskon
mataUang =
[np.count_nonzero(df['price_detail__currency']=='INR'),
 np.count_nonzero(df['price_detail__currency']!='
INR')]
label = ['INR', 'non INR']
plt.title('Mata uang yang berlaku dengan harga Diskon')
plt.pie(mataUang, labels = label, radius=1.3,
startangle=60, shadow=True, autopct='%.1f%%')
plt.show()

df = df.sort_values('discount_price__amount',
ascending=False)
df =
df.groupby('discount_price__amount').first().reset_index()
top5=df.tail(5)
top5 = top5.iloc[::-1]
colors = ['red', 'green', 'blue', 'orange', 'purple']
# Membuat grafik
plt.bar(top5['title'],
top5['discount_price__amount'],color=colors)
plt.xlabel('Example Title')
plt.ylabel('Discount Price Amount')
plt.title('Top 5 Discount Amount')
plt.xticks(rotation=90)
# Menampilkan grafik
plt.show()

df = df.sort_values('discount_price__amount',
ascending=False)
df =
df.groupby('discount_price__amount').first().reset_index()
top5=df.tail(5)

```

```

top5 = top5.iloc[::-1]
colors = ['red', 'green', 'blue', 'orange', 'purple']

# Membuat grafik
plt.bar(top5['title'],
top5['discount_price__amount'],color=colors)
plt.xlabel('sample title')
plt.ylabel('Discount Price Amount')
plt.title('Top 5 Discount Amount')
plt.xticks(rotation=90)

# Menampilkan grafik
plt.show()

df = df.sort_values('discount_price__amount',
ascending=False)
df =
df.groupby('discount_price__amount').first().reset_index()
top5=df.head(5)
colors = ['purple', 'orange', 'blue', 'green', 'red']
# Membuat grafik
plt.bar(top5['title'],
top5['discount_price__amount'],color=colors)
plt.xlabel('sample Title')
plt.ylabel('Discount Price Amount')
plt.title('5 Lowest Discount Amount')
plt.xticks(rotation=90)
# Menampilkan grafik
plt.show()

# Mengurutkan DataFrame berdasarkan jumlah subscriber
secara menurun
df = df[df['discount_price__amount'] == 3200.0] # Filter
discount price amount yang bernilai 3200.0
df = df.sort_values('num_subscribers', ascending=False)
# Mengambil 5 data teratas berdasarkan jumlah subscriber
top_5 = df.head(5)
colors = ['red', 'green', 'blue', 'orange', 'purple']
# Membuat grafik
plt.bar(top_5['title'],
top_5['num_subscribers'],color=colors)
plt.xlabel('Title')
plt.ylabel('Number of Subscribers')
plt.title('Top 5 Titles by Number of Subscribers with
Discount Price Amount 3200.0')
plt.xticks(rotation=90)

```

```

# Menampilkan grafik
plt.show()

# Mengurutkan DataFrame berdasarkan jumlah subscriber
secara menurun
df = df[df['discount_price__amount'] == 455.0] # Filter
discount price amount yang bernilai 455.0
df = df.sort_values('num_subscribers', ascending=False)

# Mengambil 5 data teratas berdasarkan jumlah subscriber
top_5 = df.head(5)
colors = ['red', 'green', 'blue', 'orange', 'purple']
# Membuat grafik
plt.bar(top_5['title'],
top_5['num_subscribers'],color=colors)
plt.xlabel('Title')
plt.ylabel('Number of Subscribers')
plt.title('Top 5 Titles by Number of Subscribers with
Discount Price Amount 455.0')
plt.xticks(rotation=90)
# Menampilkan grafik
plt.show()

# Mengurutkan DataFrame berdasarkan jumlah subscriber
secara menurun
df = df[df['discount_price__amount'] == 3200.0] # Filter
discount price amount yang bernilai 3200.0
df = df.sort_values('num_reviews', ascending=False)
# Mengambil 5 data teratas berdasarkan jumlah subscriber
top_5 = df.head(5)
# Membuat grafik
plt.barh(top_5['title'],
top_5['num_reviews'],color='purple')
plt.xlabel('Number of Reviews')
plt.ylabel('Title')
plt.title('Top 5 Titles by Number of Reviews with Discount
Price Amount 3200.0')
plt.xticks(rotation=90)
plt.gca().invert_yaxis()
# Menampilkan grafik
plt.show()
# Meski begitu, ada sekitar 61 rows lagi yang memiliki
rating 5 pada discount amount price terendah yaitu 3200.0

# Mengurutkan DataFrame berdasarkan jumlah subscriber
secara menurun

```

```

df = df[df['discount_price__amount'] == 455.0] # Filter
discount price amount yang bernilai 455.0
df = df.sort_values('num_reviews', ascending=False)
top_5 = df.head(5)
plt.barh(top_5['title'],
top_5['num_reviews'],color='darkred')
plt.xlabel('Number Of Reviews')
plt.ylabel('Title')
plt.title('Top 5 Titles by Number Of Reviews with Discount
Price Amount 455.0')
plt.gca().invert_yaxis()
plt.show()

df = df.sort_values('price_detail__amount',
ascending=False)
df =
df.groupby('price_detail__amount').first().reset_index()
top5=df.tail(5)
top5 = top5.iloc[::-1]
colors = ['red', 'green', 'blue', 'orange', 'purple']
# Membuat grafik
plt.bar(top5['title'],
top5['price_detail__amount'],color=colors)
plt.xlabel('sample Title')
plt.ylabel('Price Amount')
plt.title('Top 5 Price Amount')
plt.xticks(rotation=90)
# Menampilkan grafik
plt.show()

df = df.sort_values('price_detail__amount',
ascending=False)
df =
df.groupby('price_detail__amount').first().reset_index()
top5=df.head(5)
colors = ['purple', 'orange', 'blue', 'green', 'red']
# Membuat grafik
plt.bar(top5['title'],
top5['price_detail__amount'],color=colors)
plt.xlabel('sample Title')
plt.ylabel('Price Amount')
plt.title('5 Lowest Price Amount')
plt.xticks(rotation=90)
# Menampilkan grafik
plt.show()

df = df.sort_values('num_subscribers', ascending=False)

```

```

top10 = df.head(10)
# Membuat grafik menggunakan paket matplotlib
plt.bar(top10['title'], top10['num_subscribers'])
plt.title('10 Subscribers Terbanyak')
plt.xlabel('Judul')
plt.ylabel('Total')
plt.xticks(rotation=90)
plt.show()

df = df.sort_values('num_published_lectures',
ascending=False)
top10 = df.head(10)
# Membuat grafik menggunakan paket matplotlib
plt.bar(top10['title'], top10['num_published_lectures'])
plt.title('10 lectures Terbanyak')
plt.xlabel('Judul')
plt.ylabel('Total')
plt.xticks(rotation=90)
plt.show()
df = df.sort_values('num_reviews', ascending=False)=
top10 = df.head(10)

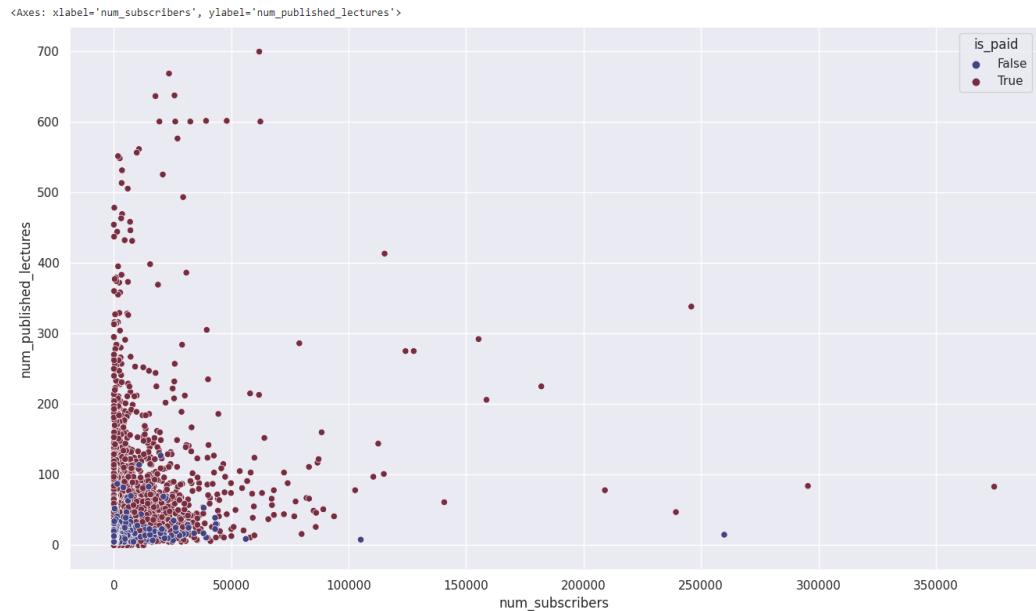
# Membuat grafik menggunakan paket matplotlib
plt.bar(top10['title'], top10['num_reviews'])
plt.title('10 reviews terbanyak')
plt.xlabel('Judul')
plt.ylabel('Total Review')
plt.xticks(rotation=90)
plt.show()

```

3.2 Hasil Eksekusi Diagram dan Penjelasan Program

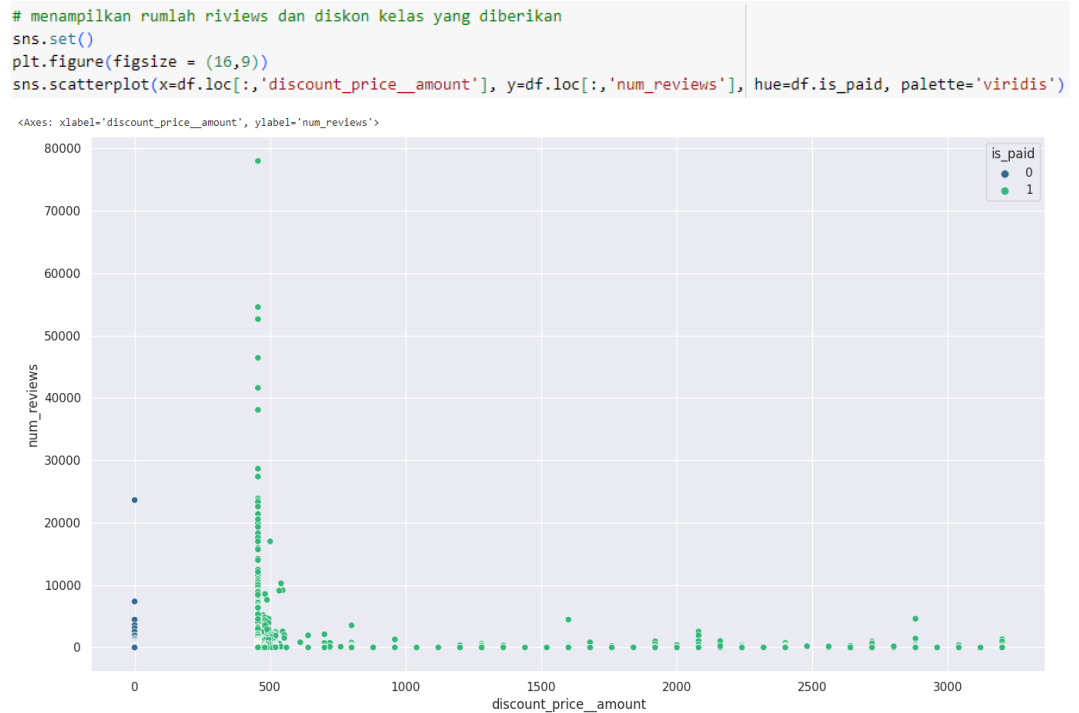
1. Grafik Scatterplot menunjukkan hubungan antara jumlah subscriber dengan jumlah kelas yang ditawarkan pada kursus

