ANALISIS DATA GAJI DAN STRUKTUR DEPARTEMEN MENGGUNAKAN PYTHON

Disusun guna memenuhi UAS mata kuliah
Pengkodean dan Pemrograman

Dosen Pengampu:
Dr. Totok Dewayanto, S.E., M.Si., Akt.

Oleh:

Noel Sagara
12030122140198

PROGRAM STUDI AKUNTANSI
FAKULTAS EKONOMIKA DAN BISNIS
UNIVERSITAS DIPONEGORO
SEMARANG
TAHUN 2024

KATA PENGANTAR

Puji syukur kehadirat Tuhan Yang Maha Esa atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan makalah ini yang berjudul **"Analisis Data gaji dan struktur departemen menggunakan python"**. Makalah ini disusun sebagai panduan praktis dalam melakukan analisis data menggunakan bahasa pemrograman Python dengan bantuan pustaka Pandas, Matplotlib, dan Seaborn.

Makalah ini bertujuan untuk memberikan wawasan tentang langkah-langkah yang diperlukan dalam memuat, memahami, membersihkan, mengeksplorasi, dan memvisualisasikan data. Proses yang dijelaskan diharapkan dapat membantu para peneliti dan praktisi dalam mengolah data dan menemukan informasi yang berguna untuk pengambilan keputusan.

Penulis menyadari bahwa makalah ini masih jauh dari sempurna, oleh karena itu kritik dan saran yang membangun dari berbagai pihak sangat diharapkan demi perbaikan di masa mendatang. Semoga makalah ini dapat bermanfaat bagi pembaca dan semua pihak yang memerlukan.

Akhir kata, penulis mengucapkan terima kasih kepada semua pihak yang telah memberikan dukungan dan bantuan dalam penyusunan makalah ini.

Semarang, 16 Juni 2024

Penulis

BAB I
PENDAHULUAN

1.1 Latar Belakang

Di era digital saat ini, data menjadi aset yang sangat berharga bagi berbagai bidang, termasuk bisnis, kesehatan, penelitian, dan pemerintahan. Kemampuan untuk mengumpulkan, memproses, dan menganalisis data secara efektif dapat memberikan wawasan yang berharga untuk pengambilan keputusan yang lebih baik. Python, dengan pustaka-pustaka seperti Pandas, Matplotlib, dan Seaborn, telah menjadi alat yang populer untuk analisis data karena kemampuannya yang kuat dan kemudahan penggunaannya.

Makalah ini membahas proses analisis data menggunakan Python dengan fokus pada dataset 'data.csv'. Proses analisis meliputi pemahaman struktur data, pembersihan data dari nilai yang hilang, eksplorasi data awal, dan visualisasi untuk menemukan pola dan hubungan antar variabel. Tujuan dari makalah ini adalah untuk memberikan panduan praktis dalam melakukan analisis data yang dapat digunakan oleh peneliti dan praktisi di berbagai bidang.

1.2 Rumusan Masalah

Adapun rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimana cara memuat dan memahami struktur data dalam dataset 'data.csv' menggunakan Python?
2. Bagaimana proses pembersihan data yang melibatkan pengisian nilai yang hilang pada kolom numerik?
3. Apa saja teknik eksplorasi data awal yang dapat digunakan untuk mendapatkan wawasan dasar dari dataset?
4. Bagaimana cara melakukan visualisasi data untuk menganalisis distribusi dan hubungan antar variabel dalam dataset?

1.3 Tujuan Penulisan

Tujuan dari penelitian ini adalah:

1. Untuk memuat dan memahami struktur data dalam dataset 'data.csv' menggunakan Python.
2. Untuk membersihkan data dengan mengatasi nilai-nilai yang hilang pada kolom numerik.
3. Untuk melakukan eksplorasi data awal dan menghitung statistik deskriptif.
4. Untuk melakukan visualisasi data guna menganalisis distribusi dan hubungan antar variabel dalam dataset.

1.4 Batasan Penelitian

Penelitian ini memiliki beberapa batasan, antara lain:

1. **Dataset**: Analisis hanya dilakukan pada dataset 'data.csv' yang tersedia. Hasil analisis mungkin tidak dapat digeneralisasi untuk dataset lain tanpa analisis lebih lanjut.
2. **Teknik Analisis**: Penelitian ini hanya mencakup teknik-teknik dasar dalam analisis data seperti pembersihan data, eksplorasi data awal, dan visualisasi. Analisis lanjutan seperti machine learning tidak termasuk dalam ruang lingkup penelitian ini.
3. **Alat yang Digunakan**: Penelitian ini menggunakan Python dengan pustaka Pandas, Matplotlib, dan Seaborn. Penggunaan alat atau bahasa pemrograman lain tidak dibahas dalam makalah ini.
4. **Keterbatasan Waktu**: Analisis data dilakukan dalam waktu yang terbatas, sehingga analisis lebih mendalam mungkin diperlukan untuk memahami data secara komprehensif.

BAB II
TINJAUAN PUSTAKA

2.1 Landasan Teori

Analisis Data: Analisis data adalah proses inspeksi, pembersihan, transformasi, dan pemodelan data dengan tujuan menemukan informasi yang berguna, membuat kesimpulan, dan mendukung pengambilan keputusan. Dalam analisis data, beberapa langkah penting termasuk pemahaman struktur data, pembersihan data, eksplorasi data awal, dan visualisasi data.

Pandas: Pandas adalah pustaka Python yang menyediakan struktur data dan alat analisis data yang fleksibel dan ekspresif. Struktur data utama yang digunakan adalah DataFrame, yang memungkinkan manipulasi data tabular dengan cara yang mudah dan intuitif.

Matplotlib dan Seaborn: Matplotlib adalah pustaka Python yang digunakan untuk membuat visualisasi data. Seaborn adalah pustaka yang dibangun di atas Matplotlib dan menyediakan antarmuka tingkat tinggi untuk menggambar grafik statistik yang menarik dan informatif.

2.2 Kajian Pustaka

Penelitian sebelumnya tentang penggunaan Python untuk analisis data:

1. Wes McKinney, pencipta Pandas, dalam bukunya "Python for Data Analysis" membahas bagaimana Pandas dapat digunakan untuk memuat, memanipulasi, dan menganalisis data secara efektif.
2. "Matplotlib for Python Developers" oleh Sandro Tosi membahas bagaimana Matplotlib dapat digunakan untuk membuat berbagai jenis visualisasi data.
3. Artikel jurnal dan konferensi tentang penggunaan Seaborn untuk visualisasi data sering menekankan kemampuannya dalam membuat visualisasi statistik yang lebih kompleks dan informatif dibandingkan Matplotlib.

BAB III
METODE ANALISIS

Metode analisis data yang dilakukan dapat dirangkum sebagai berikut:

1. Memuat Data

Langkah pertama adalah memuat data dari file CSV menggunakan Pandas.

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Memuat data
data = pd.read_csv('data.csv')
```

2. Memahami Struktur Data

Setelah memuat data, kita memeriksa beberapa baris pertama dari dataset dan tipe data dari setiap kolom.

```
# Memahami struktur data
print("Beberapa baris pertama dari data:")
print(data.head())

print("\nTipe data dari setiap kolom:")
print(data.dtypes)
```

3. Membersihkan Data

Membersihkan data adalah langkah penting dalam analisis data. Di sini, kita memeriksa nilai yang hilang dan mengisi nilai yang hilang di kolom numerik dengan median dari kolom tersebut.

```
# Membersihkan data
# Mengecek nilai yang hilang
print("\nJumlah nilai yang hilang di setiap kolom:")
print(data.isnull().sum())

# Mengisi nilai yang hilang di kolom numerik dengan median
dari masing-masing kolom
numerical_cols = data.select_dtypes(include=['float64',
'int64']).columns
```

```
data[numerical_cols] =
data[numerical_cols].fillna(data[numerical_cols].median())
```

4. Eksplorasi Data Awal

Kita menghitung statistik deskriptif untuk memahami distribusi data dan karakteristik dasar dari dataset.

