	UNIVERSITAS TEKNOLOGI YOGYAKARTA **FAKULTAS SAINS DAN TEKNOLOGI** **PROGRAM STUDI SAINS DATA PROGRAM SARJANA** TUGAS BESAR MATA KULIAH MANAJEMEN DAN INFRASTRUKTUR DATA
NAMA	Lathif Ramadhan
NIM	5231811022
KELAS	23A

Tugas besar dikumpulkan pada minggu ke-14 tanggal 31 Desember 2024

SOAL

1. Buat atau cari data set yang sesuai dengan keahlian atau tema/topik penelitian anda.
2. Buat deskripsi yang menjelaskan tentang data set tersebut
3. Carilah kemungkinan-kemungkinan metode atau cara pengolahan yang dapat diterapkan ke data set tersebut.
4. Setelah anda mendapatkan metode dan cara pengolahan yang paling optimal, lakukan pengolahan dataset tersebut, jelaskan langkah-langkah nya beserta gambar kalau ada, visualisasikan hasil pengolahan data set anda.
5. Ceritakan maksud dari hasil pengolahan dataset anda sehingga mampu memberikan pengetahuan yang bermanfaat dari data yang diolah.

Tentang Dataset (Soal No. 1 & 2)

Judul Dataset : **Bank Marketing Campaign**

Link Dataset : https://www.kaggle.com/code/janiobachmann/bank-marketing-campaign-opening-a-term-deposit

Deskripsi Dataset Bank Marketing Campaign

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	age	job	marital	education	default	balance	housing	loan	contact	day	month	duration	campaign	pdays	previous	poutcome	deposit
2	59	admin.	married	secondary	no	2343	yes	no	unknown	5	may	1042	1	-1	0	unknown	yes
3	56	admin.	married	secondary	no	45	no	no	unknown	5	may	1467	1	-1	0	unknown	yes
4	41	technician	married	secondary	no	1270	yes	no	unknown	5	may	1389	1	-1	0	unknown	yes
5	55	services	married	secondary	no	2476	yes	no	unknown	5	may	579	1	-1	0	unknown	yes
6	54	admin.	married	tertiary	no	184	no	no	unknown	5	may	673	2	-1	0	unknown	yes
7	42	management	single	tertiary	no	0	yes	yes	unknown	5	may	562	2	-1	0	unknown	yes
8	56	management	married	tertiary	no	830	yes	yes	unknown	6	may	1201	1	-1	0	unknown	yes
9	60	retired	divorced	secondary	no	545	yes	no	unknown	6	may	1030	1	-1	0	unknown	yes
10	37	technician	married	secondary	no	1	yes	no	unknown	6	may	608	1	-1	0	unknown	yes
11	28	services	single	secondary	no	5090	yes	no	unknown	6	may	1297	3	-1	0	unknown	yes
12	38	admin.	single	secondary	no	100	yes	no	unknown	7	may	786	1	-1	0	unknown	yes
13	30	blue-collar	married	secondary	no	309	yes	no	unknown	7	may	1574	2	-1	0	unknown	yes
14	29	management	married	tertiary	no	199	yes	yes	unknown	7	may	1689	4	-1	0	unknown	yes
15	46	blue-collar	single	tertiary	no	460	yes	no	unknown	7	may	1102	2	-1	0	unknown	yes
16	31	technician	single	tertiary	no	703	yes	no	unknown	8	may	943	2	-1	0	unknown	yes
17	35	management	divorced	tertiary	no	3837	yes	no	unknown	8	may	1084	1	-1	0	unknown	yes
18	32	blue-collar	single	primary	no	611	yes	no	unknown	8	may	541	3	-1	0	unknown	yes

Dataset ini berfokus pada kampanye pemasaran yang dilakukan oleh sebuah bank untuk memprediksi apakah seorang pelanggan akan membuka deposito berjangka (term deposit) berdasarkan sejumlah variabel yang relevan. Kampanye ini melibatkan berbagai data demografis dan interaksi dengan pelanggan yang mencakup detail seperti usia, pekerjaan, status pernikahan, pendidikan, apakah pelanggan memiliki pinjaman rumah atau pribadi, serta jumlah kontak yang dilakukan selama kampanye.

Fitur Utama:

1. **Data Klien Bank**:
    - **Usia (age)**: Usia klien (numerik).
    - **Pekerjaan (job)**: Jenis pekerjaan klien, seperti 'admin.', 'blue-collar', 'entrepreneur', dll. (kategorikal).
    - **Status Pernikahan (marital)**: Status pernikahan klien ('divorced', 'married', 'single', dll.) (kategorikal).
    - **Pendidikan (education)**: Tingkat pendidikan klien ('primary', 'secondary', 'tertiary', dll.) (kategorikal).
    - **Default**: Apakah klien memiliki kredit macet ('no', 'yes', 'unknown') (kategorikal).
    - **Pinjaman Rumah (housing)**: Apakah klien memiliki pinjaman rumah ('no', 'yes', 'unknown') (kategorikal).

    - **Pinjaman Pribadi (loan)**: Apakah klien memiliki pinjaman pribadi ('no', 'yes', 'unknown') (kategorikal).
    - **Saldo (balance)**: Saldo akun klien (numerik).
2. **Terkait Kontak Terakhir dalam Kampanye**:
    - **Jenis Kontak (contact)**: Jenis komunikasi yang digunakan (misalnya, 'cellular', 'telephone') (kategorikal).
    - **Bulan (month)**: Bulan kontak terakhir dilakukan (kategorikal).
    - **Hari (day)**: Hari dalam minggu kontak terakhir dilakukan (kategorikal).
    - **Durasi (duration)**: Durasi kontak terakhir (numerik, detik).
3. **Fitur Lainnya**:
    - **Kampanye (campaign)**: Jumlah kontak yang dilakukan selama kampanye ini untuk klien tersebut (numerik).
    - **Pdays**: Jumlah hari sejak klien terakhir dihubungi dalam kampanye sebelumnya (numerik; 999 berarti klien belum pernah dihubungi sebelumnya).
    - **Sebelumnya (previous)**: Jumlah kontak yang dilakukan dalam kampanye sebelumnya (numerik).
    - **Hasil Kampanye Sebelumnya (poutcome)**: Hasil kampanye pemasaran sebelumnya ('failure', 'nonexistent', 'success') (kategorikal).

Variabel Target:

- **y**: Apakah klien membuka deposito berjangka ('yes', 'no').

Tujuan Analisis:

- Memprediksi apakah seorang pelanggan akan berlangganan deposito berjangka atau tidak berdasarkan variabel-variabel yang ada.
- Mengidentifikasi faktor-faktor yang paling berpengaruh terhadap keputusan pelanggan untuk berlangganan deposito berjangka.
- Memberikan insight kepada bank untuk mengoptimalkan strategi pemasaran mereka di masa mendatang.