```
# Menunjukkan jumlah kelas yang ditawarkan kursus beserta jumlah subscriber nya
sns.set()
plt.figure(figsize = (16,9))
sns.scatterplot(x=df.loc[:, 'num_subscribers'], y=df.loc[:, 'num_published_lectures'], palette = 'icefire', hue=df.is_paid)
```



Grafik diatas merupakan grafik scatterplot yang menunjukan hubungan antara jumlah subscriber dengan jumlah kelas yang ada pada suatu kursus. Pada grafik ini dapat dilihat bahwa, banyak nya kelas yang di sediakan suatu kursus tidak mempengaruhi jumlah subscriber suatu kursus tersebut. Ada kursus yang memiliki banyak kelas yang disediakan tetapi subscriber nya sedikit, dan sebaliknya ada kursus yang menyediakan kelas sedikit tetapi memiliki banyak subscriber.

2. Grafik Scatterplot menunjukkan hubungan antara jumlah reviews kursus dengan harga diskon yang ditawarkan

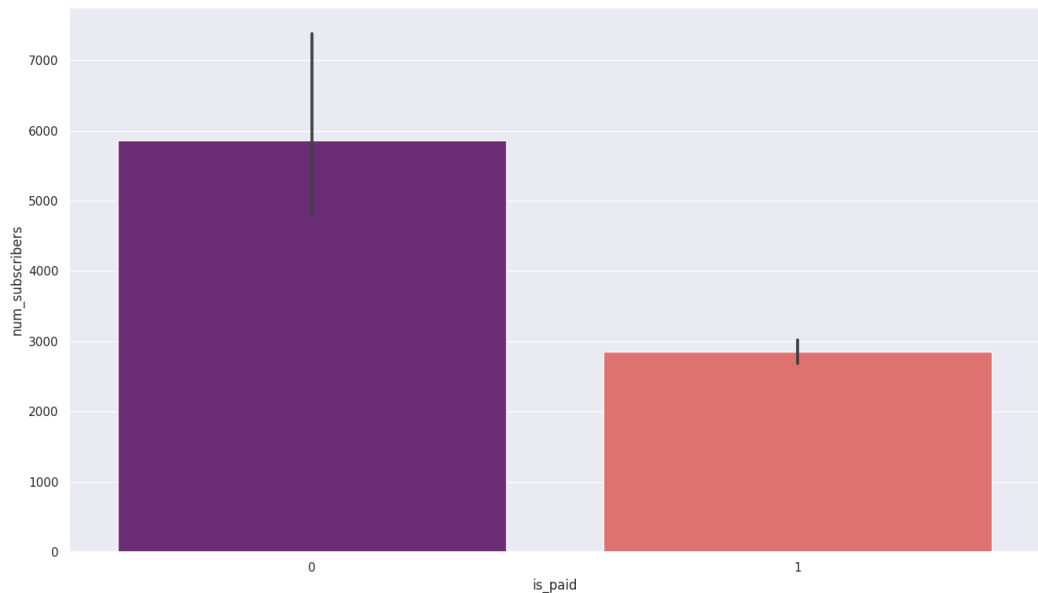


Grafik diatas merupakan grafik scatterplot yang menunjukan hubungan antara jumlah reviews kursus dengan harga diskon suatu kelas yang ada pada suatu kursus. Pada grafik ini dapat dilihat bahwa, harga diskon yang diberikan tidak mempengaruhi jumlah reviews yang di berikan oleh subscriber. Semakin besar harga diskon yang diberikan belum tentu memiliki banyak reviews dari subscribarnya.

3. Grafik diagram batang menunjukkan perbandingan banyak kursus berbayar dengan tidak berbayar (free) berdasarkan jumlah subscriber

```
# menampilkan data course yang dibayar (true) dan tidak dibayar (false), berdasarkan jumlah subscribernya
sns.set()
plt.figure(figsize = (16,9))
sns.barplot(x=df.loc[:, 'is_paid'], y=df.loc[:, 'num_subscribers'], palette = 'magma')
```

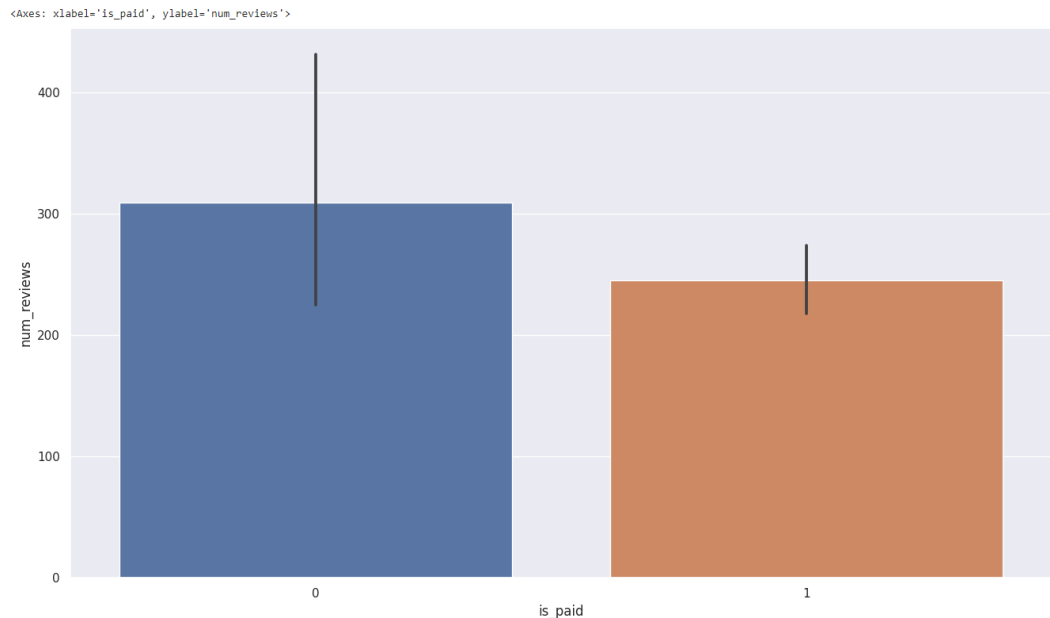
<Axes: xlabel='is_paid', ylabel='num_subscribers'>



Grafik diatas merupakan diagram batang yang menunjukan perbandingan banyak kursus berbayar dengan tidak berbayar (free) yang mempengaruhi hubungan dengan banyak subscriber yang dimiliki suatu kursus. Pada diagram ini dapat dilihat bahwa subscriber terbanyak ada pada kursus yang tidak berbayar (free). Dapat disimpulkan bahwa, kebanyakan subscriber dari UdeMy mengikuti kursus yang tidak berbayar dibanding kan dengan kursus yang berbayar

4. Grafik diagram batang menunjukkan perbandingan banyak kursus berbayar dengan tidak berbayar (free) berdasarkan jumlah review

```
# menampilkan data course yang dibayar (true) dan tidak dibayar (false), berdasarkan jumlah review
sns.set()
plt.figure(figsize = (16,9))
sns.barplot(x=df.loc[:, 'is_paid'], y=df.loc[:, 'num_reviews'])
```

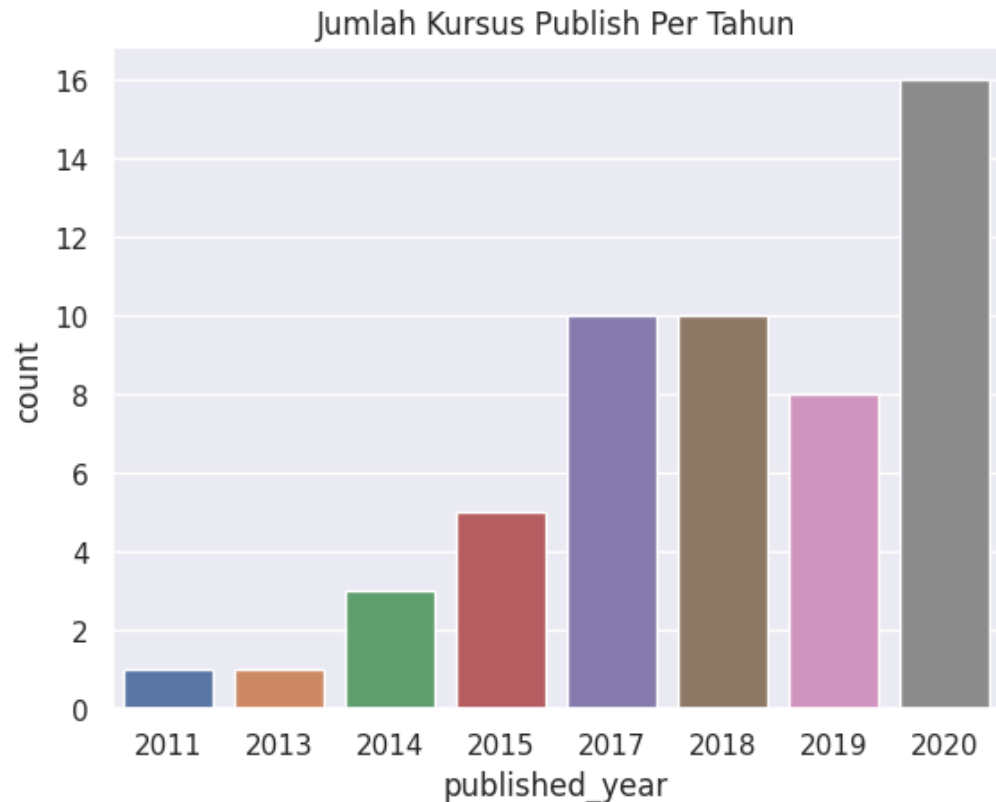


Grafik diatas merupakan diagram batang yang menunjukan perbandingan banyak kursus berbayar dengan tidak berbayar (free) yang mempengaruhi hubungan dengan banyak reviews kursus. Pada diagram ini dapat dilihat bahwa jumlah reviews kursus yang tidak berbayar lebih banyak dari kursus berbayar. Hal ini juga berkaitan dengan grafik pada pembahasan sebelumnya. Dimana jumlah subscriber dari kursus tidak berbayar lebih banyak maka, jumlah reviews dipengaruhi oleh jumlah subscribernya. Dapat disimpulkan bahwa kursus yang berbayar dan tidak berbayar mempengaruhi jumlah reviews.

5. Grafik diagram batang menunjukkan jumlah kursus yang dipublish di setiap tahunnya.

```
# menampilkan jumlah course yang di publish di setiap tahunnya  
sns.countplot(x= "published_year", data= df).set_title('Jumlah Kursus Publish Per Tahun')
```

```
Text(0.5, 1.0, 'Jumlah Kursus Publish Per Tahun')
```

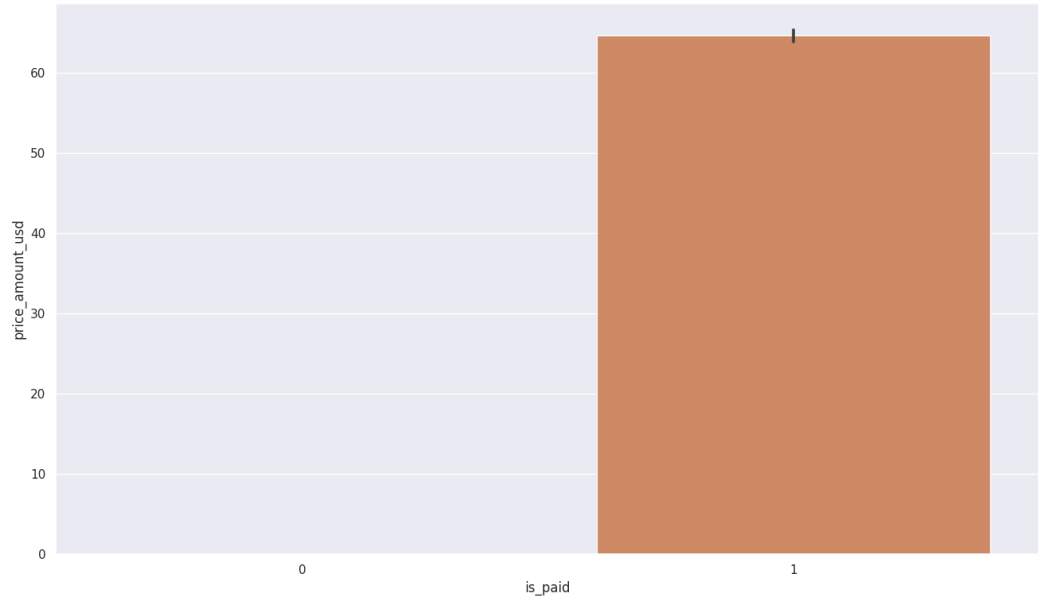


Grafik diatas merupakan diagram batang yang menunjukan jumlah kursus yang dipublish di setiap tahunnya. Pada diagram ini dapat dilihat bahwa jumlah kursus terbanyak dipublish pada tahun 2020. Dataset yang digunakan merupakan sekumpulan data yang didokumentasikan sejak tahun 2011. Oleh karena itu, grafik diatas menunjukkan jumlah kursus pada UdeMy mulai tahun 2011 sampai tahun 2020.

6. Grafik diagram batang menunjukkan perbandingan banyak kursus berbayar dengan tidak membayar (free) berdasarkan mata uang dollar