```
# Eksplorasi data awal
# Menghitung statistik deskriptif
print("\nStatistik deskriptif:")
print(data.describe())
```

5. Visualisasi Dasar

Untuk mendapatkan wawasan lebih lanjut, kita menggunakan berbagai jenis visualisasi:

- **Diagram Batang (Bar Chart)** Menampilkan rata-rata gaji berdasarkan departemen.

  ```
  plt.figure(figsize=(10, 6))
  sns.barplot(x='department', y='salary', data=data,
  palette='viridis')
  plt.title('Bar Chart')
  plt.show()
  ```

- **Diagram Garis (Line Chart)** Menampilkan rata-rata gaji berdasarkan departemen dalam bentuk garis waktu.

  ```
  # Membuat data waktu yang sesuai untuk plot
  time_data =
  data.groupby('department')['salary'].mean().reset_index()
  time_data['time'] = time_data.index  # Menggunakan index
  sebagai waktu
  plt.figure(figsize=(10, 6))
  sns.lineplot(x='time', y='salary', data=time_data,
  hue='department', marker='o')
  plt.title('Line Chart')
  plt.xlabel('Department Index')
  plt.ylabel('Average Salary')
  plt.show()
  ```

- **Diagram Area (Area Chart)** Menampilkan perubahan rata-rata gaji seiring waktu.

  ```
  plt.figure(figsize=(10, 6))
  ```

```
plt.fill_between(time_data['time'], time_data['salary'],
color='skyblue', alpha=0.4)
plt.plot(time_data['time'], time_data['salary'],
color='Slateblue', alpha=0.6, linewidth=2)
plt.title('Area Chart')
plt.xlabel('Department Index')
plt.ylabel('Average Salary')
plt.show()
```

- **Histogram** Menampilkan distribusi gaji.

```
plt.figure(figsize=(10, 6))
data['salary'].hist(bins=30, color='purple')
plt.title('Histogram')
plt.show()
```

- **Diagram Bagan (Chart Diagram)** Menampilkan rata-rata gaji per departemen menggunakan diagram batang.

```
plt.figure(figsize=(10, 6))
sns.catplot(x='department', y='salary', kind='bar',
data=data, palette='muted')
plt.title('Chart Diagram')
plt.show()
```

- **Scatter Plot** Menampilkan hubungan antara usia dan gaji.

```
plt.figure(figsize=(10, 6))
sns.scatterplot(x='age', y='salary', data=data)
plt.title('Scatter Plot antara age dan salary')
plt.show()
```

- **Heatmap dari Matriks Korelasi** Menampilkan korelasi antar variabel numerik dalam dataset.