Ringkasan

Dataset ini berasal dari UCI Machine Learning Repository dan berfokus pada analisis keberhasilan kampanye pemasaran bank dengan tujuan utama untuk memprediksi apakah pelanggan akan membuka deposito berdasarkan serangkaian variabel. Fitur-fitur dalam dataset mencakup demografi pelanggan, rincian kontak kampanye, serta hasil dari kampanye sebelumnya. Target output adalah keputusan pelanggan untuk membuka atau tidak membuka deposito berjangka. Data ini sangat berguna untuk aplikasi dalam pemasaran yang lebih terarah dan segmentasi pelanggan.

Kemungkinan Metode atau Cara Pengolahan untuk Dataset Bank Marketing (Soal No. 3)

1. **Pra-Pengolahan Data**: Sebelum melakukan analisis lebih lanjut, penting untuk membersihkan dataset terlebih dahulu. Hal ini termasuk menangani data yang hilang (misalnya, mengganti nilai yang hilang dengan rata-rata atau median) dan mengkonversi variabel kategorikal menjadi numerik menggunakan teknik seperti one-hot encoding.
2. **Eksplorasi Data (EDA)**: Dengan tujuan untuk memahami lebih dalam tentang dataset, kita bisa menggunakan teknik visualisasi seperti histogram, boxplot, atau scatter plot untuk melihat distribusi usia, saldo, dan durasi kontak. Ini akan membantu menemukan pola-pola penting yang dapat memengaruhi keputusan pelanggan dalam membuka deposito.
3. **Feature Engineering**:
    - **Encoding Kategorikal**: Menggunakan **one-hot encoding** untuk variabel seperti job, marital, dan education.
    - **Pembuatan Fitur Interaksi**: Menggabungkan variabel seperti age dan job untuk melihat pola dalam kelompok tertentu.
    - **Binning**: Mengelompokkan variabel numerik seperti age dan balance dalam interval untuk memudahkan analisis.
    - **Fitur Baru**: Membuat fitur baru seperti total jumlah interaksi (campaign + previous).
4. **Analisis Korelasi**: Menggunakan korelasi untuk melihat hubungan antar variabel yang ada, misalnya, apakah ada hubungan yang kuat antara durasi kontak dengan keputusan untuk membuka deposito. Ini penting karena variabel seperti durasi sering kali sangat berkaitan dengan hasil kampanye.
5. **Segmentasi Pelanggan**: Teknik seperti klastering (misalnya K-Means) dapat digunakan untuk mengelompokkan pelanggan berdasarkan karakteristik yang serupa, seperti usia, pekerjaan, dan status pernikahan. Dengan segmentasi ini, bank dapat menargetkan pelanggan dengan lebih tepat, meningkatkan efektivitas kampanye.
6. **Model Prediksi**: Setelah pemahaman yang cukup terhadap data, kita bisa mencoba membangun model prediksi untuk memprediksi kemungkinan pelanggan membuka deposito. Beberapa model yang bisa diterapkan termasuk:
    - **Logistic Regression**: Untuk memprediksi variabel biner (ya/tidak) seperti apakah pelanggan membuka deposito.
    - **Decision Trees**: Untuk membuat keputusan berdasarkan serangkaian pertanyaan terkait variabel input.
    - **Random Forests**: Menggunakan banyak pohon keputusan untuk meningkatkan akurasi model.
7. **Evaluasi Model**: Setelah model dibangun, kita perlu mengevaluasi kinerjanya menggunakan metrik seperti akurasi, precision, recall, dan confusion matrix. Ini membantu dalam menilai seberapa baik model dalam memprediksi keputusan pelanggan yang tepat.

Dengan pendekatan yang sistematis ini, kita dapat memperoleh wawasan yang berguna untuk meningkatkan kampanye pemasaran dan pengambilan keputusan yang lebih baik di masa mendatang.

Pengolahan Dataset dan Cerita Hasil Pengolahan Dataset Menggunakan Python (.ipynb) di Google Colab (Soal No. 4 & 5)

Judul Dataset : Bank Marketing Campaign

Link Dataset : https://www.kaggle.com/code/janiobachmann/bank-marketing-campaign-opening-a-term-deposit

Import Library

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
from google.colab import drive

import matplotlib.pyplot as plt
import seaborn as sns
import calendar

from sklearn.cluster import KMeans
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import OneHotEncoder, StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score, classification_report,
confusion_matrix
from sklearn.metrics import ConfusionMatrixDisplay
```

##Load Dataset

Mount Google Drive

```
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

Navigasi File

```
file_path ='/content/drive/My Drive/Management and Infrastruktur
Data/bank.csv'
data = pd.read_csv(file_path)
data.head()
```


```
{"summary":"{\n  \"name\": \"data\",\n  \"rows\": 11162,\n  \"fields\": [\n
{\n      \"column\": \"age\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 11,\n        \"min\": 18,\n        \"max\":
95,\n        \"num_unique_values\": 76,\n        \"samples\": [\n
54,\n          47,\n          30\n        ],\n        \"semantic_type\":
```

```
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"job\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 12,\n        \"samples\": [\n
\"self-employed\",\n          \"unknown\",\n          \"admin.\"\n
],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    },\n    {\n      \"column\": \"marital\",\n      \"properties\": {\n
\"dtype\": \"category\",\n        \"num_unique_values\": 3,\n
\"samples\": [\n          \"married\",\n          \"single\",\n
\"divorced\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"education\",\n      \"properties\": {\n        \"dtype\": \"category\",\n
\"num_unique_values\": 4,\n        \"samples\": [\n          \"tertiary\",\n
\"unknown\",\n          \"secondary\"\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"default\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 2,\n        \"samples\": [\n
\"yes\",\n          \"no\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"balance\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 3225,\n        \"min\": -6847,\n        \"max\": 81204,\n
\"num_unique_values\": 3805,\n        \"samples\": [\n          3026,\n
1792\n        ],\n        \"semantic_type\": \"\",\n        \"description\":
\"\"\n      }\n    },\n    {\n      \"column\": \"housing\",\n
\"properties\": {\n        \"dtype\": \"category\",\n
\"num_unique_values\": 2,\n        \"samples\": [\n          \"no\",\n
\"yes\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"loan\",\n
\"properties\": {\n        \"dtype\": \"category\",\n
\"num_unique_values\": 2,\n        \"samples\": [\n          \"yes\",\n
\"no\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"contact\",\n      \"properties\": {\n        \"dtype\": \"category\",\n
\"num_unique_values\": 3,\n        \"samples\": [\n          \"unknown\",\n
\"cellular\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"day\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 8,\n
\"min\": 1,\n        \"max\": 31,\n        \"num_unique_values\": 31,\n
\"samples\": [\n          10,\n          27\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\n    },\n
{\n      \"column\": \"month\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 12,\n        \"samples\": [\n
\"apr\",\n          \"mar\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"duration\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 347,\n        \"min\": 2,\n        \"max\": 3881,\n
\"num_unique_values\": 1428,\n        \"samples\": [\n          597,\n
346\n        ],\n        \"semantic_type\": \"\",\n        \"description\":
\"\"\n      }\n    },\n    {\n      \"column\": \"campaign\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 2,\n
\"min\": 1,\n        \"max\": 63,\n        \"num_unique_values\": 36,\n
```