```
# menampilkan course yang dibayar(true) atau tidak dibayar(false) dengan tidak menggunakan mata uang dollar
sns.set()
plt.figure(figsize = (16,9))
sns.barplot(x=df.loc[:, 'is_paid'], y=df.loc[:, 'price_amount_usd'])
```

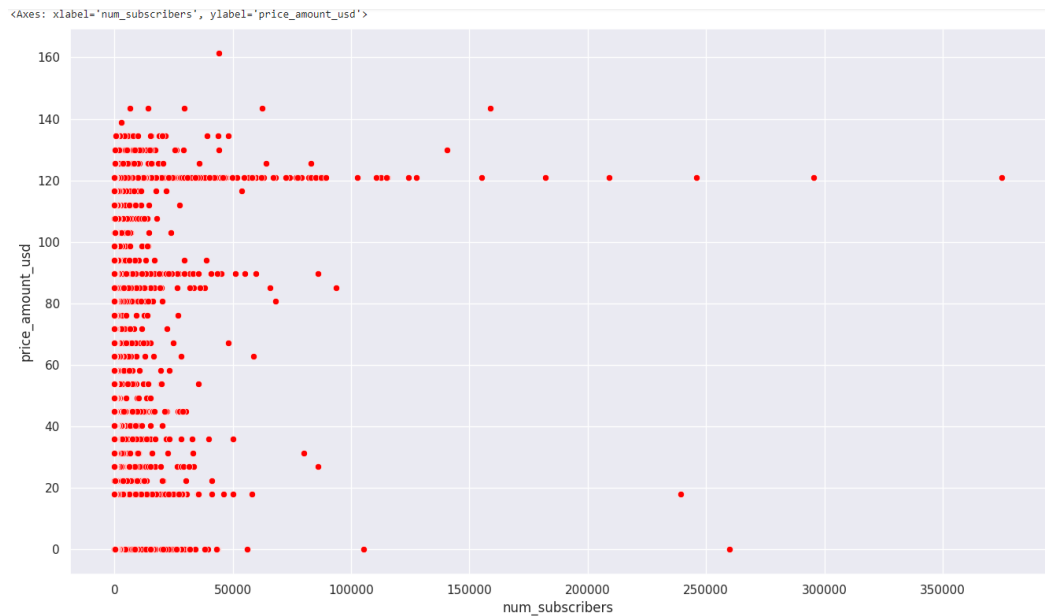
<Axes: xlabel='is_paid', ylabel='price_amount_usd'>



Grafik diatas merupakan diagram batang yang menunjukan perbandingan banyak kursus berbayar dengan tidak membayar (free) berdasarkan mata uang dollar. Pada grafik dapat dilihat bahwa pada data ini semua kursus yang berbayar tidak menggunakan ada yang menggunakan mata uang dollar.

7. Grafik Scatterplot menunjukkan hubungan antara jumlah suscriber dengan harga kursus yang ditawarkan

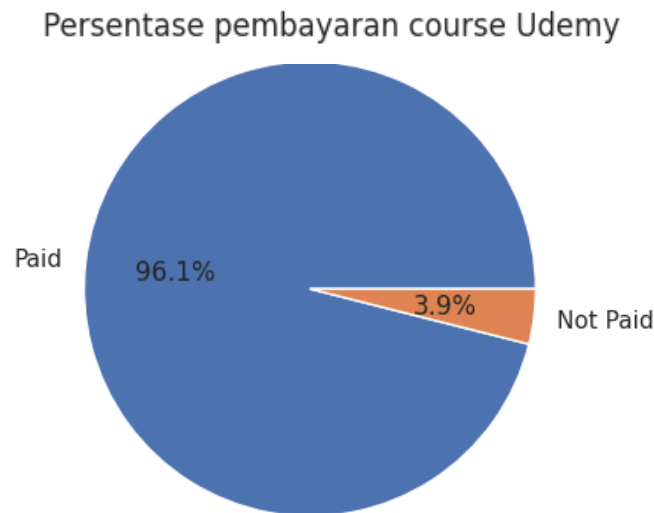
```
# menampilkan jumlah pelanggan berdasarkan pengaruh harga course
sns.set()
plt.figure(figsize = (16,9))
sns.scatterplot(x=df.loc[:, 'num_subscribers'], y=df.loc[:, 'price_amount_usd'], color = 'red')
```



Grafik diatas merupakan grafik scatterplot yang menunjukan hubungan antara jumlah suscriber dengan harga kursus yang ditawarkan. Pada grafik ini dapat dilihat bahwa, harga kursus yang diberikan tidak dapat mempengaruhi jumlah subscriber pada suatu kursus. Semakin mahal harga yang ditawarkan suatu kursus belum tentu memiliki jumlah subscriber yang sedikit, hal ini dapat dibuktikan berdasarkan grafik scatterplot diatas.

8. Grafik Pie menunjukkan persentase perbandingan kursus berbayar dan tidak berbayar

```
paid_course_data = [len(df[df['is_paid']==True]),  
                    len(df[df['is_paid']==False])]  
paid_course_labels = ['Paid', 'Not Paid']  
plt.title('Persentase pembayaran course Udemy')  
plt.pie(paid_course_data, labels = paid_course_labels, autopct='%0.1f%%')  
plt.show()
```



Grafik diatas merupakan grafik pie yang menunjukan persentase perbandingan kursus berbayar dan tidak berbayar. Pada grafik ini dapat dilihat bahwa, secara keseluruhan course Udemy menyediakan lebih banyak kursus yang berbayar dibandingkan dengan kursus yang gratis. Dengan persentase perbandingan banyak kursus berbayar : kursus tidak berbayar yaitu 96.1% : 3.9%

9. Grafik Pie menunjukkan persentase perbandingan jenis mata uang yang berlaku pada harga diskon kursus

```
mataUang = [np.count_nonzero(df['price_detail_currency']=='INR'),  
            np.count_nonzero(df['price_detail_currency']!='INR')]  
label = ['INR', 'non INR']  
plt.title('Mata uang yang berlaku dengan harga Diskon')  
plt.pie(mataUang, labels = label, radius=1.3, startangle=60, shadow=True, autopct='%0.1f%%')  
plt.show()
```



Grafik diatas merupakan grafik pie yang menunjukan jenis mata uang yang berlaku pada harga diskon kursus. Pada grafik ini dapat dilihat bahwa, secara keseluruhan course Udemmy hanya menyediakan mata uang INR (Ruppee India) yang berlaku pada harga diskon suatu kursus. Oleh karena itu, pada grafik menunjukkan 100% harga diskon suatu kursus menggunakan mata uang INR (Ruppee India)

10. Diagram Top 10 Sumber Pendapatan Terbesar

```
# top10 title dengan sumber pendapatan terbanyak  
sort = df.sort_values(by=["total"], ascending= False)  
%matplotlib inline  
  
# ambil sampel df 10 baris  
sort_10= sort.head(10)  
  
# sort_10.plot.barh(x='title', y='total',  
#                  title='Cuisine Preference', color='green')
```



```
fig = plt.figure(figsize=(11,3))

ax = fig.add_axes([0,0,1,1])
ax.set_xlabel('title')
ax.set_ylabel('total_pendapatan (miliar)')

ax.bar(sort_10['title'],sort_10['total'])
plt.xticks(rotation=90)
plt.title('Top10 Sumber Pendapatan Terbesar')
plt.show()
```

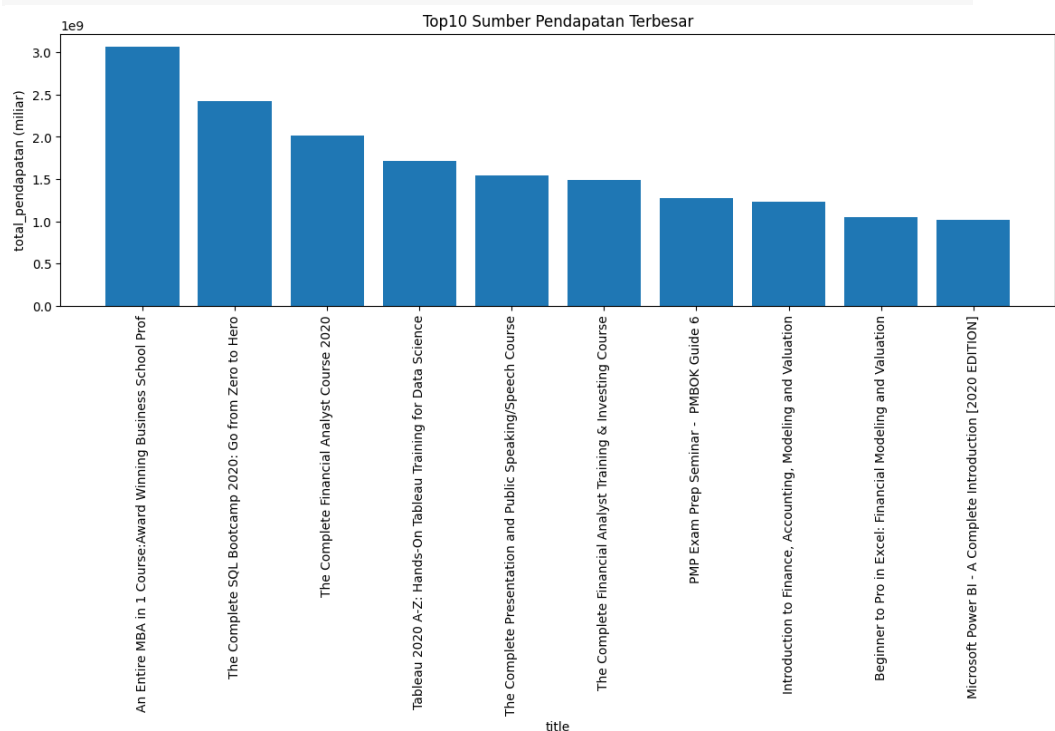


Diagram bar diatas menjelaskan top 10 kursus yang memiliki pendapatan terbanyak. Pada diagram diatas bisa terlihat bahwa kursus yang memiliki pendapatan terbanyak adalah kursus “An Entire MBA in 1 Course : Award Winning Business School”.

11. Diagram top 5 jumlah diskon terbesar

```
df = df.sort_values('discount_price_amount', ascending=False)
df = df.groupby('discount_price_amount').first().reset_index()
top5=df.tail(5)
top5 = top5.iloc[::-1]
colors = ['red', 'green', 'blue', 'orange', 'purple']

# Membuat grafik
plt.bar(top5['title'], top5['discount_price_amount'],color=colors)
plt.xlabel('sample title')
plt.ylabel('Discount Price Amount')
plt.title('Top 5 Discount Amount')
plt.xticks(rotation=90)