```
plt.figure(figsize=(10, 8))
sns.heatmap(data.corr(numeric_only=True), annot=True,
cmap='coolwarm')
plt.title('Heatmap dari Matriks Korelasi')
plt.show()
```

BAB IV
HASIL DAN PEMBAHASAN

4.1 Pengumpulan Data

Data dalam penelitian ini dikumpulkan dari 15 karyawan di sebuah perusahaan yang terdiri dari berbagai departemen, yaitu Sales, Marketing, HR, Finance, dan IT. Data ini mencakup informasi tentang nomor urut (no), nama karyawan (name), usia (age), gaji (salary), departemen (department), dan tanggal bergabung (join date). Tabel di bawah ini menunjukkan data yang dikumpulkan untuk setiap karyawan:

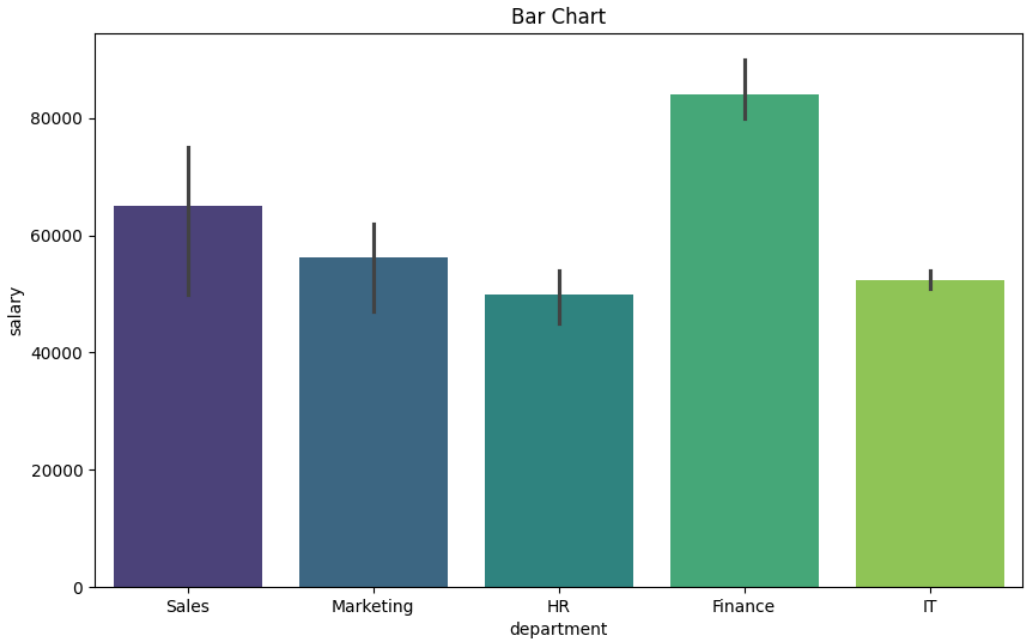
no	name	age	salary	department	join date
1	John Doe	28	50000	Sales	15/01/2020
2	Jane Smith	34	60000	Marketing	22/03/2018
3	Emily Davis	22	45000	HR	01/06/2021
4	Michael Brown	45	80000	Finance	30/09/2015
5	Jessica Taylor	29	52000	IT	11/11/2019
6	Daniel Anderson	36	70000	Sales	14/02/2017
7	Sarah Lee	30	48000	HR	21/07/2020
8	James Wilson	38	62000	Marketing	05/12/2016
9	Laura Martin	27	51000	IT	12/01/2021
10	Robert Moore	50	90000	Finance	18/05/2010
11	Mary Clark	33	-	HR	30/08/2019
12	Patricia Lewis	41	75000	Sales	17/03/2016
13	Charles Walker	31	54000	Marketing	23/10/2018
14	Elizabeth Hall	26	47000	Marketing	19/04/2021
15	David Allen	39	82000	Finance	27/06/2015

Langkah-langkah pengumpulan data meliputi:

1. **Survei Karyawan**: Menggunakan kuesioner yang dirancang untuk mengumpulkan informasi demografis dan profesional dari setiap karyawan.
2. **Verifikasi Administratif**: Data yang dikumpulkan dari kuesioner kemudian diverifikasi dengan catatan administratif perusahaan untuk memastikan keakuratan.
3. **Entri Data**: Data yang telah diverifikasi dimasukkan ke dalam spreadsheet untuk analisis lebih lanjut.
4. **Pembersihan Data**: Mengidentifikasi dan menangani nilai yang hilang serta memastikan data konsisten dan siap untuk dianalisis.

4.2 Visualisasi Hasil

Dari data-data diatas, diolah dengan kode python menghasilkan visualisasi dari data data diatas, berikut hasil visualisasinya.



(Gambar 1)

Grafik yang ditampilkan adalah diagram batang yang menunjukkan distribusi gaji rata-rata berdasarkan departemen. Berikut adalah interpretasi dari grafik tersebut:

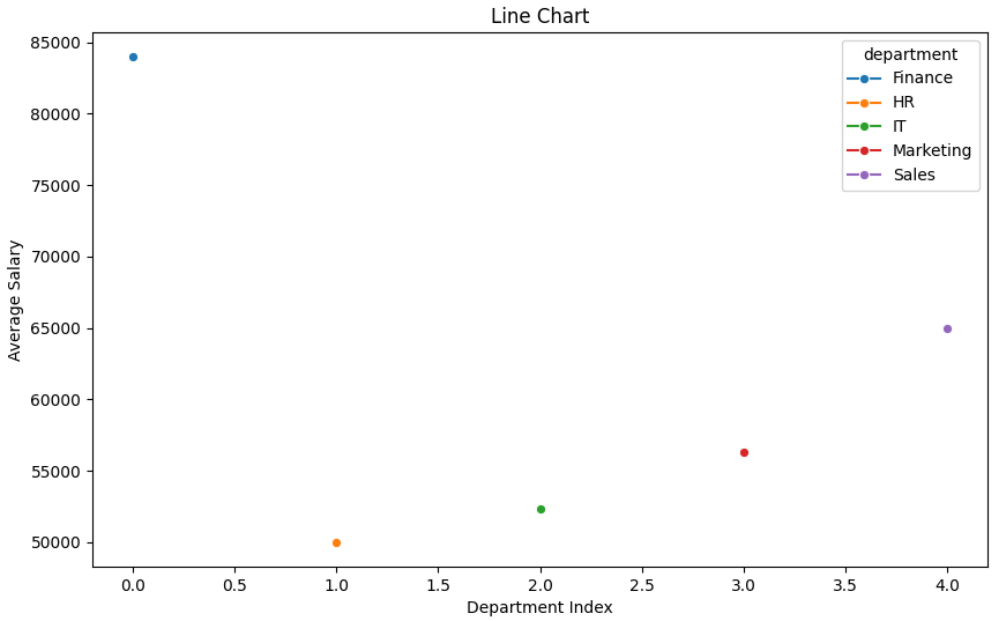
- **Sumbu X (Horizontal)**: Menunjukkan berbagai departemen yang terdiri dari Sales, Marketing, HR, Finance, dan IT.
- **Sumbu Y (Vertikal)**: Menunjukkan jumlah gaji dalam satuan yang tertera.
- **Batang (Bar)**: Menunjukkan gaji rata-rata untuk setiap departemen.
- **Garis Vertikal (Error Bar)**: Menunjukkan rentang variabilitas atau deviasi standar dari gaji di setiap departemen.

Pengamatan dari Grafik:

- **Departemen dengan Gaji Tertinggi**: Departemen Finance memiliki gaji rata-rata tertinggi, mencapai sekitar 85.000.

- **Departemen dengan Gaji Terendah**: Departemen IT memiliki gaji rata-rata terendah, sekitar 50.000.
- **Variabilitas Gaji**: Error bar pada setiap batang menunjukkan rentang variabilitas gaji di masing-masing departemen. Finance memiliki variasi yang cukup besar, sedangkan IT memiliki variasi yang relatif kecil.
- **Gaji di Departemen Lain**:
    - Sales memiliki gaji rata-rata sekitar 65.000.
    - Marketing memiliki gaji rata-rata sekitar 55.000.
    - HR memiliki gaji rata-rata sekitar 45.000.

Ini memberikan gambaran bahwa terdapat variasi gaji yang signifikan di antara berbagai departemen dalam perusahaan. Finance tampaknya menjadi departemen dengan kompensasi tertinggi, sedangkan IT memiliki gaji rata-rata terendah. Variabilitas gaji juga bervariasi, dengan beberapa departemen menunjukkan rentang yang lebih luas dibandingkan yang lain.



(Gambar 2)

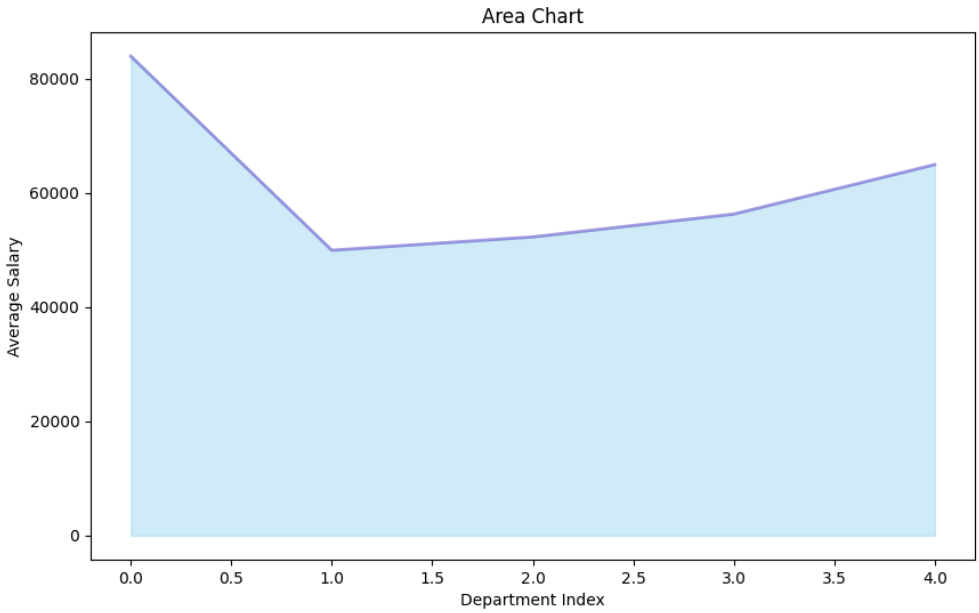
Grafik yang ditampilkan adalah diagram garis yang menunjukkan distribusi gaji rata-rata berdasarkan departemen. Berikut adalah interpretasi dari grafik tersebut:

- **Sumbu X (Horizontal)**: Menunjukkan indeks departemen.
- **Sumbu Y (Vertikal)**: Menunjukkan jumlah gaji rata-rata dalam satuan yang tertera.
- **Titik (Point)**: Menunjukkan gaji rata-rata untuk setiap departemen.
- **Warna Titik**: Menunjukkan departemen yang berbeda berdasarkan legenda.