\"samples\": [\n   31,\n   7\n   ],\n
\"semantic_type\": \"\",\n   \"description\": \"\"\n   }\n   },\n
{\n   \"column\": \"pdays\",\n   \"properties\": {\n   \"dtype\":
\"number\",\n   \"std\": 108,\n   \"min\": -1,\n   \"max\":
854,\n   \"num_unique_values\": 472,\n   \"samples\": [\n
294,\n   148\n   ],\n   \"semantic_type\": \"\",\n
\"description\": \"\"\n   }\n   },\n   {\n   \"column\":
\"previous\",\n   \"properties\": {\n   \"dtype\": \"number\",\n
\"std\": 2,\n   \"min\": 0,\n   \"max\": 58,\n
\"num_unique_values\": 34,\n   \"samples\": [\n   30,\n
14\n   ],\n   \"semantic_type\": \"\",\n   \"description\":
\"\"\n   }\n   },\n   {\n   \"column\": \"poutcome\",\n
\"properties\": {\n   \"dtype\": \"category\",\n
\"num_unique_values\": 4,\n   \"samples\": [\n   \"other\",\n
\"success\"\n   ],\n   \"semantic_type\": \"\",\n
\"description\": \"\"\n   }\n   },\n   {\n   \"column\":
\"deposit\",\n   \"properties\": {\n   \"dtype\": \"category\",\n
\"num_unique_values\": 2,\n   \"samples\": [\n   \"no\",\n
\"yes\"\n   ],\n   \"semantic_type\": \"\",\n
\"description\": \"\"\n   }\n   }\n
]\n}","type":"dataframe","variable_name":"data"}

1. Pra-Pengolahan Data

Mencari Informasi Dataset

Secara sederhana, `data.info()` memberikan kita gambaran umum tentang dataset yang sedang kita gunakan. Ibaratnya seperti membaca *summary* atau ringkasan dari sebuah buku sebelum kita membacanya secara keseluruhan.

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11162 entries, 0 to 11161
Data columns (total 17 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   age        11162 non-null  int64
 1   job        11162 non-null  object
 2   marital    11162 non-null  object
 3   education  11162 non-null  object
 4   default    11162 non-null  object
 5   balance    11162 non-null  int64
 6   housing    11162 non-null  object
 7   loan       11162 non-null  object
 8   contact    11162 non-null  object
 9   day        11162 non-null  int64
 10  month      11162 non-null  object
 11  duration   11162 non-null  int64
 12  campaign   11162 non-null  int64
```

```
 13  pdays      11162 non-null  int64
 14  previous   11162 non-null  int64
 15  poutcome   11162 non-null  object
 16  deposit    11162 non-null  object
dtypes: int64(7), object(10)
memory usage: 1.4+ MB
```

Nah, dari output `data.info()` diatas, kita bisa mendapatkan informasi penting berikut:

1. **Jumlah Baris dan Kolom**: Dataset ini memiliki **11.162 baris data**, yang artinya ada 11.162 observasi atau individu yang tercatat. Selain itu, dataset ini memiliki **17 kolom**, yang berarti ada 17 variabel atau atribut yang mendeskripsikan setiap observasi.

1. **Nama dan Tipe Data Kolom**: Tabel di output `data.info()` menunjukkan nama setiap kolom beserta tipe datanya.

- Ada **7 kolom** dengan tipe data `int64`, yang berarti kolom-kolom ini berisi bilangan bulat (integer). Contohnya: `age`, `balance`, `day`, `duration`, `campaign`, `pdays`, dan `previous`.
- Ada **10 kolom** dengan tipe data `object`, yang biasanya menunjukkan data kategorikal atau teks. Contohnya: `job`, `marital`, `education`, `default`, `housing`, `loan`, `contact`, `month`, `poutcome`, dan `deposit`.

1. **Jumlah Data yang Tidak Kosong (Non-Null)**: Pada bagian Non-Null Count, kita bisa melihat jumlah data yang tidak kosong (tidak hilang/ *missing*) pada setiap kolom. Untungnya, semua kolom pada dataset ini memiliki 11.162 data non-null, yang berarti tidak ada *missing values* yang perlu kita khawatirkan.

1. **Penggunaan Memori**: Di bagian bawah output, tertulis `memory usage: 1.4+ MB`. Ini menunjukkan seberapa besar memori yang digunakan oleh dataset ini di Colab. Informasi ini berguna untuk memperkirakan kebutuhan memori saat kita memproses dataset yang lebih besar.

Mengecek nilai yang hilang (*missing value*)

Secara sederhana, kode `data.isnull().sum()` digunakan untuk **mengecek apakah ada data yang hilang (missing values)** dalam dataset kita.

Cara kerjanya begini:

1. `data.isnull()`: Bagian ini akan memeriksa setiap sel di dalam dataset. Jika sel tersebut kosong atau tidak berisi data, maka akan diberi tanda `True`, dan jika berisi data, akan diberi tanda `False`.
1. `.sum()`: Kemudian, fungsi `sum()` akan menjumlahkan semua tanda `True` (yang menandakan *missing values*) pada setiap kolom.

```
data.isnull().sum()
```

```
age          0
job          0
```

```
marital      0
education    0
default      0
balance      0
housing      0
loan         0
contact      0
day          0
month        0
duration     0
campaign     0
pdays        0
previous     0
poutcome     0
deposit      0
dtype: int64
```

Kita bisa melihat bahwa **semua kolom memiliki nilai 0**. Artinya, **tidak ada data yang hilang (missing values)** di dalam dataset kamu. Semua kolom terisi penuh dengan data.

Ini adalah kabar baik! Karena kita tidak perlu melakukan penanganan khusus untuk *missing values*, seperti *imputation* atau menghapus baris/kolom yang memiliki data kosong. Kita bisa langsung melanjutkan ke tahap analisis data selanjutnya.

Transformasi Data

Beberapa algoritma machine learning bekerja lebih baik jika data memiliki distribusi normal. Kita bisa mempertimbangkan transformasi data seperti logaritmik atau Box-Cox.