# Menampilkan grafik
plt.show()
```

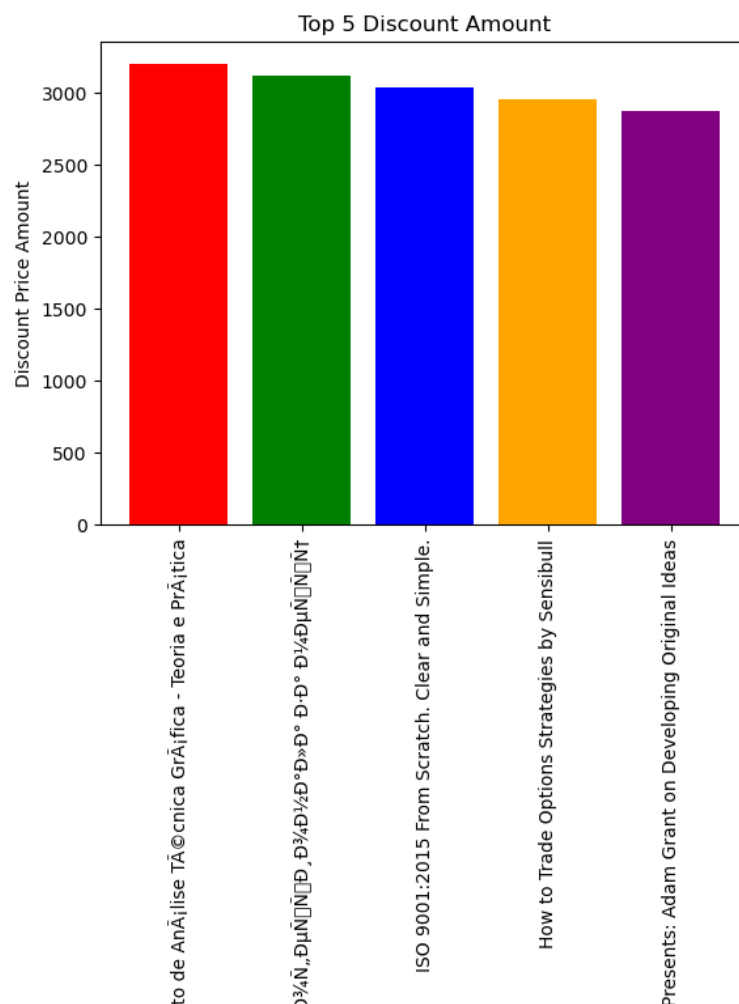


Diagram tersebut menampilkan 5 jumlah diskon tertinggi yang ada di udemy ini, diantaranya ada diskon sebanyak 3200, 3120, 3040, dan seterusnya (dalam mata uang india/INR).

12. 5 jumlah diskon terendah yang diberikan

```
df = df.sort_values('discount_price__amount', ascending=False)
df = df.groupby('discount_price__amount').first().reset_index()
top5=df.head(5)
colors = ['purple', 'orange', 'blue', 'green', 'red']

# Membuat grafik
plt.bar(top5['title'], top5['discount_price__amount'],color=colors)
plt.xlabel('sample Title')
plt.ylabel('Discount Price Amount')
plt.title('5 Lowest Discount Amount')
plt.xticks(rotation=90)

# Menampilkan grafik
plt.show()
```

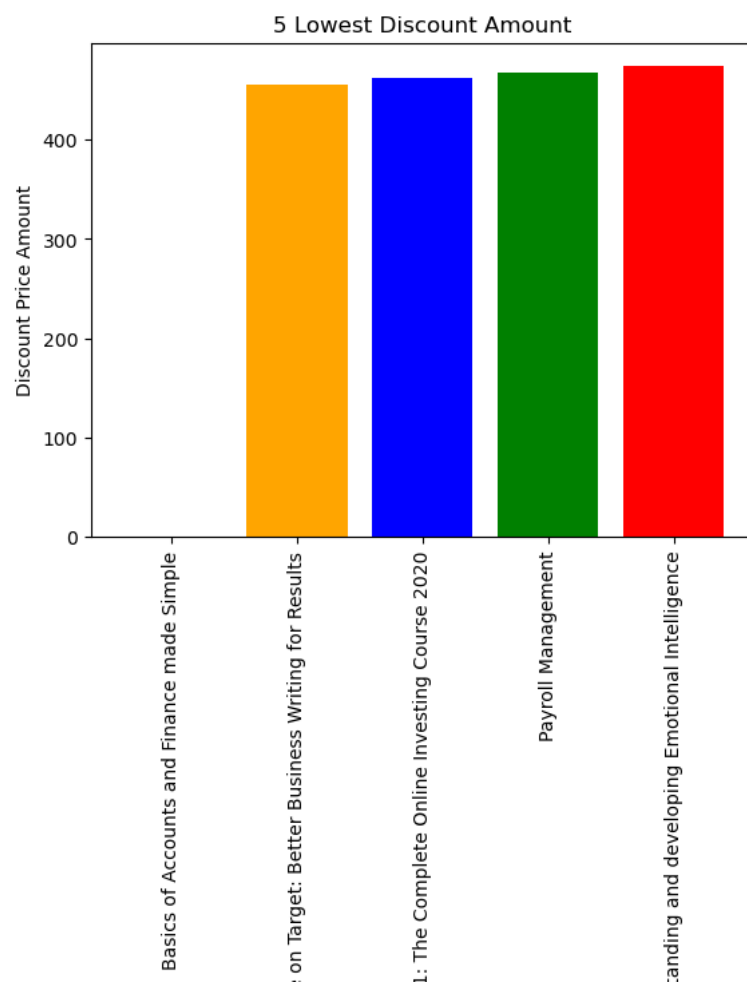


Diagram tersebut menampilkan 5 jumlah diskon terendah yang ada di udemy ini, diantaranya ada diskon sebanyak 474, 468, dan seterusnya. (dalam mata uang india/INR).

13. Top 5 kursus dengan jumlah subscriber terbanyak, khusus jumlah diskon 3200

```
# Mengurutkan DataFrame berdasarkan jumlah subscriber secara menurun
df = df[df['discount_price_amount'] == 3200.0] # Filter discount price amount yang bernilai 3200.0
df = df.sort_values('num_subscribers', ascending=False)

# Mengambil 5 data teratas berdasarkan jumlah subscriber
top_5 = df.head(5)

colors = ['red', 'green', 'blue', 'orange', 'purple']
# Membuat grafik
plt.bar(top_5['title'], top_5['num_subscribers'], color=colors)
plt.xlabel('Title')
plt.ylabel('Number of Subscribers')
plt.title('Top 5 Titles by Number of Subscribers with Discount Price Amount 3200.0')
plt.xticks(rotation=90)

# Menampilkan grafik
plt.show()
```

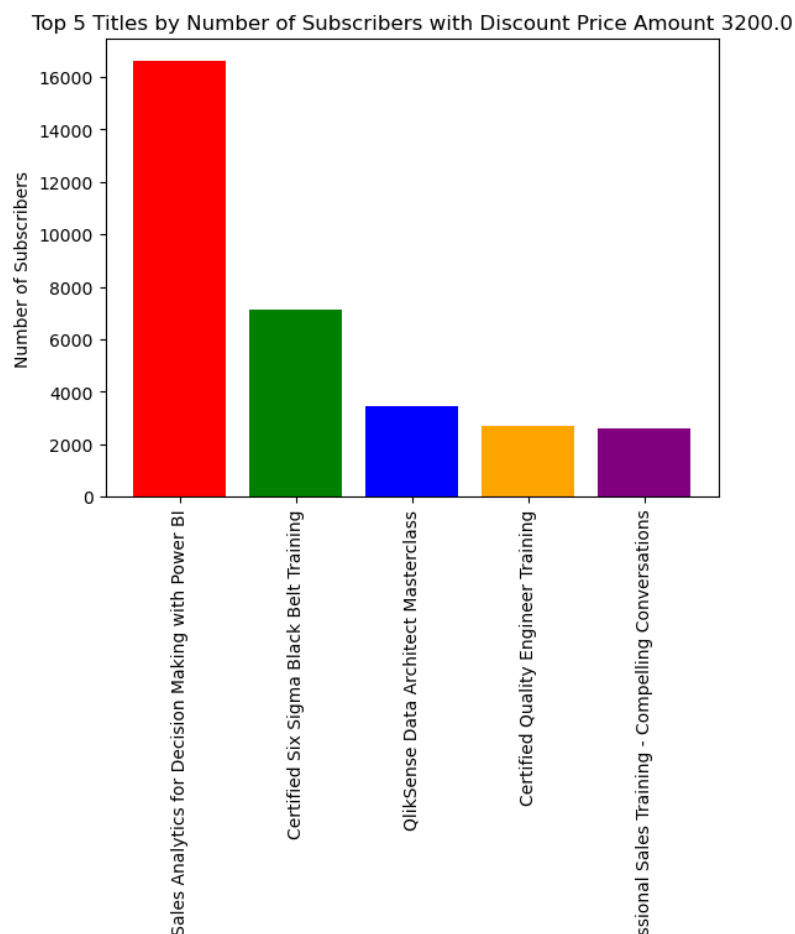


Diagram tersebut menampilkan top 5 jumlah kursus yang ada di udemy berdasarkan jumlah subscriber terbanyak, namun terkhusus untuk jumlah diskon yang diberikan sebanyak 3200 (dalam mata uang india/INR)

14. Top 5 kursus dengan jumlah subscriber terbanyak, khusus jumlah diskon 455

```
# Mengurutkan DataFrame berdasarkan jumlah subscriber secara menurun
df = df[df['discount_price_amount'] == 455.0] # Filter discount price amount yang bernilai 455.0
df = df.sort_values('num_subscribers', ascending=False)

# Mengambil 5 data teratas berdasarkan jumlah subscriber
top_5 = df.head(5)
colors = ['red', 'green', 'blue', 'orange', 'purple']

# Membuat grafik
plt.bar(top_5['title'], top_5['num_subscribers'], color=colors)
plt.xlabel('Title')
plt.ylabel('Number of Subscribers')
plt.title('Top 5 Titles by Number of Subscribers with Discount Price Amount 455.0')
plt.xticks(rotation=90)

# Menampilkan grafik
plt.show()
```

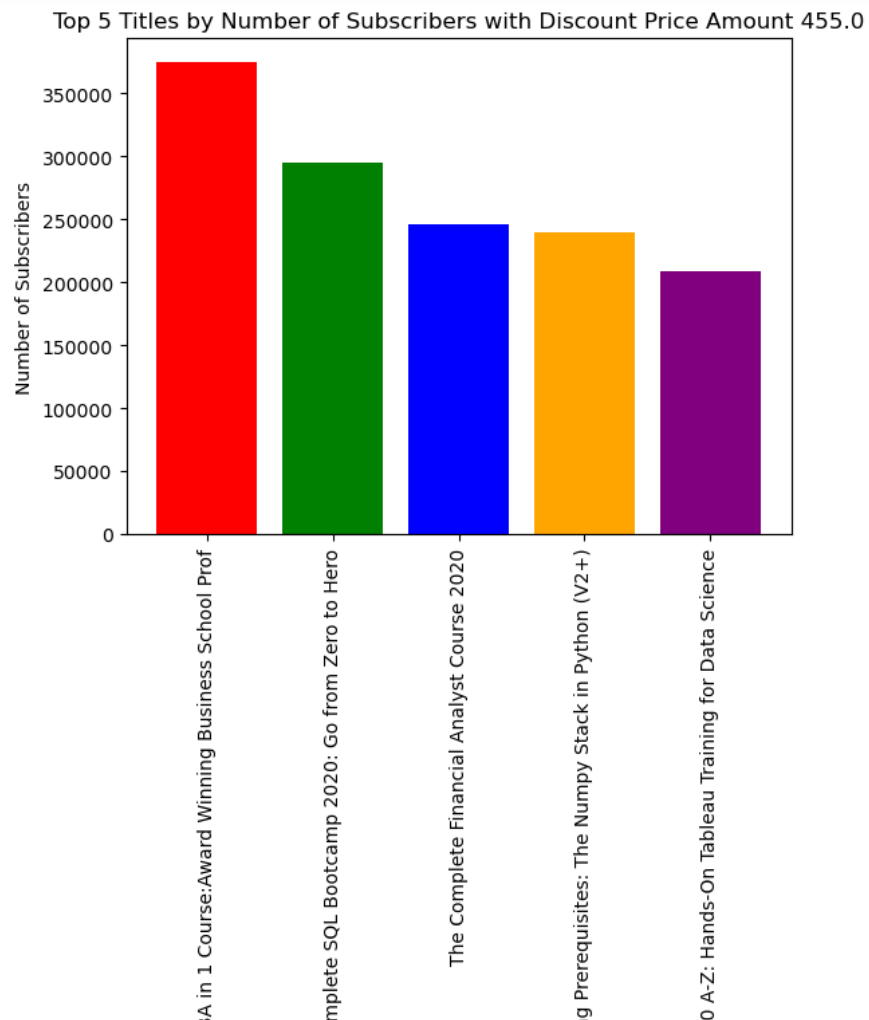


Diagram tersebut menampilkan top 5 jumlah kursus yang ada di udemy berdasarkan jumlah subscriber terbanyak, namun terkhusus untuk jumlah diskon yang diberikan sebanyak 455 (dalam mata uang india/INR)

15. Top 5 kursus dengan jumlah reviews terbanyak, khusus untuk jumlah diskon 3200