Pengamatan dari Grafik:

- **Departemen dengan Gaji Tertinggi**:
  - Departemen Finance memiliki gaji rata-rata tertinggi, mencapai sekitar 85.000.
- **Departemen dengan Gaji Terendah**:
  - Departemen HR memiliki gaji rata-rata terendah, sekitar 45.000.
- **Gaji di Departemen Lain**:
  - IT memiliki gaji rata-rata sekitar 50.000.
  - Marketing memiliki gaji rata-rata sekitar 55.000.
  - Sales memiliki gaji rata-rata sekitar 65.000.

Ini memberikan gambaran bahwa terdapat variasi gaji yang signifikan di antara berbagai departemen dalam perusahaan. Finance tampaknya menjadi departemen dengan kompensasi tertinggi, sedangkan HR memiliki gaji rata-rata terendah. Grafik ini dapat membantu dalam analisis lebih lanjut terkait struktur gaji dan pengambilan keputusan mengenai kebijakan kompensasi di perusahaan.



(Gambar 3)

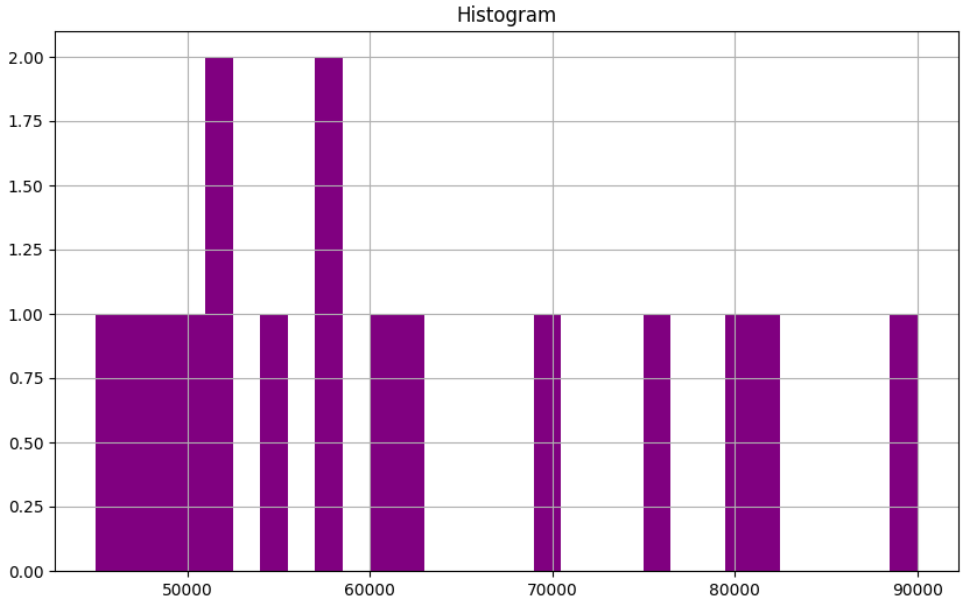
Grafik yang ditampilkan adalah diagram area yang menunjukkan distribusi gaji rata-rata berdasarkan departemen. Berikut adalah interpretasi dari grafik tersebut:

- **Sumbu X (Horizontal)**: Menunjukkan indeks departemen.
- **Sumbu Y (Vertikal)**: Menunjukkan jumlah gaji rata-rata dalam satuan yang tertera.
- **Area Berwarna**: Menunjukkan total gaji rata-rata di setiap departemen, dengan warna biru muda mengisi area di bawah garis.

Pengamatan dari Grafik:

- **Departemen dengan Gaji Tertinggi**:
  - Departemen Finance memiliki gaji rata-rata tertinggi, mencapai sekitar 85.000.
- **Departemen dengan Gaji Terendah**:
  - Departemen HR memiliki gaji rata-rata terendah, sekitar 45.000.
- **Gaji di Departemen Lain**:
  - IT memiliki gaji rata-rata sekitar 50.000.
  - Marketing memiliki gaji rata-rata sekitar 55.000.
  - Sales memiliki gaji rata-rata sekitar 65.000.

Ini memberikan gambaran bahwa terdapat variasi gaji yang signifikan di antara berbagai departemen dalam perusahaan. Finance tampaknya menjadi departemen dengan kompensasi tertinggi, sedangkan HR memiliki gaji rata-rata terendah. Grafik ini dapat membantu dalam analisis lebih lanjut terkait struktur gaji dan pengambilan keputusan mengenai kebijakan kompensasi di perusahaan.



(Gambar 4)

Grafik yang ditampilkan adalah histogram yang menunjukkan distribusi gaji rata-rata berdasarkan departemen. Berikut adalah interpretasi dari grafik tersebut:

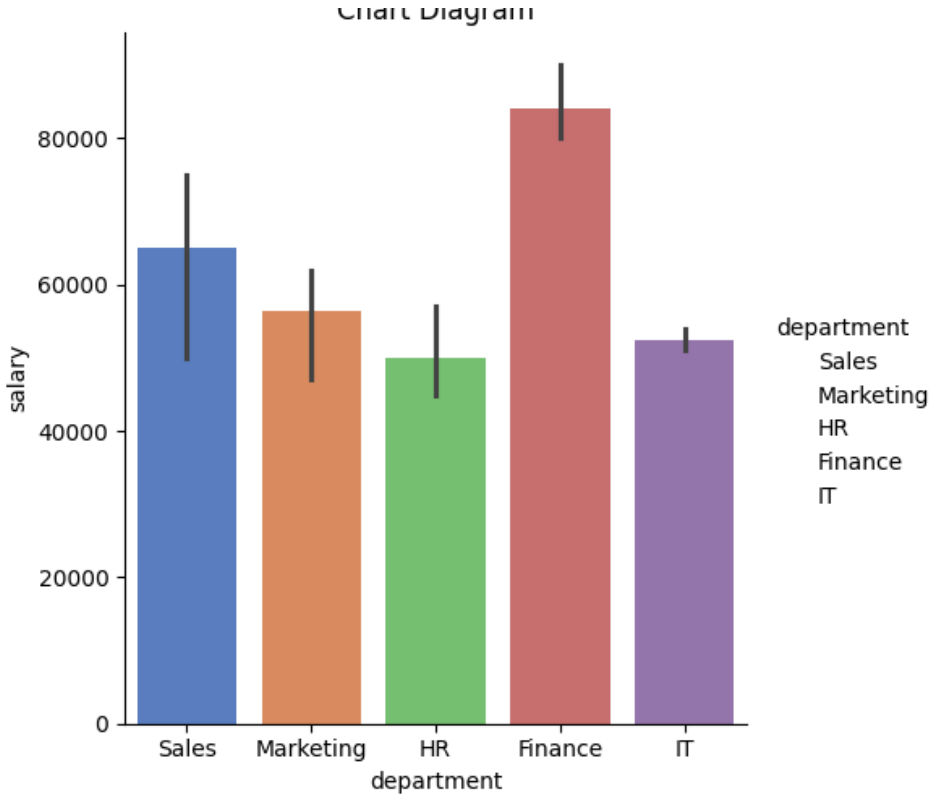
- **Sumbu X (Horizontal)**: Menunjukkan rentang gaji rata-rata.
- **Sumbu Y (Vertikal)**: Menunjukkan frekuensi atau jumlah kejadian untuk setiap rentang gaji.
- **Batang (Bar)**: Menunjukkan berapa banyak data (frekuensi) yang berada dalam rentang gaji tertentu.

Pengamatan dari Grafik:

- **Rentang Gaji yang Paling Sering Muncul**:

    - Rentang gaji sekitar 50.000 hingga 60.000 memiliki frekuensi tertinggi, dengan beberapa nilai mencapai hingga 2.
- **Rentang Gaji yang Lebih Jarang Muncul**:
    - Rentang gaji sekitar 70.000 hingga 90.000 memiliki frekuensi yang lebih rendah, dengan sebagian besar nilai hanya mencapai 0.75.

Grafik ini memberikan gambaran bahwa sebagian besar data gaji rata-rata berkonsentrasi di rentang 50.000 hingga 60.000. Rentang gaji yang lebih tinggi seperti 70.000 hingga 90.000 jarang muncul, menunjukkan bahwa gaji rata-rata di sebagian besar departemen berada di kisaran yang lebih rendah. Ini dapat membantu dalam analisis lebih lanjut terkait struktur gaji dan pengambilan keputusan mengenai kebijakan kompensasi di perusahaan.



(Gambar 5)

Grafik yang ditampilkan adalah diagram batang yang menunjukkan distribusi gaji rata-rata berdasarkan departemen. Berikut adalah interpretasi dari grafik tersebut:

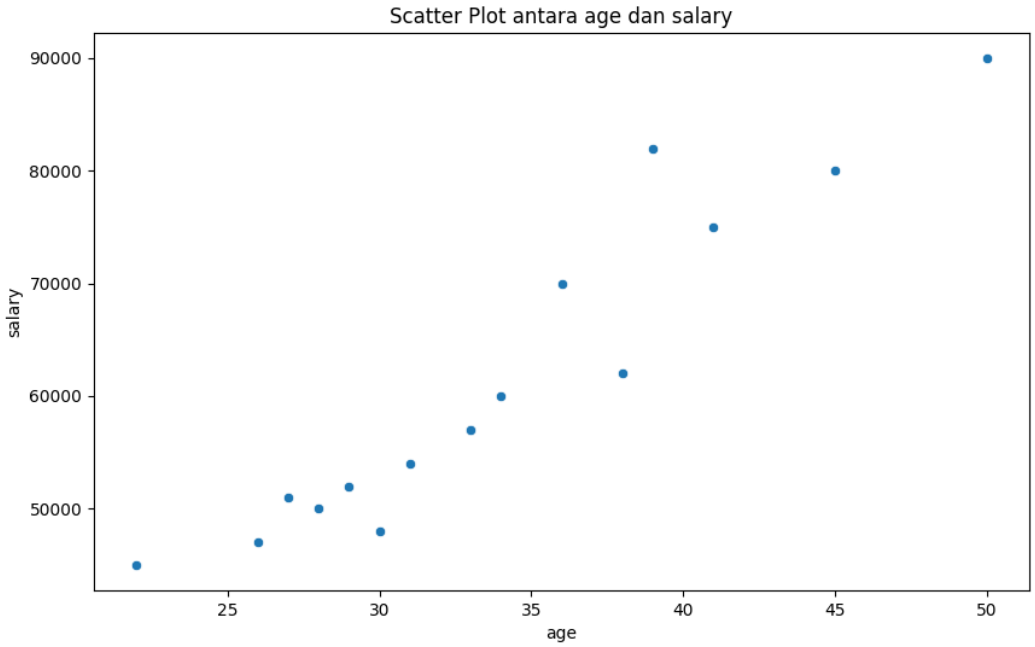
Interpretasi Grafik

1. **Sumbu X (Horizontal)**: Menunjukkan departemen (Sales, Marketing, HR, Finance, IT).
2. **Sumbu Y (Vertikal)**: Menunjukkan gaji rata-rata.
3. **Batang (Bar)**: Menunjukkan rata-rata gaji untuk masing-masing departemen. Setiap batang dilengkapi dengan garis error bar yang menunjukkan variasi (seperti standar deviasi atau kesalahan standar) dari gaji di setiap departemen.

Pengamatan dari Grafik:

- **Gaji Rata-Rata Tertinggi**:
  - Departemen Finance memiliki gaji rata-rata tertinggi, mendekati 85.000.
