PORTOFOLIO *mini*

Highlight Project from Digital Skill Fair 41.0: Python & EDA Practice

AUDRY PRITHA EKAHARYANI

ibimbing

HELLO *I'm*
AUDRY PRITHA EKAHARYANI

An Informatics student at Universitas Teknologi Yogyakarta with a strong interest in data analysis and data management.

Throughout my studies, I have learned the fundamentals of data processing and visualization using tools such as Google Colab, VSCode, Google Sheets and RapidMiner.

To deepen these skills in a practical setting, I participated in the Digital Skill Fair 41.0 program by ibimbing , where I learned to apply data exploration, data cleaning and insight presentation techniques using Python.

TOOLS & SKILLS

🛠️ Tools Used

Python

Pandas

Matplotlib

Seaborn

Kaggle

💡 Skills Acquired

Data Cleaning

Exploratory Data Analysis

Data Visualization

Insight Generation

PROJECT 01

This program is designed to classify student exam scores based on user input name, student ID (NIM) and a score ranging from 0 to 100 and then display the corresponding grade category, from A (Excellent) to E (Very Poor).

Problem Statement

How to develop a simple Python program that processes user input, identifies data types and categorizes exam scores based on predefined grading criteria.

Method

- Utilized input() to collect user data and int() to convert the exam score into an integer.
- Displayed data types using type() to ensure input validity.
- Applied if-elif-else statements to categorize scores based on predefined value ranges.

code

D: > tugas_datascience1_audry.py > ...

```python
variabel_nama = "audry"
variabel_nim = 1124022
variabel_nilai = 92

nama = input("Masukkan nama mahasiswa: ")
nim = input("Masukkan NIM: ")
nilai = int(input("Masukkan nilai ujian (0-100): "))

if 85 <= nilai <= 100:
    kategori_nilai = "A (Sangat Baik)"
elif 75 <= nilai <= 84:
    kategori_nilai = "B (Baik)"
elif 60 <= nilai <= 74:
    kategori_nilai = "C (Cukup)"
elif 40 <= nilai <= 59:
    kategori_nilai = "D (Kurang)"
else:
    kategori_nilai = "E (Sangat Kurang)"


print(f"\nNama: {nama} (type: {type(nama)})")
print(f"NIM: {nim} (type: {type(nim)})")
print(f"Nilai: {nilai} (type: {type(nilai)})")

print("\nHasil Evaluasi:")
print(f"Mahasiswa: {nama} (NIM: {nim})")
print(f"Nilai Ujian: {nilai}")
print(f"Kategori Nilai: {kategori_nilai}")
```

output

```
PS C:\Users\user\AppData\Local\Programs\Microsoft VS Code> & C
Masukkan nama mahasiswa: audry
Masukkan NIM: 11221122
Masukkan nilai ujian (0-100): 88

Nama: audry (type: <class 'str'>)
NIM: 11221122 (type: <class 'str'>)
Nilai: 88 (type: <class 'int'>)

Hasil Evaluasi:
Mahasiswa: audry (NIM: 11221122)
Nilai Ujian: 88
Kategori Nilai: A (Sangat Baik)
PS C:\Users\user\AppData\Local\Programs\Microsoft VS Code>
```

The program successfully displays the user input, including the student's name, ID (NIM) and exam score, along with their respective data types.

The exam score is then classified into a grade category based on a predefined range.

In this example, a score of 88 is classified into category A (Excellent).

PROJECT 02

Performed initial cleaning on the sales dataset using Exploratory Data Analysis (EDA), focusing on handling missing values and removing duplicate records as a crucial step to ensure data quality before conducting further analysis.

code

```
data.info()
```

output

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2005 entries, 0 to 2004
Data columns (total 14 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   transaction_id    2004 non-null   object
 1   customer_id       2002 non-null   object
 2   customer_name     2003 non-null   object
 3   customer_age      2003 non-null   float64
 4   gender            2003 non-null   object
 5   product_id        2005 non-null   object
 6   product_name      2005 non-null   object
 7   product_category  2005 non-null   object
 8   quantiy           2005 non-null   int64
 9   prce              2005 non-null   int64
 10  payment_mode      2004 non-null   object
 11  purchase_date     2004 non-null   object
 12  time_of_purchase  2004 non-null   object
 13  status            2004 non-null   object
dtypes: float64(1), int64(2), object(11)
memory usage: 219.4+ KB
PS D:\tugas_datascience2_audry>
```

The dataset consists of 2,005 rows and 14 columns, with most columns having an object data type and several missing values.