```
# Mengurutkan DataFrame berdasarkan jumlah subscriber secara menurun
df = df[df['discount_price_amount'] == 3200.0] # Filter discount price amount yang bernilai 3200.0
df = df.sort_values('num_reviews', ascending=False)

# Mengambil 5 data teratas berdasarkan jumlah subscriber
top_5 = df.head(5)

# Membuat grafik
plt.barh(top_5['title'], top_5['num_reviews'], color='purple')
plt.xlabel('Number of Reviews')
plt.ylabel('Title')
plt.title('Top 5 Titles by Number of Reviews with Discount Price Amount 3200.0')
plt.xticks(rotation=90)
plt.gca().invert_yaxis()

# Menampilkan grafik
plt.show()

# Meski begitu, ada sekitar 61 rows lagi yang memiliki rating 5 pada discount amount price terendah yaitu 3200.0
```

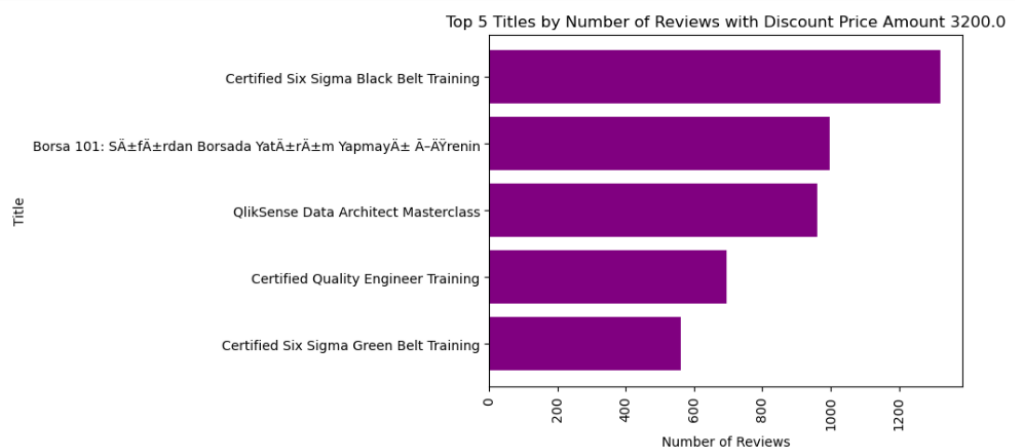


Diagram tersebut menampilkan top 5 jumlah kursus yang ada di udemy berdasarkan jumlah reviews terbanyak, namun terkhusus untuk jumlah diskon yang diberikan sebanyak 3200 (dalam mata uang india/INR)

16. Top 5 kursus dengan jumlah reviews terbanyak, khusus untuk jumlah diskon 455

```
# Mengurutkan DataFrame berdasarkan jumlah subscriber secara menurun
df = df[df['discount_price_amount'] == 455.0] # Filter discount price amount yang bernilai 455.0
df = df.sort_values('num_reviews', ascending=False)
top_5 = df.head(5)

plt.barh(top_5['title'], top_5['num_reviews'], color='darkred')
plt.xlabel('Number Of Reviews')
plt.ylabel('Title')
plt.title('Top 5 Titles by Number Of Reviews with Discount Price Amount 455.0')
plt.gca().invert_yaxis()

plt.show()
```

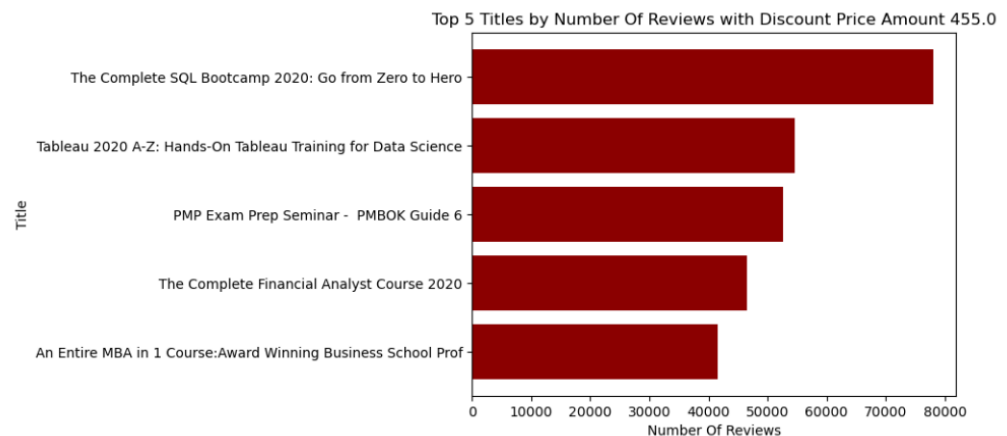


Diagram tersebut menampilkan top 5 jumlah kursus yang ada di udemy berdasarkan jumlah reviews terbanyak, namun terkhusus untuk jumlah diskon yang diberikan sebanyak 455 (dalam mata uang india/INR)

17. Top 5 harga tertinggi yang tersedia

```
df = df.sort_values('price_detail__amount', ascending=False)
df = df.groupby('price_detail__amount').first().reset_index()
top5=df.tail(5)
top5 = top5.iloc[::-1]
colors = ['red', 'green', 'blue', 'orange', 'purple']

# Membuat grafik
plt.bar(top5['title'], top5['price_detail__amount'],color=colors)
plt.xlabel('sample Title')
plt.ylabel('Price Amount')
plt.title('Top 5 Price Amount')
plt.xticks(rotation=90)

# Menampilkan grafik
plt.show()
```

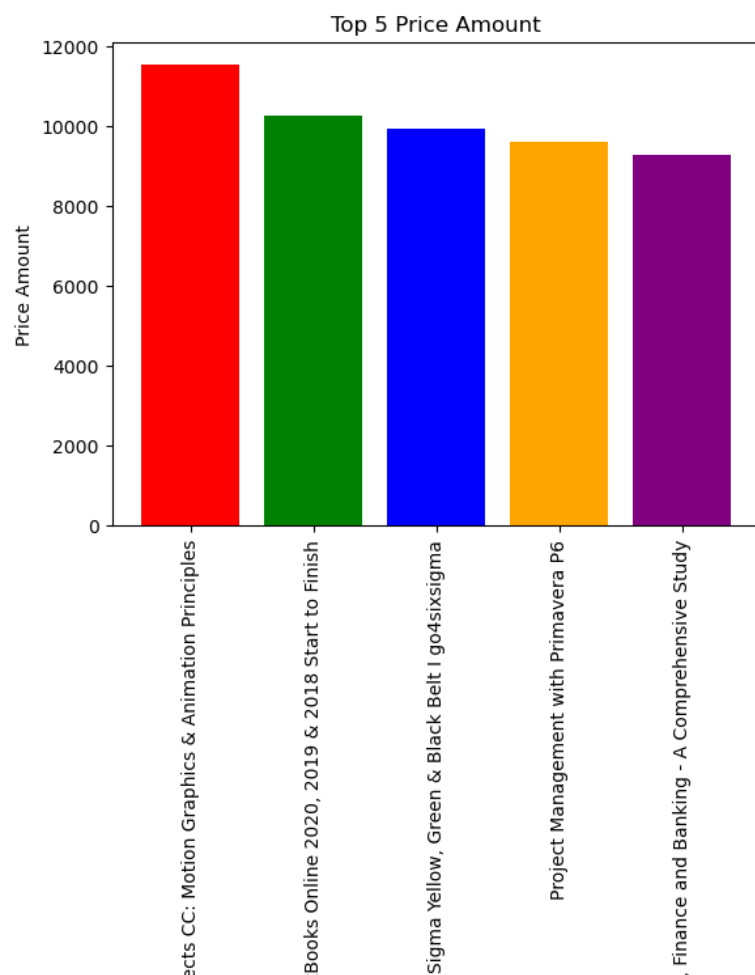


Diagram tersebut menampilkan top 5 harga tertinggi yang tersedia di udemy ini dari berbagai kursus yang ada, diantaranya ada harga sebanyak 11520,10240,9920, dan seterusnya. (dalam mata uang india/INR).

18. 5 harga terendah yang tersedia

```
df = df.sort_values('price_detail__amount', ascending=False)
df = df.groupby('price_detail__amount').first().reset_index()
top5=df.head(5)
colors = ['purple', 'orange', 'blue', 'green', 'red']

# Membuat grafik
plt.bar(top5['title'], top5['price_detail__amount'],color=colors)
plt.xlabel('sample Title')
plt.ylabel('Price Amount')
plt.title('5 Lowest Price Amount')
plt.xticks(rotation=90)

# Menampilkan grafik
plt.show()
```

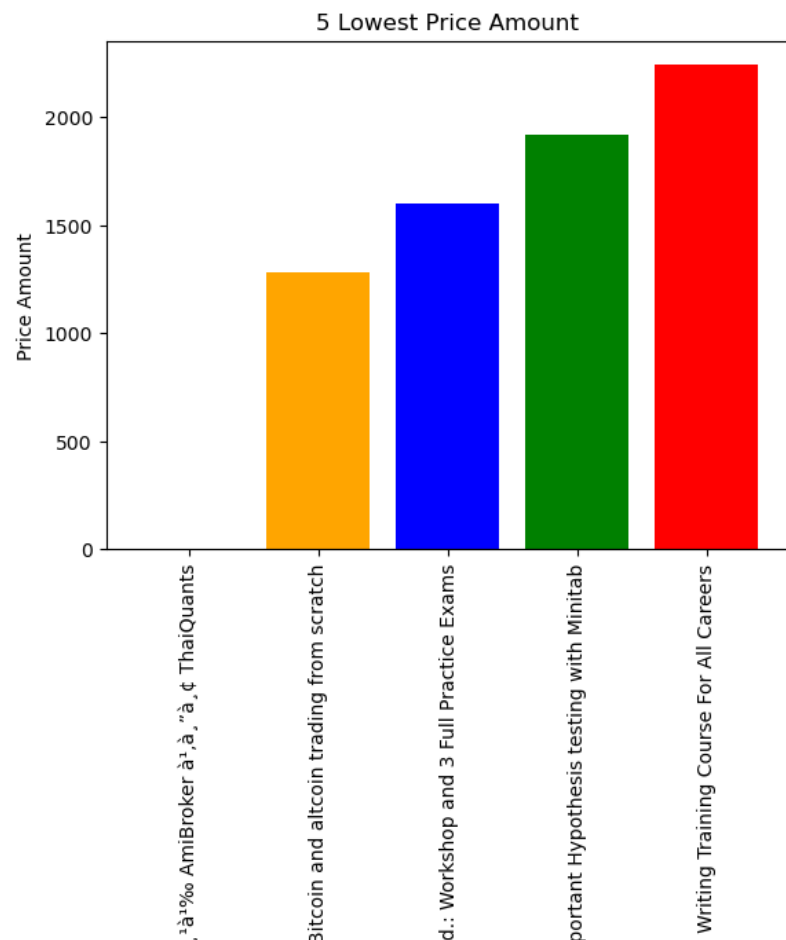


Diagram tersebut menampilkan 5 harga terendah yang tersedia di udemy ini dari berbagai kursus yang ada, diantaranya ada harga sebanyak 2240,1920,1600, dan seterusnya.(dalam mata uang india/INR).

19. Diagram top 10 pelanggan terbanyak