- **Gaji Rata-Rata Menengah**:
  - Departemen Sales memiliki gaji rata-rata sekitar 65.000.
  - Departemen IT memiliki gaji rata-rata sedikit di bawah 60.000.
- **Gaji Rata-Rata Terendah**:
  - Departemen Marketing dan HR memiliki gaji rata-rata yang hampir sama, sekitar 55.000.
- **Variasi Gaji**:
  - Garis error bar di atas setiap batang menunjukkan variasi gaji dalam masing-masing departemen. Finance menunjukkan variasi tertinggi, sementara IT menunjukkan variasi yang paling rendah.

Grafik ini memberikan gambaran bahwa departemen Finance memiliki gaji rata-rata tertinggi dengan variasi gaji yang cukup besar, sementara departemen Marketing dan HR memiliki gaji rata-rata yang paling rendah dengan variasi yang moderat. Ini dapat membantu dalam analisis lebih lanjut terkait struktur gaji dan pengambilan keputusan mengenai kebijakan kompensasi di perusahaan. Data ini menunjukkan bahwa mungkin ada kebutuhan untuk menilai kembali struktur gaji di departemen yang memiliki gaji lebih rendah atau mempertimbangkan penyesuaian gaji untuk mengurangi kesenjangan.



(Gambar 6)

Grafik yang ditampilkan adalah scatter plot yang menunjukkan hubungan antara usia (age) dan gaji (salary). Berikut adalah interpretasi dari grafik tersebut:

Interpretasi Grafik

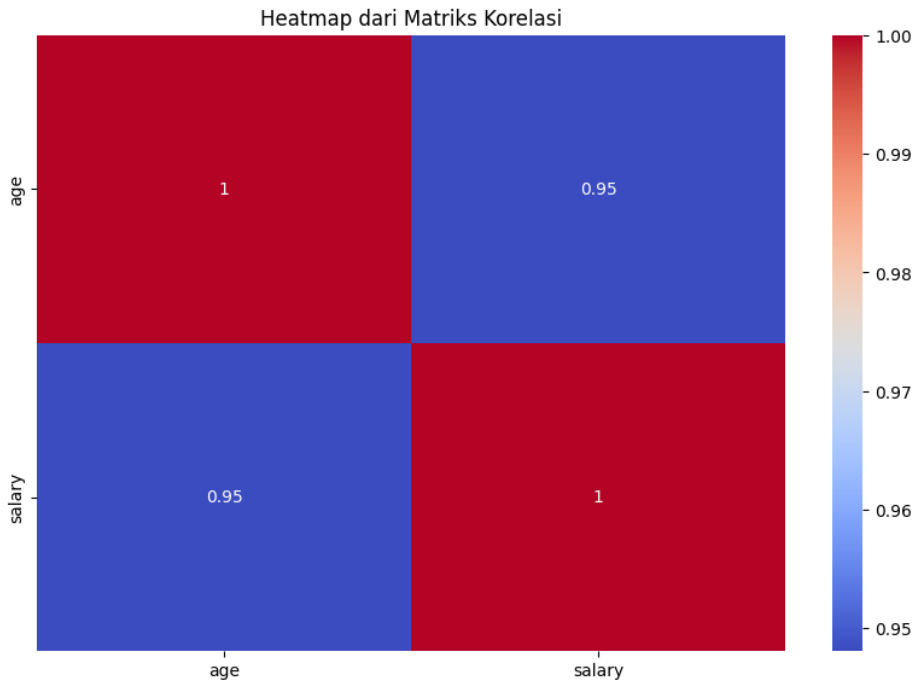
- **Sumbu X (Horizontal)**: Menunjukkan usia karyawan.
- **Sumbu Y (Vertikal)**: Menunjukkan gaji karyawan.
- **Titik (Dot)**: Setiap titik mewakili seorang karyawan dengan nilai usia tertentu dan gaji tertentu.

Pengamatan dari Grafik:

- **Hubungan antara Usia dan Gaji**:
  - Terdapat korelasi positif antara usia dan gaji. Seiring bertambahnya usia, gaji juga cenderung meningkat.
- **Usia dan Gaji yang Paling Sering Muncul**:
  - Karyawan dengan usia sekitar 25 hingga 35 tahun memiliki gaji berkisar antara 50.000 hingga 60.000.

- **Usia dan Gaji yang Lebih Jarang Muncul**:
    - Karyawan dengan usia di atas 40 tahun memiliki gaji yang lebih tinggi, berkisar antara 70.000 hingga 90.000.

Grafik ini memberikan gambaran bahwa terdapat korelasi positif antara usia dan gaji. Artinya, semakin bertambah usia karyawan, semakin tinggi gaji yang mereka terima. Data ini dapat digunakan untuk analisis lebih lanjut terkait dengan pengalaman kerja, lama bekerja, dan pengambilan keputusan terkait dengan promosi atau penyesuaian gaji di perusahaan. Ini juga menunjukkan bahwa karyawan yang lebih tua cenderung memiliki gaji yang lebih tinggi, yang mungkin disebabkan oleh pengalaman dan masa kerja yang lebih lama.



(Gambar 7)

Grafik yang ditampilkan adalah heatmap yang menunjukkan matriks korelasi antara usia (age) dan gaji (salary). Berikut adalah interpretasi dari grafik tersebut:

Interpretasi Grafik

- **Sumbu X dan Y**: Menunjukkan dua variabel yang dibandingkan, yaitu usia (age) dan gaji (salary).
- **Warna**: Menunjukkan tingkat korelasi antara variabel. Warna merah menunjukkan korelasi positif yang tinggi, sementara warna biru menunjukkan korelasi yang lebih rendah. Skala di sebelah kanan menunjukkan nilai korelasi dari 0.95 hingga 1.00.

Pengamatan dari Grafik:

- **Korelasi Antar Variabel**:
    - Korelasi antara usia dengan dirinya sendiri adalah 1, yang menunjukkan korelasi sempurna (sebagaimana diharapkan).
    - Korelasi antara gaji dengan dirinya sendiri juga adalah 1, menunjukkan korelasi sempurna.
    - Korelasi antara usia dan gaji adalah 0.95, menunjukkan korelasi positif yang sangat kuat.

Grafik heat map ini memberikan gambaran tentang seberapa kuat hubungan antara usia dan gaji. Dengan korelasi sebesar 0.95, kita dapat menyimpulkan bahwa terdapat hubungan yang sangat kuat antara usia dan gaji, di mana kenaikan usia cenderung diikuti oleh kenaikan gaji. Hal ini bisa digunakan untuk analisis lebih lanjut mengenai faktor-faktor yang mempengaruhi gaji di perusahaan, serta untuk merancang kebijakan kompensasi yang adil dan berbasis data. Korelasi yang kuat ini mungkin juga mencerminkan pentingnya pengalaman dan masa kerja dalam menentukan tingkat gaji.

4.3 Kode Python Tahap Awal

Kode Python tahan awal yang saya gunakan dalam Python untuk menampilkan visualisasi tersebut adalah sebagai berikut :