```
data.describe()
```

```
PS D:\tugas_datascience2_audry> C:
transaction_id      1
customer_id         3
customer_name       2
customer_age        2
gender              2
product_id          0
product_name        0
product_category    0
quantiy             0
prce                0
payment_mode        1
purchase_date       1
time_of_purchase    1
status              1
dtype: int64
PS D:\tugas_datascience2_audry>
```

Some columns contain missing values (1–3 entries), such as customer_id, customer_name and customer_age, which need to be cleaned before analysis.

```
data.isna().sum()
```

	customer_age	quantiy	prce
count	2003.000000	2005.000000	2005.000000
mean	38.902147	3.043890	7952.523192
std	12.628259	1.411766	6043.285703
min	18.000000	1.000000	101.000000
25%	28.000000	2.000000	2948.000000
50%	39.000000	3.000000	6640.000000
75%	50.000000	4.000000	11745.000000
max	60.000000	5.000000	24985.000000

```
PS D:\tugas_datascience2_audry>
```

Customers are, on average, 39 years old and purchase 3 products per transaction. Product prices vary, with an average of Rp7,952 and a maximum of Rp24,985. This data reflects customer profiles and shopping behavior.

PROJECT 02

Handling missing values in several columns.

code

```
for column in data.columns:
    if data[column].dtype == 'object':
        data[column].fillna(data[column].mode()[0], inplace=True)
    else:
        data[column].fillna(data[column].mean(), inplace=True)
```

output

```
transaction_id      0
customer_id         0
customer_name       0
customer_age        0
gender              0
product_id          0
product_name        0
product_category    0
quantiy             0
prce                0
payment_mode        0
purchase_date       0
time_of_purchase    0
status              0
dtype: int64
PS D:\tugas_datascience2_audry>
```

output

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2005 entries, 0 to 2004
Data columns (total 14 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   transaction_id    2005 non-null   object
 1   customer_id       2005 non-null   object
 2   customer_name     2005 non-null   object
 3   customer_age      2005 non-null   float64
 4   gender            2005 non-null   object
 5   product_id        2005 non-null   object
 6   product_name      2005 non-null   object
 7   product_category  2005 non-null   object
 8   quantiy           2005 non-null   int64
 9   prce              2005 non-null   int64
 10  payment_mode      2005 non-null   object
 11  purchase_date     2005 non-null   object
 12  time_of_purchase  2005 non-null   object
 13  status            2005 non-null   object
dtypes: float64(1), int64(2), object(11)
memory usage: 219.4+ KB
```

All missing values were successfully handled (0 remaining)

PROJECT 02

Checking for duplicates across all columns.

code

```
check_duplicate = data.duplicated().sum()

print(f"Jumlah data yang duplikat = {check_duplicate}")
```

Before removal, 4 duplicate records were detected in the dataset.

output

```
Jumlah data yang duplikat = 4
PS D:\tugas_datascience2_audry>
```

Handling duplicate

```
data = data.drop_duplicates()
```

Checking for duplicates after handling them.

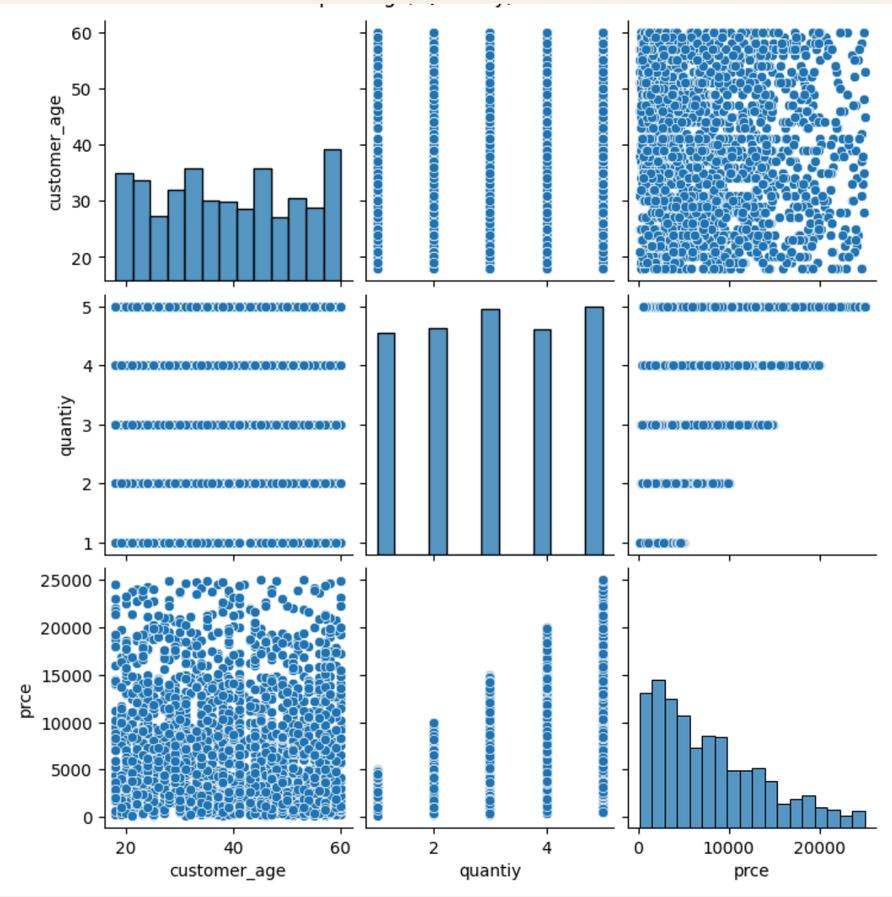
```
handle_duplicate = data.duplicated().sum()

print(f"Jumlah data yang duplikat = {handle_duplicate}")
```