```
data['balance_log'] = np.log1p(data['balance'])
```

```
/usr/local/lib/python3.10/dist-packages/pandas/core/arraylike.py:399:
RuntimeWarning: divide by zero encountered in log1p
  result = getattr(ufunc, method)(*inputs, **kwargs)
/usr/local/lib/python3.10/dist-packages/pandas/core/arraylike.py:399:
RuntimeWarning: invalid value encountered in log1p
  result = getattr(ufunc, method)(*inputs, **kwargs)
```

Alasan: Transformasi data dapat membantu meningkatkan linearitas dan normalitas data, yang dapat meningkatkan performa beberapa model.

2. Eksplorasi Data (EDA)

Statistik deskriptif

Secara sederhana, fungsi `data.describe()` memberikan **statistik deskriptif** dari data numerik dalam dataset. Statistik ini membantu kita memahami karakteristik dasar data, seperti:

- **count**: Jumlah data non-null pada setiap kolom.
- **mean**: Rata-rata nilai pada setiap kolom.

- **std**: Standar deviasi, menunjukkan sebaran data di sekitar rata-rata.
- **min**: Nilai minimum pada setiap kolom.
- **25%**: Kuartil pertama (Q1), nilai di mana 25% data berada di bawahnya.
- **50%**: Kuartil kedua (Q2) atau median, nilai di mana 50% data berada di bawahnya.
- **75%**: Kuartil ketiga (Q3), nilai di mana 75% data berada di bawahnya.
- **max**: Nilai maksimum pada setiap kolom.

Dengan menjalankan `data.describe()` pada dataset ini, akan muncul tabel yang menampilkan statistik-statistik tersebut untuk setiap kolom numerik.

Statistik deskriptif ini sangat berguna dalam EDA karena:

- **Identifikasi Outlier**: Dengan melihat nilai `min`, `max`, dan `kuartil`, kita bisa mengidentifikasi potensi outlier atau nilai ekstrem yang mungkin perlu ditangani lebih lanjut.
- **Memahami Distribusi Data**: Statistik seperti `mean`, `std`, dan kuartil memberikan gambaran tentang distribusi data, apakah terdistribusi normal, miring, atau memiliki pola tertentu.
- **Perbandingan Antar Kolom**: Dengan membandingkan statistik deskriptif antar kolom, kita bisa melihat perbedaan karakteristik data pada setiap variabel.

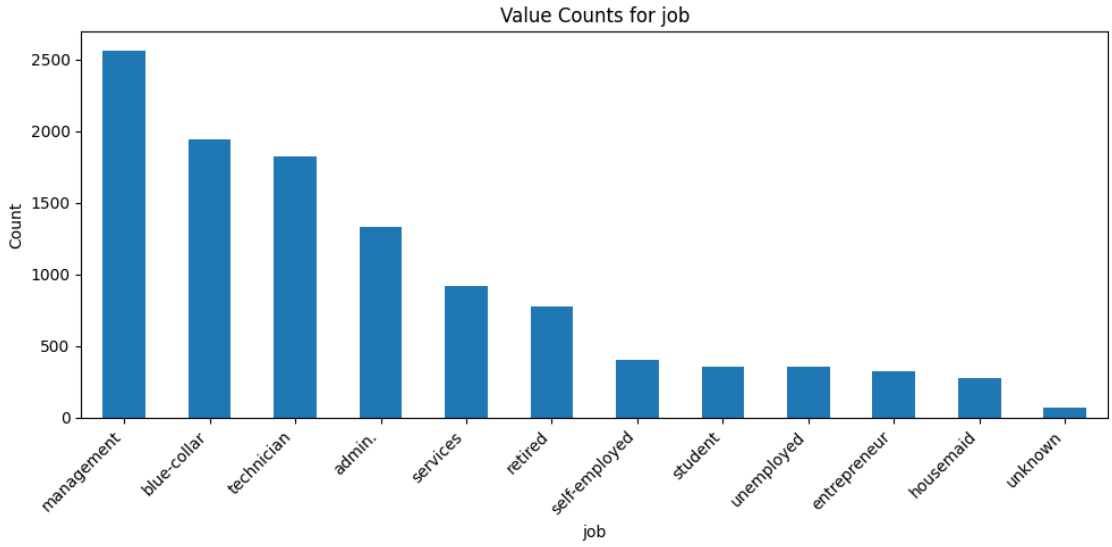
Jadi, dengan menjalankan `data.describe()` dan memahami outputnya, kita akan mendapatkan pemahaman awal yang penting tentang data numerik dalam dataset ini sebelum melakukan analisis lebih lanjut.

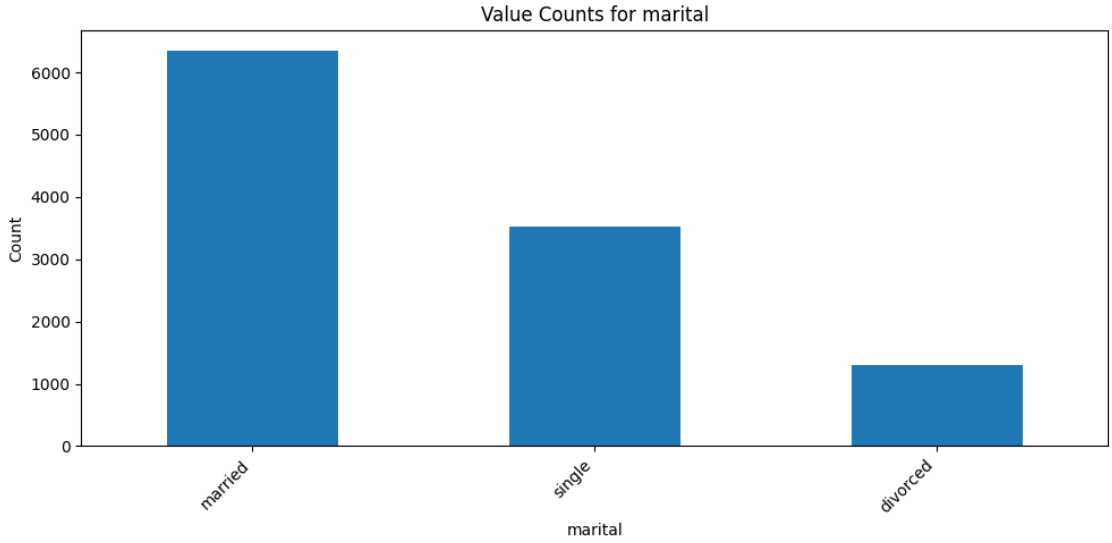
```
data.describe()
```

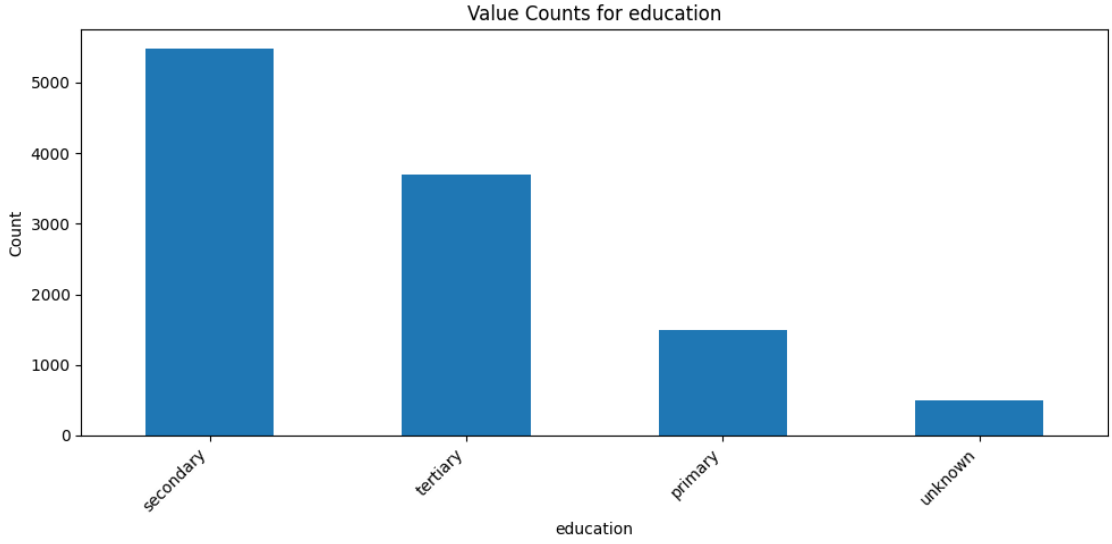