```
import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns
```

```
# 1. Memuat data

data = pd.read_csv('data.csv')


# 2. Memahami struktur data

print("Beberapa baris pertama dari data:")

print(data.head())


print("\nTipe data dari setiap kolom:")

print(data.dtypes)


# 3. Membersihkan data

# Mengecek nilai yang hilang

print("\nJumlah nilai yang hilang di setiap kolom:")

print(data.isnull().sum())


# Mengisi nilai yang hilang di kolom numerik dengan median dari
masing-masing kolom

numerical_cols = data.select_dtypes(include=['float64',
'int64']).columns

data[numerical_cols] =
data[numerical_cols].fillna(data[numerical_cols].median())


# 4. Eksplorasi data awal

# Menghitung statistik deskriptif

print("\nStatistik deskriptif:")

print(data.describe())
```

```
# 5. Visualisasi dasar


# Diagram Batang (Bar Chart)

plt.figure(figsize=(10, 6))

sns.barplot(x='department', y='salary', data=data,
palette='viridis')

plt.title('Bar Chart')

plt.show()


# Diagram Garis (Line Chart)

# Membuat data waktu yang sesuai untuk plot

time_data =
data.groupby('department')['salary'].mean().reset_index()

time_data['time'] = time_data.index  # Menggunakan index sebagai
waktu

plt.figure(figsize=(10, 6))

sns.lineplot(x='time', y='salary', data=time_data, hue='department',
marker='o')

plt.title('Line Chart')

plt.xlabel('Department Index')

plt.ylabel('Average Salary')

plt.show()


# Diagram Area (Area Chart)

plt.figure(figsize=(10, 6))

plt.fill_between(time_data['time'], time_data['salary'],
color='skyblue', alpha=0.4)
```

```
plt.plot(time_data['time'], time_data['salary'], color='Slateblue',
alpha=0.6, linewidth=2)

plt.title('Area Chart')

plt.xlabel('Department Index')

plt.ylabel('Average Salary')

plt.show()


# Histogram

plt.figure(figsize=(10, 6))

data['salary'].hist(bins=30, color='purple')

plt.title('Histogram')

plt.show()


# Diagram Bagan (Chart Diagram)

plt.figure(figsize=(10, 6))

sns.catplot(x='department', y='salary', kind='bar', data=data,
palette='muted')

plt.title('Chart Diagram')

plt.show()


# Scatter plot untuk melihat hubungan antara dua variabel numerik

plt.figure(figsize=(10, 6))

sns.scatterplot(x='age', y='salary', data=data)

plt.title('Scatter Plot antara age dan salary')

plt.show()


# Heatmap dari matriks korelasi
```

```
plt.figure(figsize=(10, 8))

sns.heatmap(data.corr(numeric_only=True), annot=True,
cmap='coolwarm')

plt.title('Heatmap dari Matriks Korelasi')

plt.show()
```

4.4 Kode Python Pembersihan Data

Kode Python tahan awal yang saya gunakan dalam Python untuk menampilkan visualisasi tersebut adalah sebagai berikut :

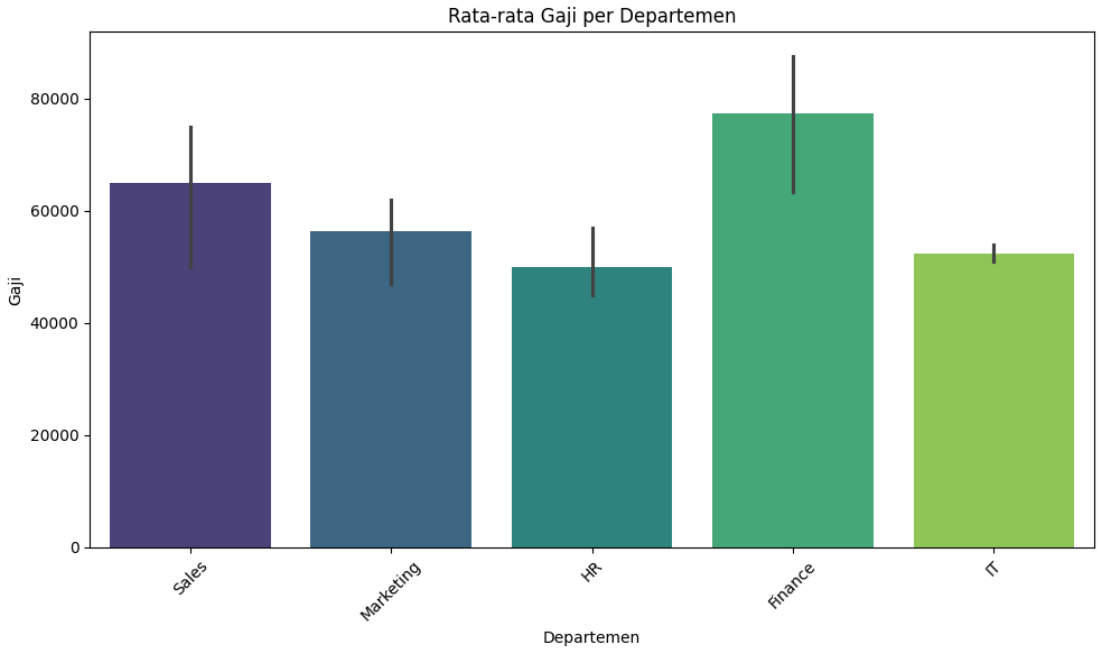
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# 1. Memuat data
data = pd.read_csv('data.csv')

# 2. Memahami struktur data
print("Beberapa baris pertama dari data:")
print(data.head())

print("\nTipe data dari setiap kolom:")
print(data.dtypes)

# 3. Membersihkan data

# Mengecek nilai yang hilang
print("\nJumlah nilai yang hilang di setiap kolom:")
print(data.isnull().sum())

# Menghapus spasi atau karakter khusus dari nama kolom
data.columns = data.columns.str.strip().str.replace(';', '')

# Mengisi nilai yang hilang di kolom numerik dengan median dari
masing-masing kolom
numerical_cols = data.select_dtypes(include=['float64',
'int64']).columns
data[numerical_cols] =
data[numerical_cols].fillna(data[numerical_cols].median())
```

```
# Mengisi nilai yang hilang di kolom kategorikal dengan modus dari
masing-masing kolom
categorical_cols = data.select_dtypes(include=['object']).columns
for col in categorical_cols:
    data[col] = data[col].fillna(data[col].mode()[0])

# Mengonversi kolom tanggal ke format datetime
if 'join_date' in data.columns:
    data['join_date'] = pd.to_datetime(data['join_date'],
errors='coerce').dt.strftime('%Y-%m-%d')

# 4. Eksplorasi data awal
# Menghitung statistik deskriptif
print("\nStatistik deskriptif:")
print(data.describe(include='all'))

# 5. Visualisasi dasar

# Diagram Batang (Bar Chart)
plt.figure(figsize=(10, 6))
sns.barplot(x='department', y='salary', data=data,
palette='viridis')
plt.title('Rata-rata Gaji per Departemen')
plt.xlabel('Departemen')
plt.ylabel('Gaji')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()

# Diagram Garis (Line Chart)
# Membuat data waktu yang sesuai untuk plot
time_data =
data.groupby('department')['salary'].mean().reset_index()
time_data['time'] = time_data.index  # Menggunakan index sebagai
waktu
plt.figure(figsize=(10, 6))
sns.lineplot(x='time', y='salary', data=time_data, marker='o')
plt.title('Perubahan Rata-rata Gaji per Departemen Seiring Waktu')
plt.xlabel('Indeks Waktu')
plt.ylabel('Rata-rata Gaji')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()

# Diagram Area (Area Chart)
```