```
# Membuat grafik menggunakan paket matplotlib
plt.bar(top10['title'], top10['num_subscribers'])
plt.title('10 Subscribers Terbanyak')
plt.xlabel('Judul')
plt.ylabel('Total')
plt.xticks(rotation=90)
plt.show()
```

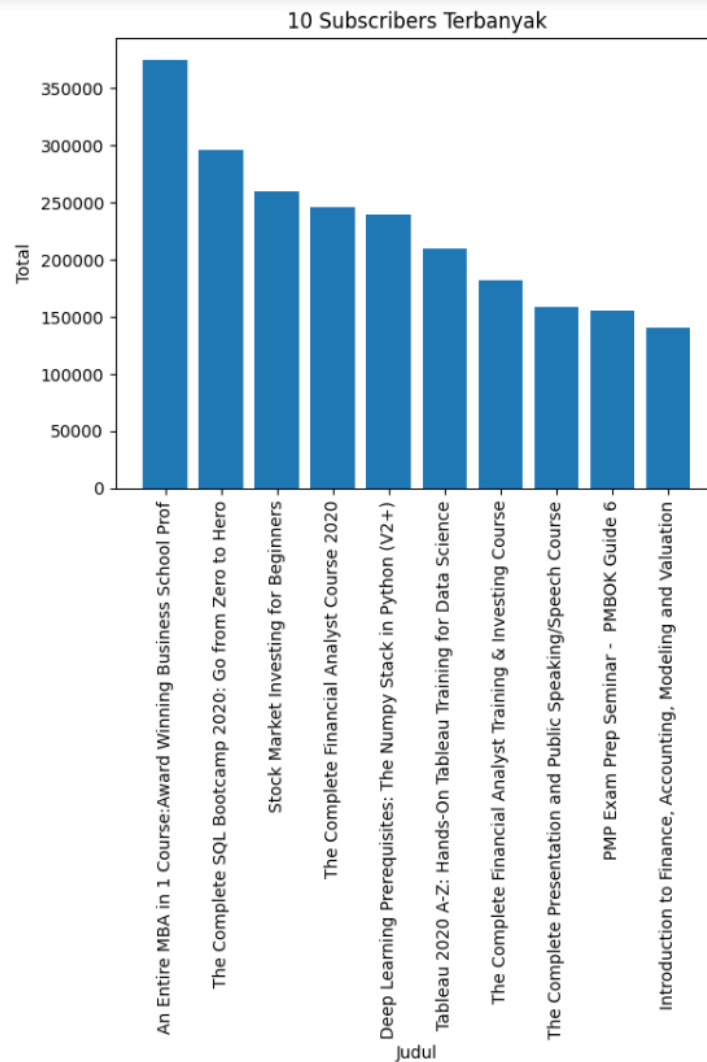


Diagram bar diatas menjelaskan top 10 kursus yang memiliki subscriber terbanyak. Terlihat pada grafik di atas bahwa course yang memiliki subscriber terbanyak yaitu “An Entire MBA in 1 Course : Award Winning Business School”.

20. Diagram 10 lectures terbanyak

```
# Membuat grafik menggunakan paket matplotlib
plt.bar(top10['title'], top10['num_published_lectures'])
plt.title('10 lectures Terbanyak')
plt.xlabel('Judul')
plt.ylabel('Total')
plt.xticks(rotation=90)
plt.show()
```

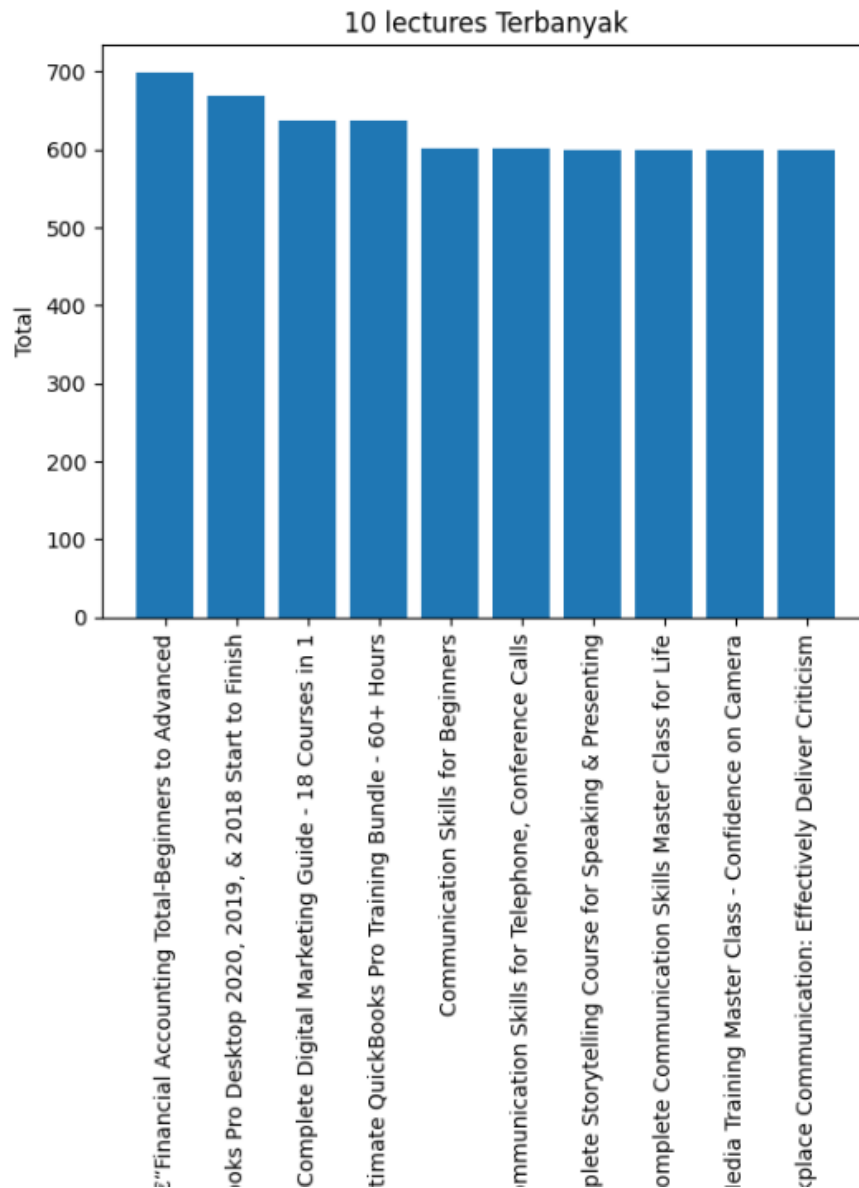


Diagram bar diatas menjelaskan top 10 kursus yang memiliki lectures terbanyak.

21. Diagram 10 reviews terbanyak

```
# Membuat grafik menggunakan paket matplotlib
plt.bar(top10['title'], top10['num_reviews'])
plt.title('10 reviews terbanyak')
plt.xlabel('Judul')
plt.ylabel('Total Review')
plt.xticks(rotation=90)
plt.show()
```

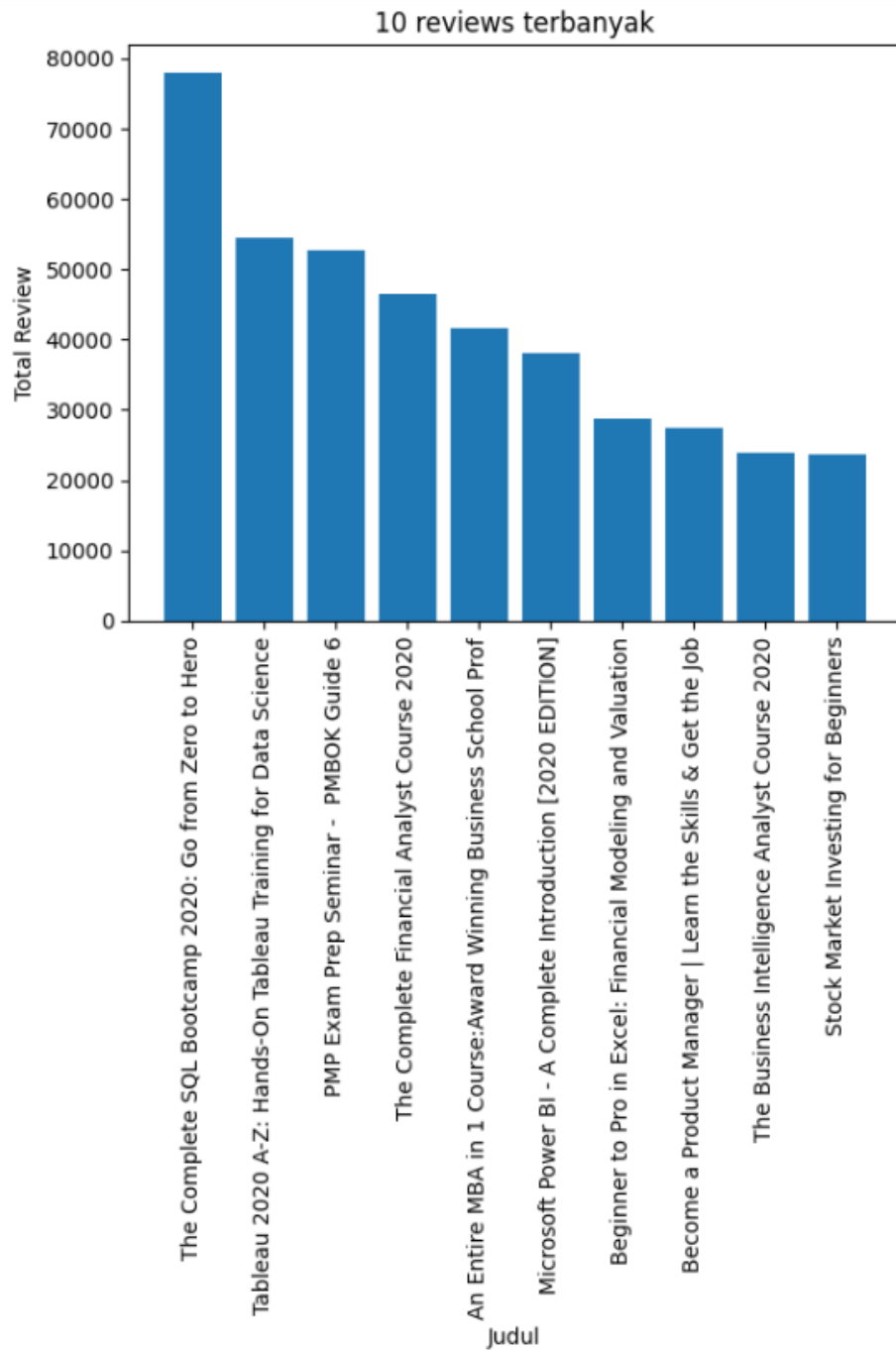
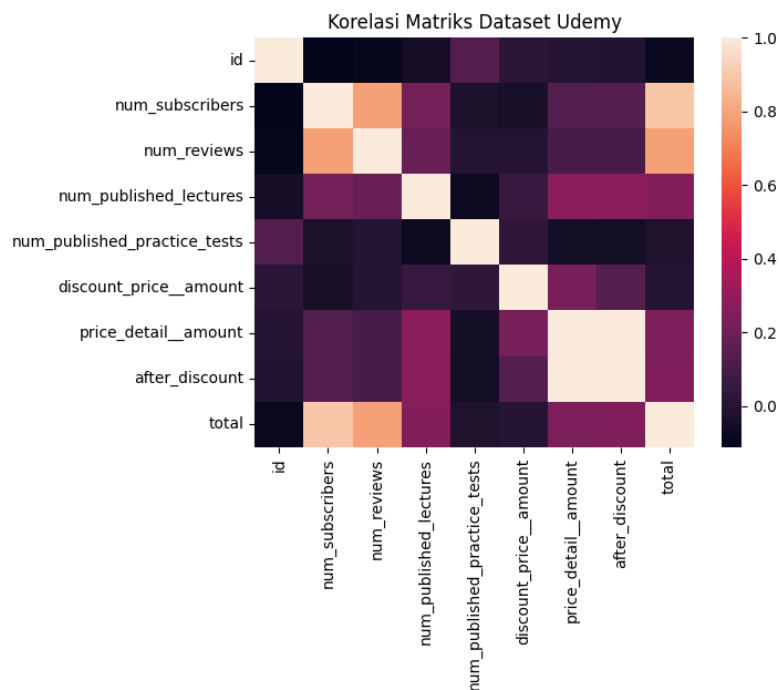


Diagram bar diatas menjelaskan top 10 kursus yang memiliki reviews terbanyak. Terlihat pada grafik di atas bahwa course “The Complete SQL Bootcamp 2020: Go from Zero to Hero” yang paling banyak di review

22. Grafik Heatmap untuk menentukan korelasi antar entitas