```
{"repr_error":"Out of range float values are not JSON compliant: -
inf","type":"dataframe"}
```

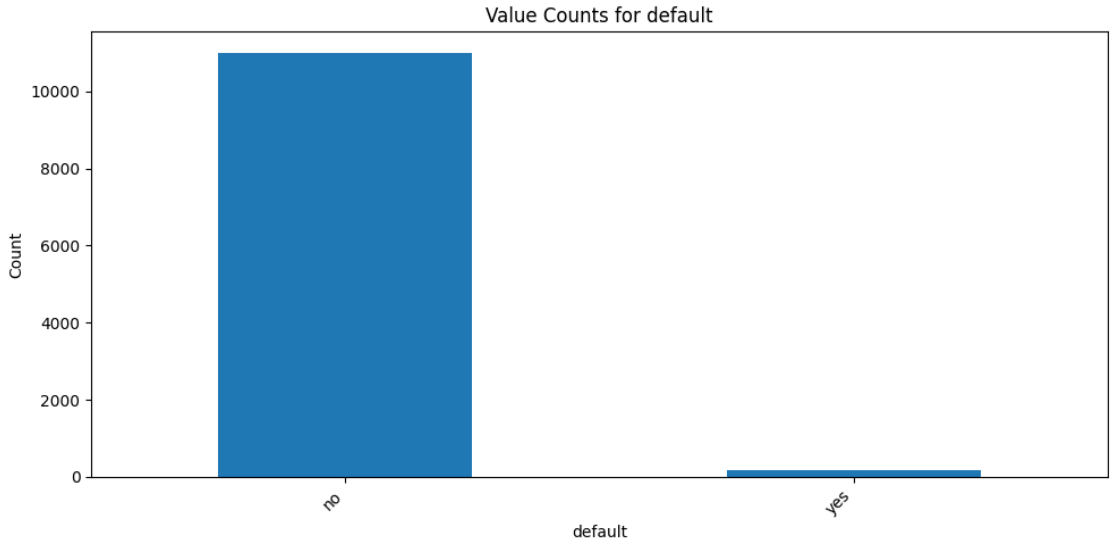

Bar Plot untuk Distribusi Data Kategorikal

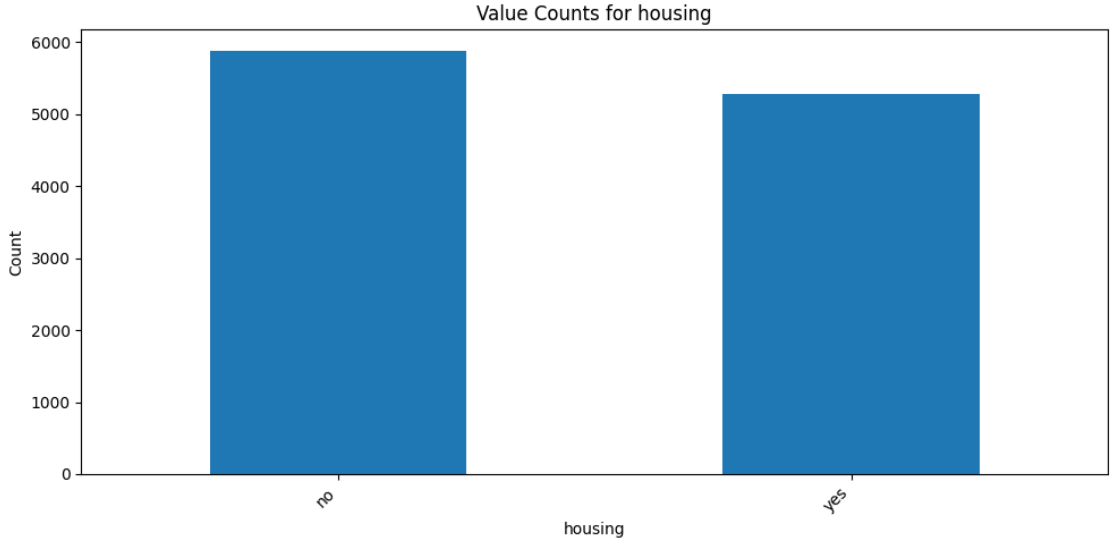
```
categorical_cols = [col for col in data.columns if data[col].dtype ==
'object']
for col in categorical_cols:
  plt.figure(figsize=(10,5))  # Mengatur ukuran figure
  data[col].value_counts().plot(kind='bar') # Membuat bar plot
  plt.title(f'Value Counts for {col}') # Memberi judul plot
  plt.xlabel(col) # Memberi label sumbu x
  plt.ylabel('Count') # Memberi label sumbu y
  plt.xticks(rotation=45, ha='right') # Merotasi label sumbu x agar mudah