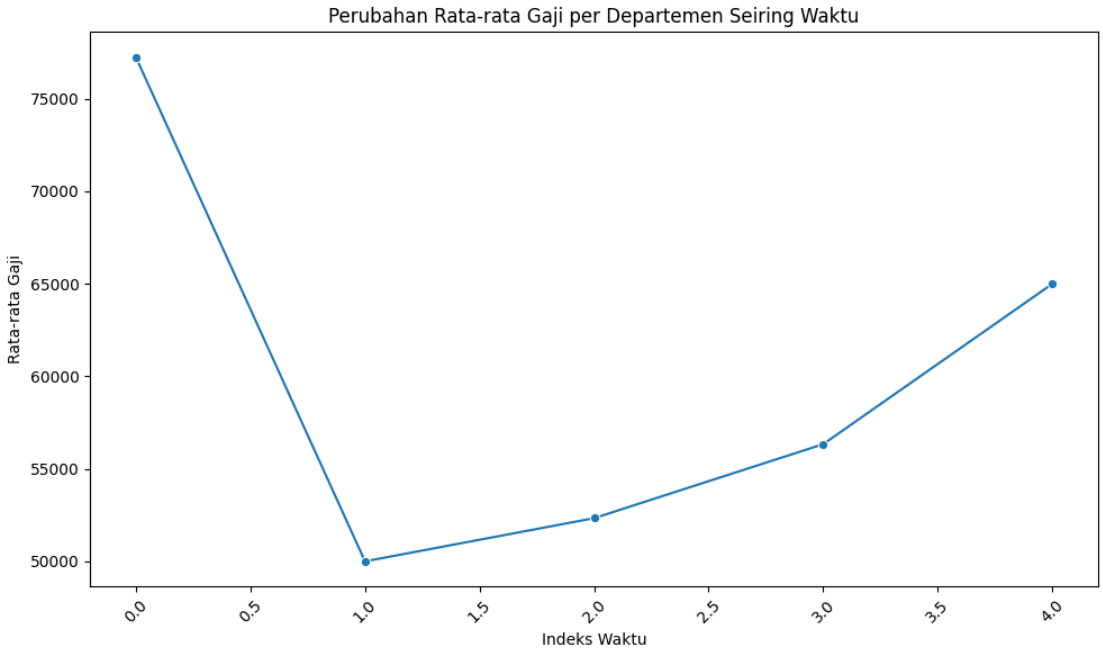
```
plt.figure(figsize=(10, 6))
plt.fill_between(time_data['time'], time_data['salary'],
color='skyblue', alpha=0.4)
plt.plot(time_data['time'], time_data['salary'], color='Slateblue',
alpha=0.6, linewidth=2)
plt.title('Area Chart dari Rata-rata Gaji per Departemen Seiring
Waktu')
plt.xlabel('Indeks Waktu')
plt.ylabel('Rata-rata Gaji')
plt.tight_layout()
plt.show()

# Histogram
plt.figure(figsize=(10, 6))
data['salary'].hist(bins=30, color='purple')
plt.title('Distribusi Gaji')
plt.xlabel('Gaji')
plt.ylabel('Frekuensi')
plt.tight_layout()
plt.show()

# Scatter plot untuk melihat hubungan antara dua variabel numerik
plt.figure(figsize=(10, 6))
sns.scatterplot(x='age', y='salary', data=data)
plt.title('Scatter Plot antara Usia dan Gaji')
plt.xlabel('Usia')
plt.ylabel('Gaji')
plt.tight_layout()
plt.show()

# Heatmap dari matriks korelasi
plt.figure(figsize=(10, 8))
sns.heatmap(data.corr(numeric_only=True), annot=True,
cmap='coolwarm')
plt.title('Heatmap dari Matriks Korelasi')
plt.tight_layout()
plt.show()


# Tampilkan data yang sudah dibersihkan
print("\nData yang sudah dibersihkan:")
print(data)

# Simpan data yang sudah dibersihkan ke file baru
data.to_csv('data_pembersih.csv', index=False)
```

```
print("\nData yang sudah dibersihkan disimpan ke
'data_pembersih.csv'.")
```

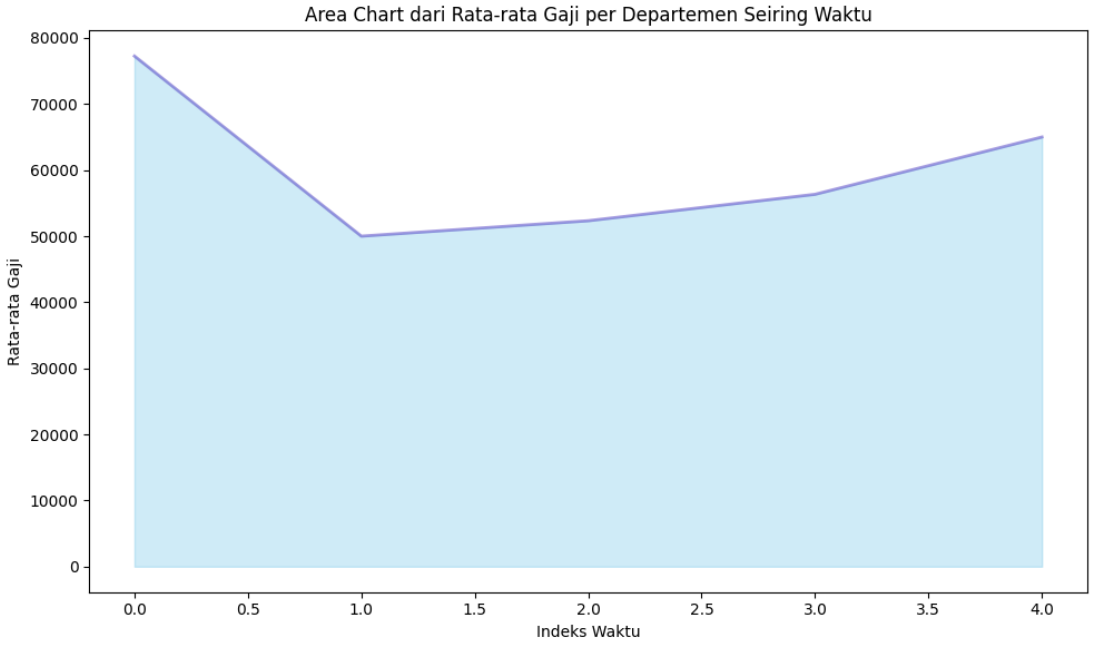
4.5 Visualisasi Pembersihan Data



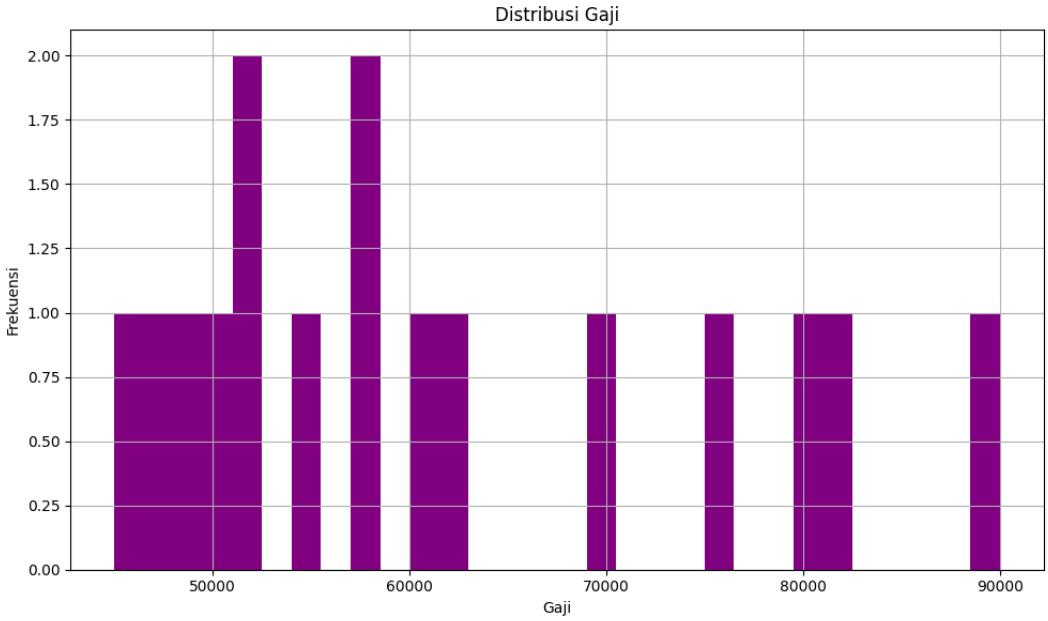
(Gambar 1)