```
# menampilkan korelasi antar entitas menggunakan heatmap
plt.title('Korelasi Matriks Dataset UdeMy')
sns.heatmap(data=df.corr())
```

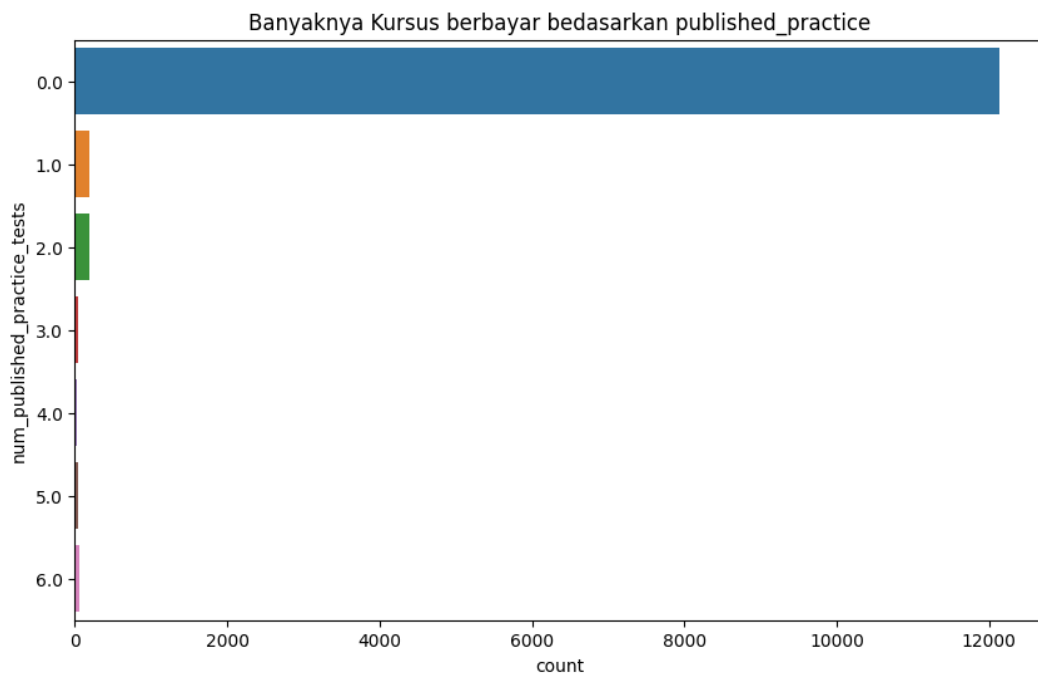
Hasil Visualisasi :



Grafik diatas menunjukkan hubungan korelasi yang dimiliki antar matriks, dapat dilihat bahwa semakin cerah warna pada grafik maka semakin erat hubungannya (dengan angka tertinggi). Dari hasil grafik dapat didapati bahwa price_detail_amount erat kaitannya dengan after_discount, kemudian diikuti dengan antara num_subscribers dengan total, num_reviews dengan total. Begitupun sebaliknya semakin gelap warna yg didapat maka hubungan semakin kecil seperti halnya hubungan id dengan total.

23. Grafik Bar Banyaknya Kursus Berbayar Berdasarkan Published_Practice

```
plt.figure(figsize=(10,6))
plt.title("Banyaknya Kursus berbayar berdasarkan published_practice")
plt.xlabel("rating")
plt.ylabel("published_practice")
# data.iloc yang akan ditampilkan
sns.countplot(y=df.iloc[4:,11])
plt.show()
```



Grafik diatas menunjukkan diagram batang banyaknya kursus berbayar berdasarkan published_practice berdasarkan rating. Didapati bahwa nilai dari published_practice terdiri dari 0,1,2,3,4,5 dan 6 serta pada sumbu x dijelaskan banyaknya rating. Hasil grafik menunjukkan num_publish_practice 0 memiliki rating 12000 (paling tinggi).

24. Grafik Scatterplot Hubungan rating dengan num_reviews pengelompokkan berdasarkan is_paid (kursus berbayar atau tidak)

```
# diagram scatterplot
plt.figure(figsize=(5,5),dpi=200)
sns.scatterplot(data=df, x='rating', y='num_reviews', hue='is_paid', palette='Set1')
plt.title('Hubungan Rating dengan num_reviews berdasarkan tipe kursus')
```

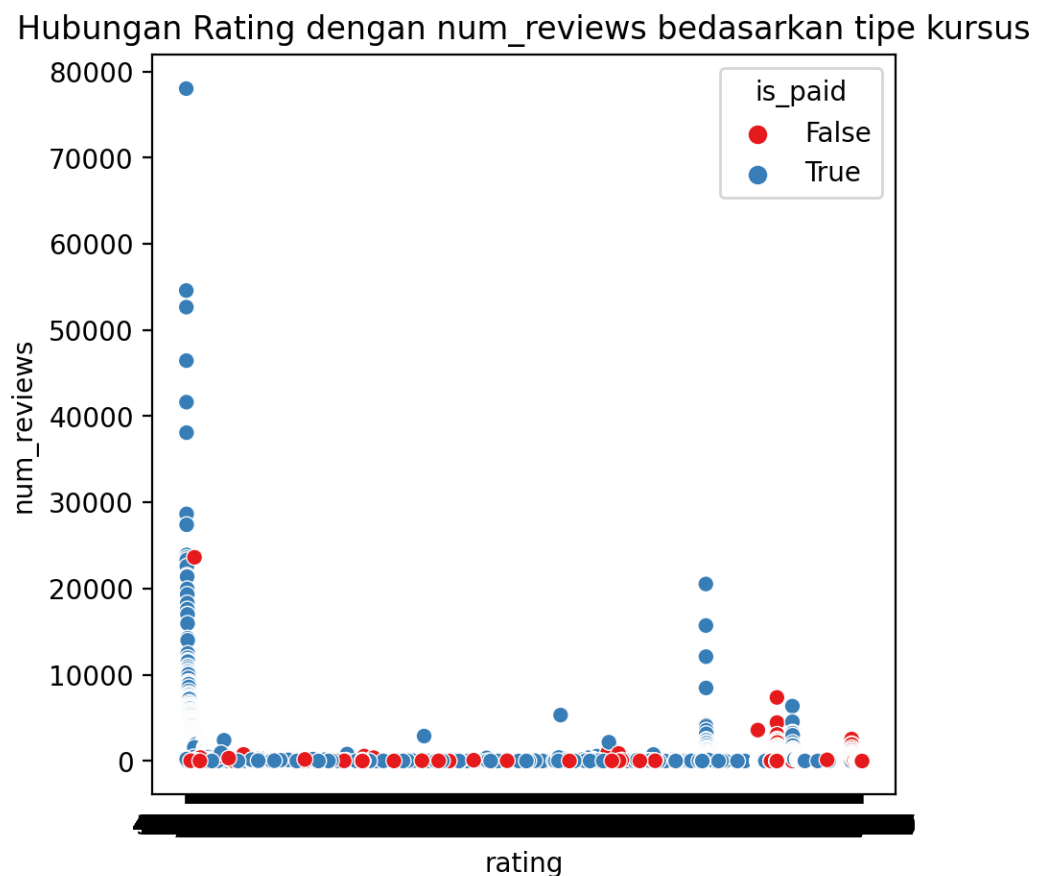
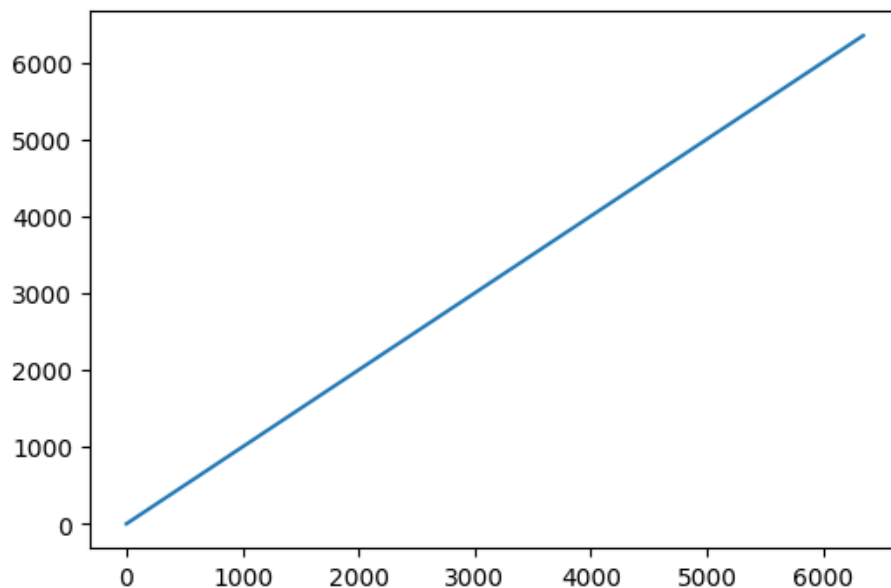


Diagram scatterplot diatas merupakan sebaran dari data rating berdasarkan kategori is_paid (kursus berbayar atau tidak). Sehingga pada diagram lebih banyak ditampilkan dengan kursus berbayar. Kemudian, pada rentang rating rendah memiliki angka review yang cukup tinggi.

25. Grafik Line menampilkan banyaknya rating dengan kursus berbayar

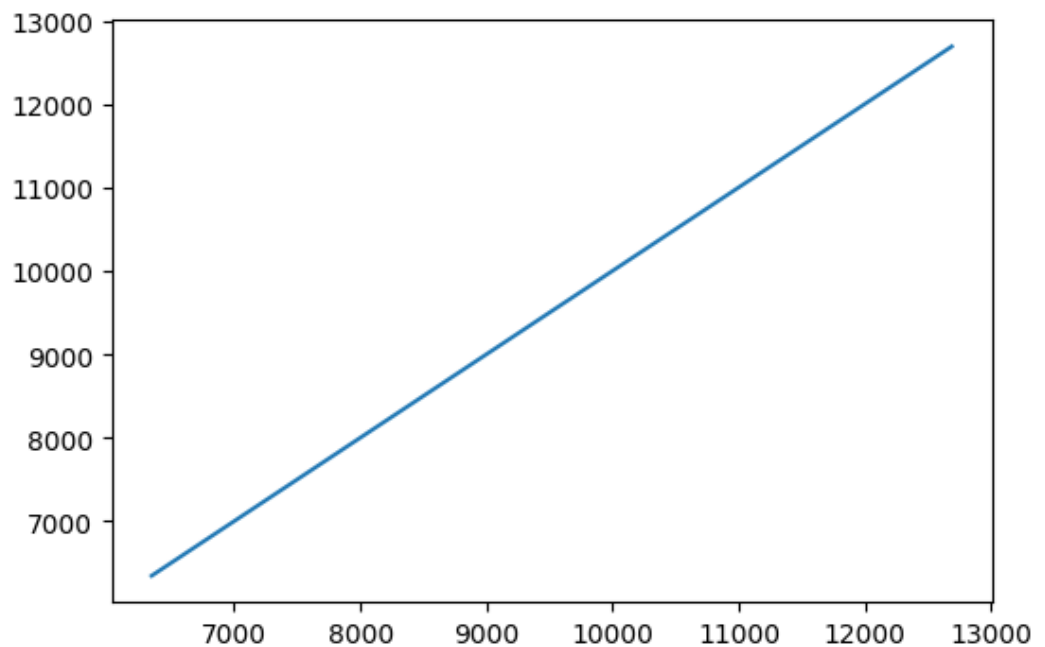
```
import matplotlib.pyplot as plt
df = list(range(1, len(df)+1))
df = pd.DataFrame({'rating': df, 'is_paid': df })
df.head()
df1 = df[0:6345]
df1.head()
```



Pada grafik tersebut, data yang dipakai ialah rating dan is_paid karena kita akan mencari perbandingan antar keduanya. Namun, untuk menampilkan diagram garis ini harus membagi data terlebih dahulu menjadi 2 bagian. Bagian awal dibagi setengah dari jumlah data yaitu 6345. Dengan demikian, diperoleh hasil rating untuk sumbu x dan is_paid untuk sumbu y. Didapati kesimpulan bahwa berbanding lurus, semakin tinggi rating maka tentunya kursus tersebut berbayar.

26. Grafik Line menampilkan banyaknya rating dengan kursus berbayar

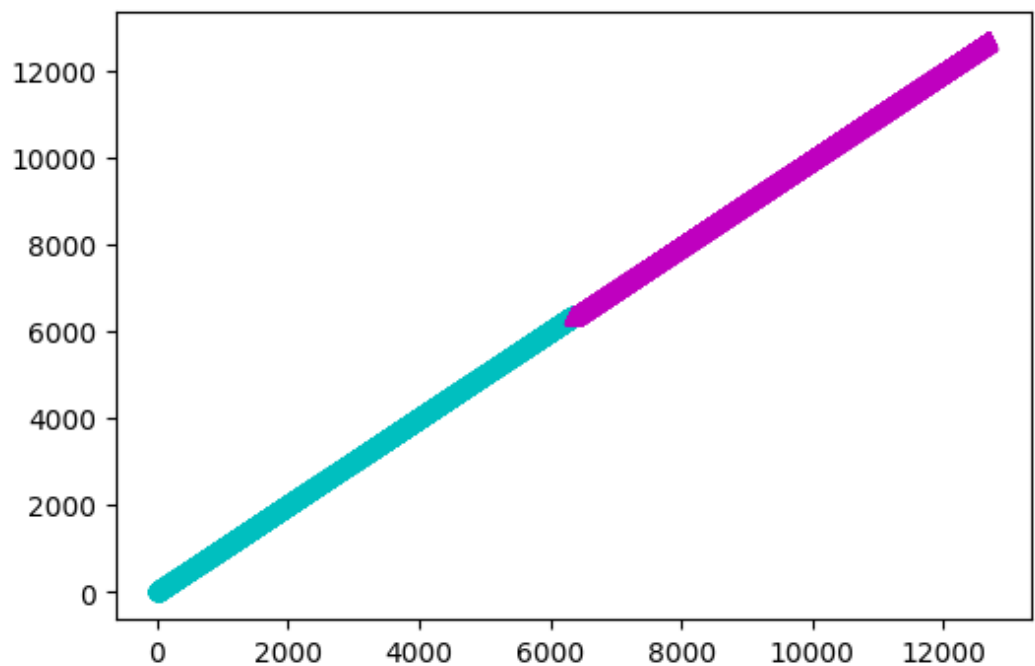
```
df2 = df[6345:12690]
df2.head()
# grafik line hubungan rating dengan kursus berbayar atau tidak
fig = plt.figure(figsize=(6,4))
ax = plt.axes()
ax.plot(df2['rating'],df2['is_paid'])
```



Grafik diatas merupakan lanjutan dari line grafik diatas yakni untuk baris >6345. Didapati hasil dari grafik tersebut masih sama yakni berbanding lurus.

27. Grafik Line Gabungan