dibaca
  plt.tight_layout() # Mengatur layout agar tidak tumpang tindih
  plt.show() # Menampilkan plot
```

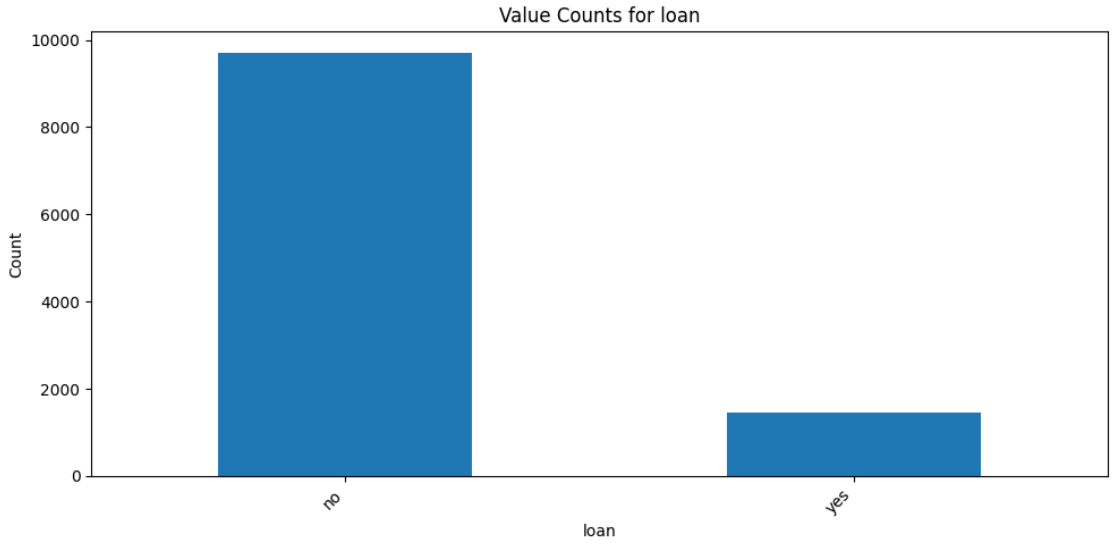

Value Counts for job
Count
2500
2000
1500
1000
500
0
management
blue-collar
technician
admin.
services
retired
self-employed
student
unemployed
entrepreneur
housemaid
unknown
job

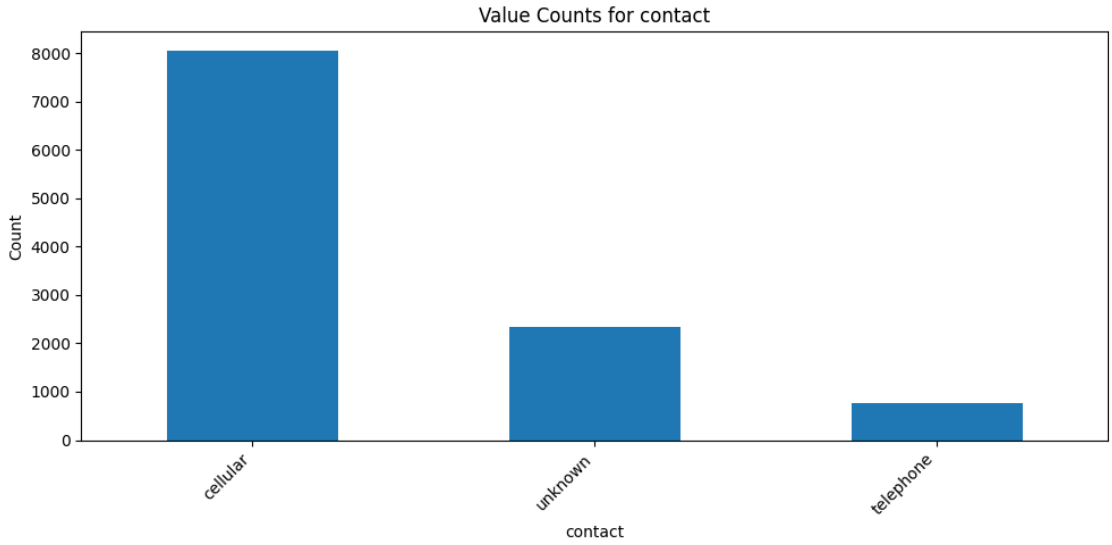

Value Counts for marital
Count
6000
5000
4000
3000
2000
1000
0
married
single
divorced
marital

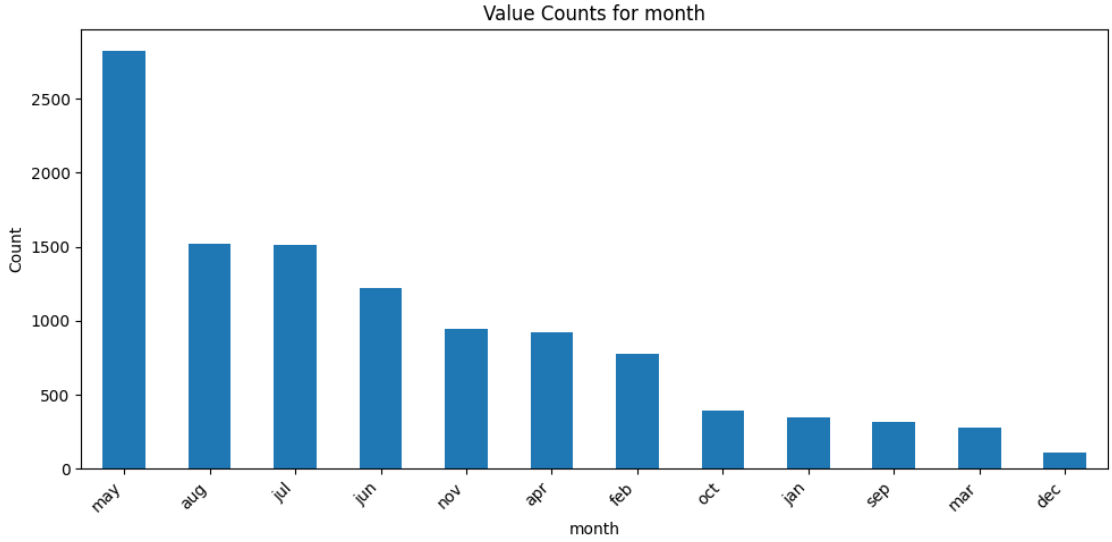

Value Counts for education
Count
5000
4000
3000
2000
1000
0
secondary
tertiary
primary
unknown
education

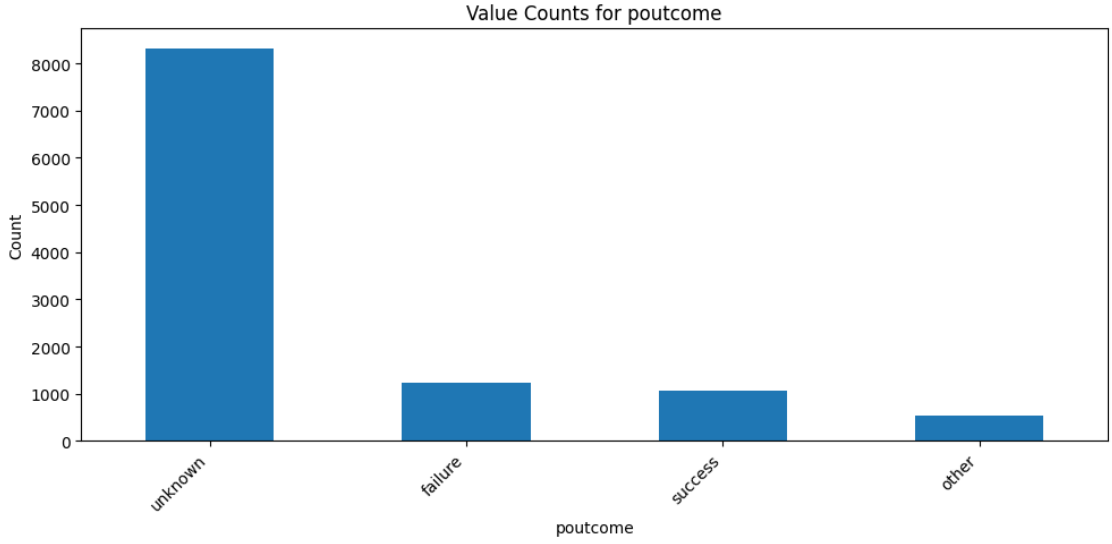