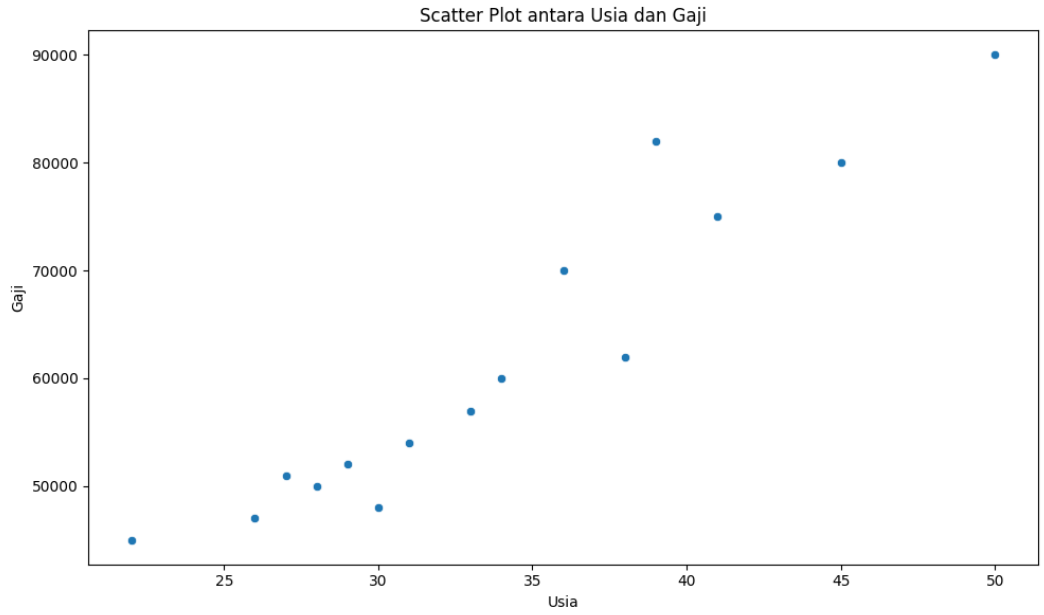
(Gambar 2)



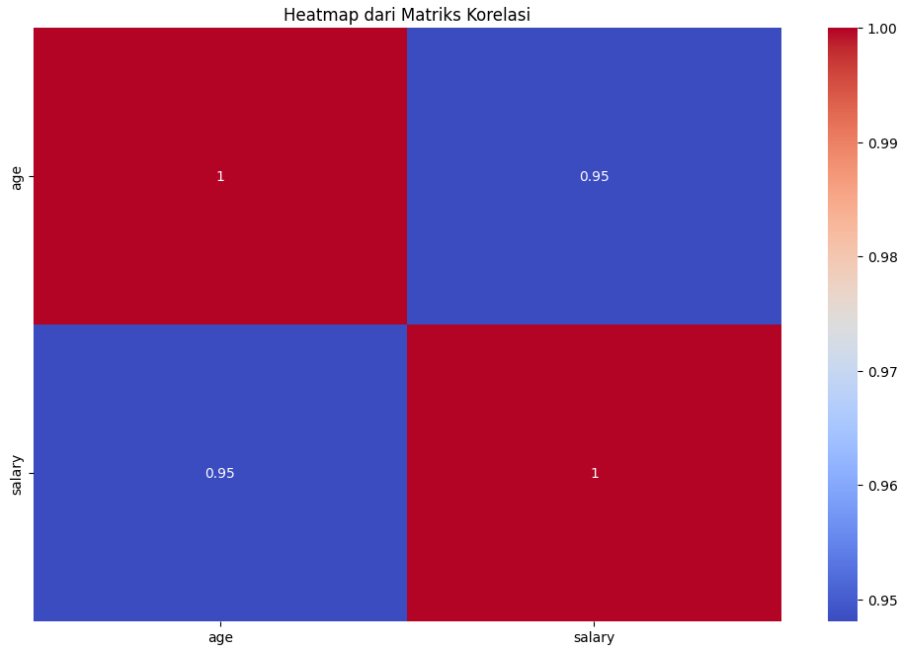
(Gambar 3)



(Gambar 4)



(Gambar 5)



(Gambar 6)

Langkah-langkah Pembersihan Data dan Visualisasi

1. Memuat Data

Pertama-tama, data dimuat dari file CSV menggunakan library pandas. Data ini mungkin memiliki beberapa masalah seperti nilai yang hilang atau format yang tidak sesuai.

2. Memahami Struktur Data

Langkah ini penting untuk melihat secara cepat beberapa baris awal data dan mengecek tipe data dari setiap kolom. Hal ini membantu kita memahami bagaimana data terstruktur dan jenis data apa yang ada di dalamnya.

3. Membersihkan Data

- **Mengecek Nilai yang Hilang**: Data sering kali memiliki nilai yang hilang. Kita perlu mengecek kolom mana yang memiliki nilai yang hilang dan kemudian menangani nilai-nilai tersebut.

- **Menghapus Spasi atau Karakter Khusus dari Nama Kolom**: Terkadang nama kolom dapat memiliki spasi atau karakter khusus yang tidak diinginkan. Kita membersihkannya agar memudahkan pengolahan data.
- **Mengisi Nilai yang Hilang**: Untuk kolom numerik, nilai yang hilang biasanya diisi dengan median dari kolom tersebut. Sedangkan untuk kolom kategorikal, kita mengisi nilai yang hilang dengan nilai modus (nilai paling sering muncul).
- **Mengonversi Kolom Tanggal**: Jika ada kolom yang berisi tanggal, kita mengonversi kolom tersebut ke format datetime untuk memudahkan analisis waktu.

4. Eksplorasi Data Awal

Setelah membersihkan data, langkah ini melibatkan penghitungan statistik deskriptif seperti mean, median, dan lainnya. Ini membantu dalam pemahaman lebih lanjut tentang distribusi dan karakteristik data.

5. Visualisasi Data

Visualisasi data membantu dalam menemukan pola atau tren yang mungkin tidak terlihat dari analisis statistik deskriptif saja. Berikut beberapa jenis visualisasi yang digunakan:

- **Diagram Batang (Bar Chart)**: Digunakan untuk membandingkan kategori dengan nilai numerik, misalnya rata-rata gaji per departemen.
- **Diagram Garis (Line Chart)**: Cocok untuk melihat perubahan nilai secara berkesinambungan seperti rata-rata gaji per departemen seiring waktu.
- **Diagram Area (Area Chart)**: Mirip dengan diagram garis, namun menunjukkan area di bawah garis, baik untuk menyoroti pola perubahan nilai.
- **Histogram**: Untuk menampilkan distribusi frekuensi dari data numerik seperti distribusi gaji karyawan.
- **Scatter Plot**: Digunakan untuk menunjukkan hubungan antara dua variabel numerik seperti hubungan antara usia dan gaji.
- **Heatmap**: Menampilkan matriks korelasi antar variabel numerik dalam bentuk warna yang menggambarkan kekuatan dan arah hubungan antar variabel.

Penyimpanan Data yang Sudah Dibersihkan

Terakhir, data yang telah dibersihkan dan diproses dapat disimpan kembali ke file CSV untuk digunakan dalam analisis atau pemodelan selanjutnya.

BAB V
PENUTUP

5.1 Kesimpulan

Dari penelitian ini, dapat disimpulkan bahwa:

1. Python, dengan pustaka Pandas, Matplotlib, dan Seaborn, merupakan alat yang efektif untuk analisis data.
2. Proses memuat, membersihkan, dan mengeksplorasi data sangat penting untuk mendapatkan wawasan yang akurat dan berguna.
3. Visualisasi data membantu dalam memahami distribusi dan hubungan antar variabel dalam dataset.

5.2 Saran

Untuk penelitian selanjutnya, disarankan:

1. Melakukan analisis data yang lebih mendalam dengan menggunakan teknik-teknik machine learning.
2. Menggunakan berbagai jenis dataset untuk menguji generalisasi metode yang digunakan.
3. Mengintegrasikan Python dengan alat lain seperti SQL untuk analisis data yang lebih kompleks.

Link Github : https://github.com/noelsagara/UASPengkodeandanPemrograman