```
# gabungan antara 2 tabel diatas
ax.set_title('Hubungan rating dengan is_paid')
ax.set_ylabel('is_paid')
ax.set_xlabel('rating')
fig = plt.figure(figsize=(6,4))
ax = plt.axes()
ax.plot(df1['rating'],df1['is_paid'],'co')
ax.plot(df2['rating'],df2['is_paid'],'m^')
```



Grafik tersebut adalah gabungan dari seluruh data, grafik ini masih sama yakni menunjukkan hubungan rating dengan kursus berbayar. Didapati pada hasil karena 2 grafik sebelumnya berbanding lurus, grafik ini sama halnya dengan yang diatas. Semakin tinggi ratingnya maka kursus tersebut bersifat berbayar.

28. Grafik Pie Persentase Subscriber dengan Kategori Rating

```
#diagram pie presentase subscriber dengan kategori discount_price
%matplotlib inline
plt.title('Presentase subscriber dengan kategori rating')
# ambil sampel df 10 baris
sort_10= sort.head(10)

df=(sort_10['discount_price__amount']).value_counts().plot.pie(autopct='
% 1.1f%%',labels=None,legend=True)
ax.bar(sort_10['title'],sort_10['num_subscribers'])
plt.tight_layout()
```

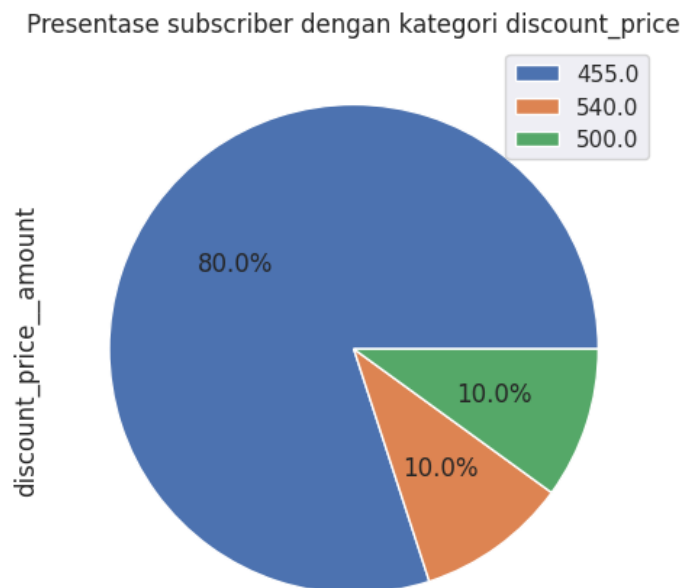


Diagram ini mengelompokkan persentase subscriber berdasarkan kategori discount_price. Persentase discount terbanyak yaitu 455.0 berada pada 80% dari subscriber, persentase discount 540.0 berada 10% dari subscriber, dan persentase 500.0 berada 10% dari subscriber pada data.

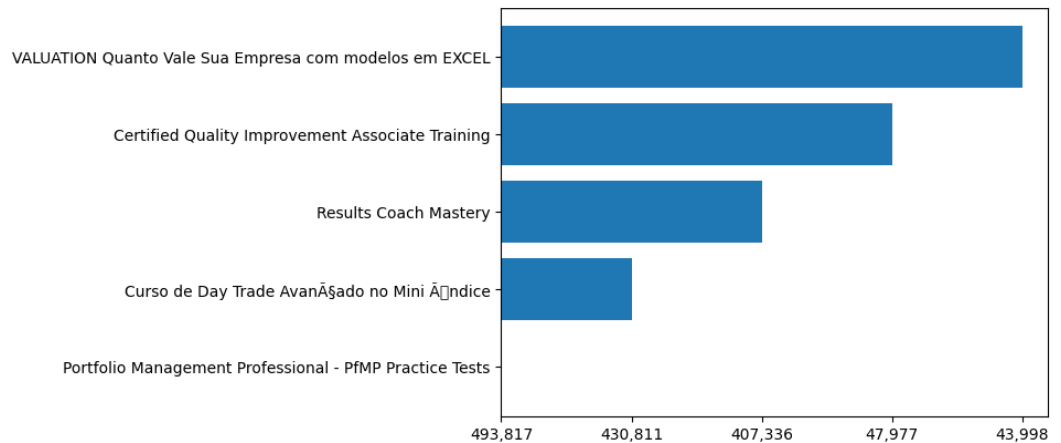
29. Diagram Bar Sortir 5 title dan 5 rating dengan diskon sedikit

```
# sortir 5title dan rating dengan diskon sedikit
```

```
x=df.sort_values(by='discount_price__amount')['title'].tail(5)
```

```
y=df.sort_values(by='discount_price__amount')['rating'].tail(5)
```

```
plt.barh(x,y)
```



Berikut ditampilkan hasil dari grafik dengan diskon paling sedikit, didapati bahwa urutan kursus ialah :

1. Valuation Quanto Vale Sua Empresa com modelos em EXCEL
2. Certified Quality improvement Associate Training
3. Results Coach Mastery
4. Curso de Day
5. Portofolio Management Professional

30. Diagram Boxplot hubungan published_practice dengan diskon yang didapat

```
plt.title('Published Practice dengan Diskon yang didapat')
sns.boxplot(x="num_published_practice_tests", y=
"discount_price__amount", data=df)
```

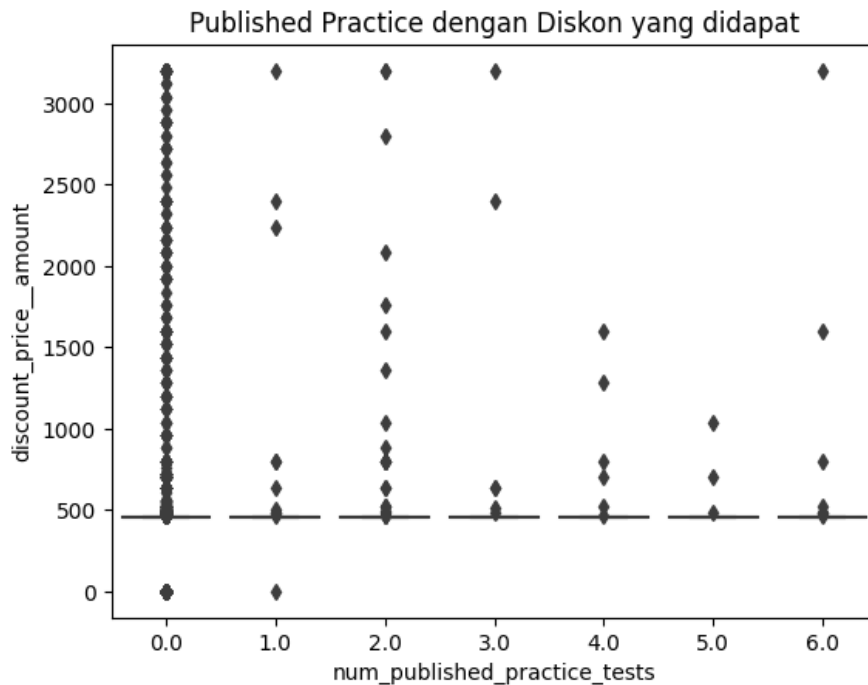


Diagram diatas merupakan boxplot hubungan dari num_published dengan diskon_price. Didapati bahwa publish 0 memiliki banyak sebaran diskon, disusul dengan num_published 2, 1,3,6,4 dan 0. Boxplot ini dengan mudah menandakan posisi sebaran yang tersedia pada data. Hal ini dapat dilihat dari panjang atau pendeknya data yang dihasilkan.

BAB IV

PENUTUP

4.1 Kesimpulan

Analisis Visualisasi Data sangat berguna bagi kehidupan sehari-hari guna memvisualisasikan sebuah data yang telah kita peroleh sebelumnya menjadi bentuk yang menarik dan mudah dipahami oleh orang banyak. Selain itu, visualisasi data dapat menampilkan data yang bersifat efektif. Hal ini untuk data yang dipahami agar bersifat lebih spesifik oleh pembaca hingga dapat memberikan informasi yang bermanfaat tentang proses keuangan Udemy. Dataset Udemy memberikan gambaran terperinci terkait kursus yang disediakan, namun dengan adanya visualisasi data dapat membantu pembaca dalam mengetahui arti dari data tersebut.

4.2 Saran

Dalam penyusunan penerapan Visualisasi Data di bidang keuangan terutama pada dataset yang diperoleh pada kaggle oleh kursus Udemy, tentunya kami masih banyak belajar dan tidak luput dari kata salah. Maka dari itu, kami berharap masukan dari berbagai pihak demi sistem penunjang keputusan yang efektif untuk kedepannya.

DAFTAR PUSTAKA

Kothari, Jil.2021.”Finance & Accounting Courses - Udemy (13K+ course)”

Diakses pada 8 Mei 2023 pukul 11.00 WIB.

Ranti, S. (2023) *Mengenal Google Colab, Fungsi Dan Manfaatnya*,

KOMPAS.com. Available at:

<https://tekno.kompas.com/read/2023/03/26/17150047/mengenal-google-colab-fungsi-dan-manfaatnya-> (Accessed: 13 May 2023).

ID (1980) *Amazon*. Available at: [https://aws.amazon.com/id/what-](https://aws.amazon.com/id/what-is/python/#:~:text=Python%20adalah%20bahasa%20pemrograman%20yang,dapat%20dijalankan%20di%20berbagai%20platform)

[is/python/#:~:text=Python%20adalah%20bahasa%20pemrograman%20yang,dapat%20dijalankan%20di%20berbagai%20platform](https://aws.amazon.com/id/what-is/python/#:~:text=Python%20adalah%20bahasa%20pemrograman%20yang,dapat%20dijalankan%20di%20berbagai%20platform). (Accessed: 13 May 2023).

A., F. (2023) *Apa Itu Python? Yuk Kenalan Dengan Bahasa Pemrograman ini!*,

Hostinger Tutorial. Available at:

<https://www.hostinger.co.id/tutorial/python-adalah> (Accessed: 13 May 2023).