After applying drop_duplicates(), no duplicate records remained.

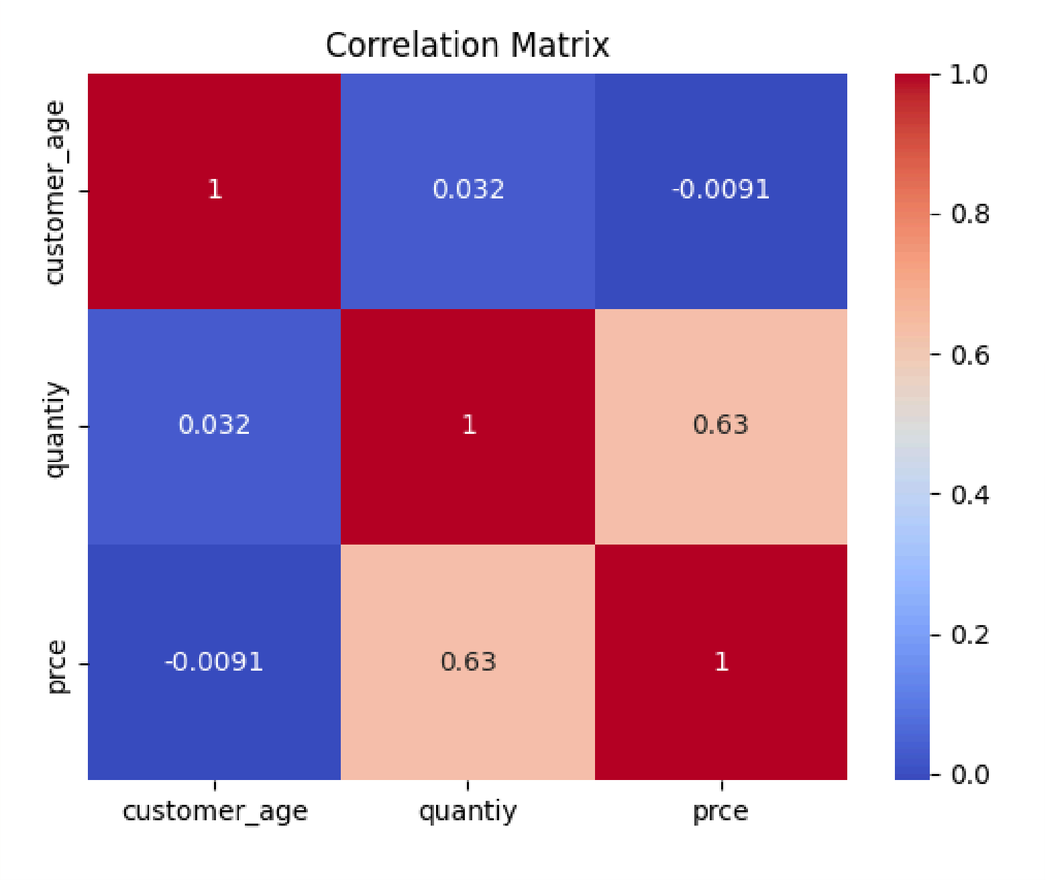
```
Jumlah data yang duplikat = 0
PS D:\tugas_datascience2_audry>
```

PROJECT 02

There is no significant relationship observed between customer age and either quantity or price. In contrast, a positive correlation between quantity and price (correlation coefficient of 0.63) is clearly evident from the upward trend in the data point distribution.



A visual representation of the relationships between variables



Numerical correlation value

THANK YOU

www.linkedin.com/in/audryprithae14

https://github.com/audrype14/dsf41-python-eda-highlight