Value Counts for default
Count
10000
8000
6000
4000
2000
0
no
yes
default

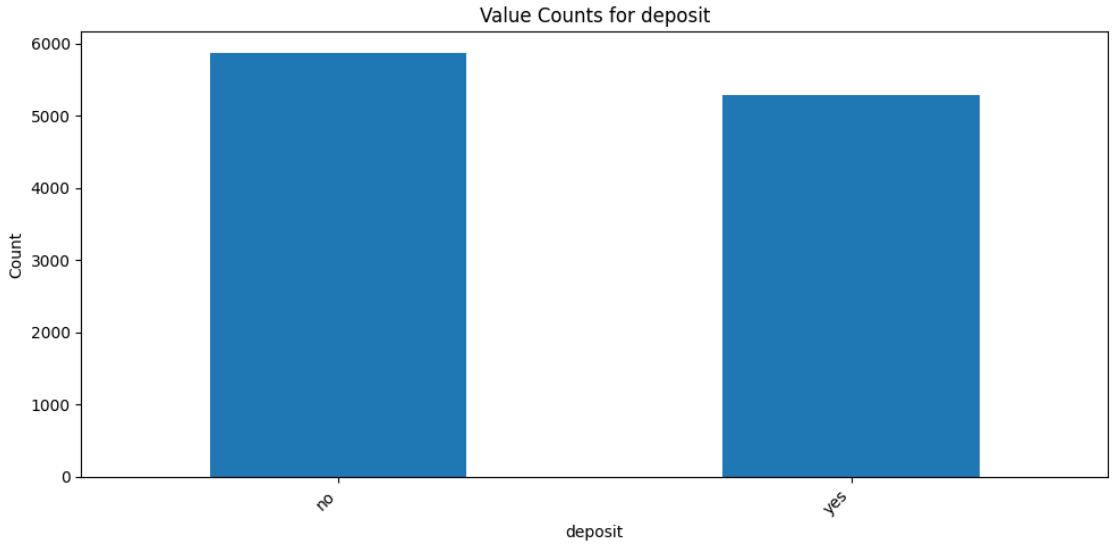

Value Counts for housing
Count
6000
5000
4000
3000
2000
1000
0
no
yes
housing


Value Counts for loan
Count
10000
8000
6000
4000
2000
0
no
yes
loan


Value Counts for contact
Count
0
1000
2000
3000
4000
5000
6000
7000
8000
cellular
unknown
telephone
contact

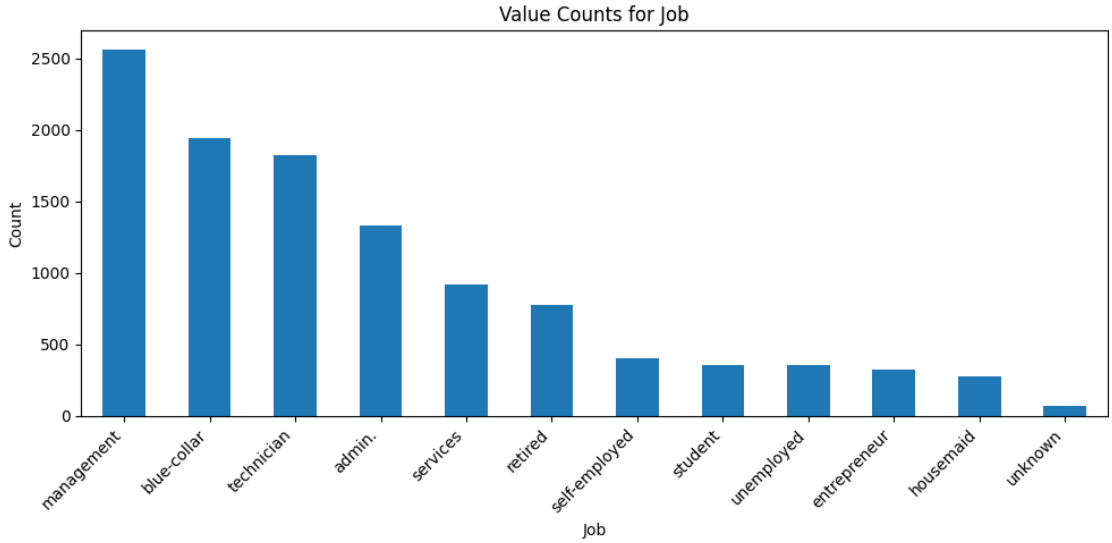

Value Counts for month
Count
0
500
1000
1500
2000
2500
may
aug
jul
jun
nov
apr
feb
oct
jan
sep
mar
dec
month


Value Counts for poutcome
Count
0
1000
2000
3000
4000
5000
6000
7000
8000
unknown
failure
success
other
poutcome



1. Kolom `Job`:

```
# Kolom 'job'
plt.figure(figsize=(10, 5))
data['job'].value_counts().plot(kind='bar')
plt.title('Value Counts for Job')
plt.xlabel('Job')
plt.ylabel('Count')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```

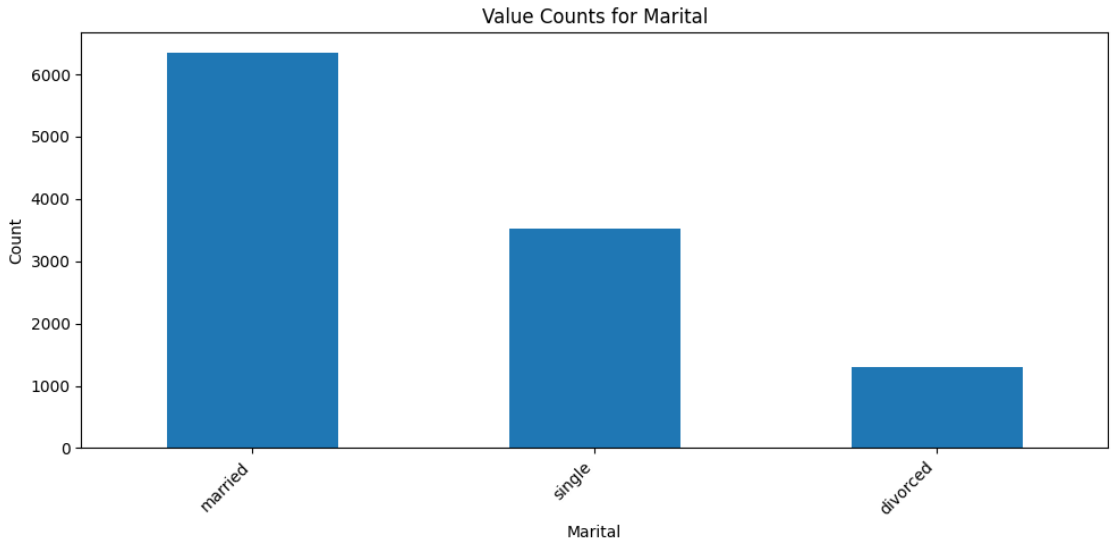


- Nasabah dengan pekerjaan "`management`" cenderung lebih banyak membuka deposito. Ini bisa mengindikasikan bahwa kelompok ini memiliki daya beli yang lebih tinggi atau lebih sadar akan pentingnya menabung.

- Nasabah dengan pekerjaan "`blue-collar`" dan "`technician`" juga cukup banyak yang membuka deposito. Ini menunjukkan bahwa produk deposito bank tersebut menarik minat dari berbagai kalangan, tidak hanya kalangan profesional.
- Nasabah dengan pekerjaan "`student`", "`unemployed`", dan "`housemaid`" cenderung lebih sedikit yang membuka deposito. Ini mungkin karena kelompok ini memiliki keterbatasan finansial atau prioritas keuangan yang berbeda.

2. Kolom `Job`:

```
# Kolom 'marital'
plt.figure(figsize=(10, 5))
data['marital'].value_counts().plot(kind='bar')
plt.title('Value Counts for Marital')
plt.xlabel('Marital')
plt.ylabel('Count')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```



- **Sebagian besar nasabah yang melakukan deposito sudah menikah**. Ini menunjukkan bahwa orang yang sudah menikah cenderung lebih tertarik atau mampu untuk membuka deposito. Kemungkinan, mereka memiliki stabilitas finansial yang lebih baik atau memiliki tujuan finansial yang lebih besar, seperti menabung untuk masa depan keluarga.
- **Nasabah yang masih single juga cukup banyak yang melakukan deposito**. Ini menandakan bahwa kelompok ini juga memiliki kesadaran akan pentingnya menabung, meskipun mungkin tujuan menabung mereka berbeda dengan yang sudah menikah.
- **Nasabah yang bercerai merupakan kelompok terkecil yang melakukan deposito**. Hal ini bisa disebabkan oleh beberapa faktor, seperti:

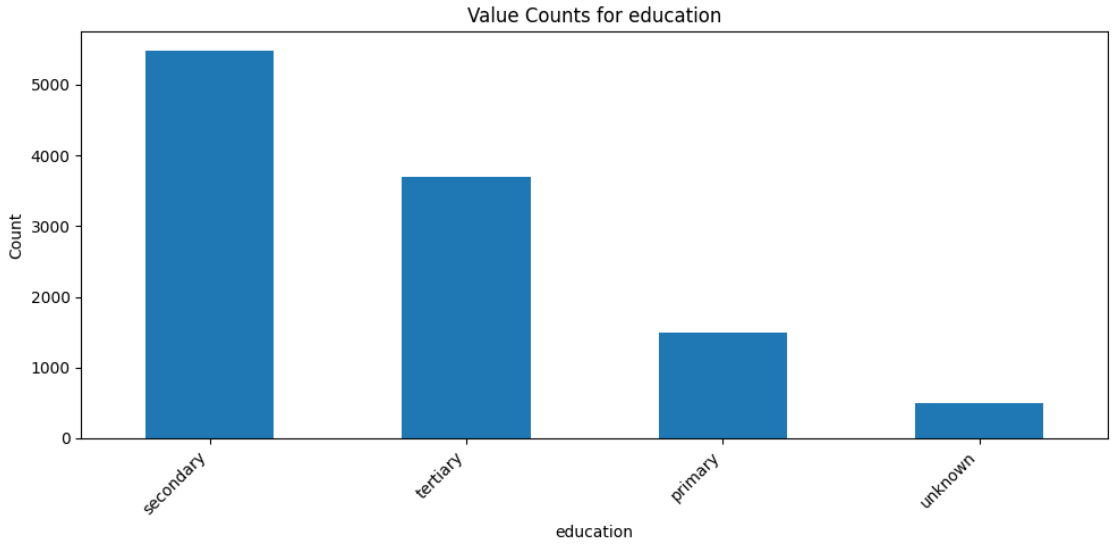
- **Kondisi finansial**: Nasabah yang baru bercerai mungkin sedang mengalami ketidakstabilan finansial dan belum siap untuk membuka deposito.
- **Prioritas**: Mereka mungkin memiliki prioritas finansial yang berbeda, seperti melunasi utang atau membangun kembali kehidupan mereka.
- **Persepsi**: Mereka mungkin memiliki persepsi yang berbeda tentang produk deposito atau merasa tidak yakin dengan masa depan finansial mereka.

Apa artinya bagi bank?

- **Fokus pada nasabah yang sudah menikah**: Bank bisa lebih fokus pada segmentasi pasar ini dengan menawarkan produk dan layanan yang relevan dengan kebutuhan keluarga, seperti deposito berjangka dengan bunga kompetitif atau produk investasi yang aman.
- **Jangan mengabaikan nasabah single**: Meskipun jumlahnya lebih sedikit, nasabah single juga memiliki potensi besar. Bank bisa menawarkan produk yang lebih fleksibel dan menarik bagi mereka, seperti deposito dengan jangka waktu pendek atau produk investasi yang mudah diakses.
- **Perhatikan nasabah yang bercerai**: Meskipun jumlahnya paling sedikit, ada potensi untuk meningkatkan jumlah nasabah dari kelompok ini. Bank bisa menawarkan program edukasi finansial atau produk yang dirancang khusus untuk membantu mereka membangun kembali stabilitas finansial.

3. Kolom `education`:

```
# Kolom 'education'
plt.figure(figsize=(10, 5))
data['education'].value_counts().plot(kind='bar')
plt.title('Value Counts for education')
plt.xlabel('education')
plt.ylabel('Count')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```



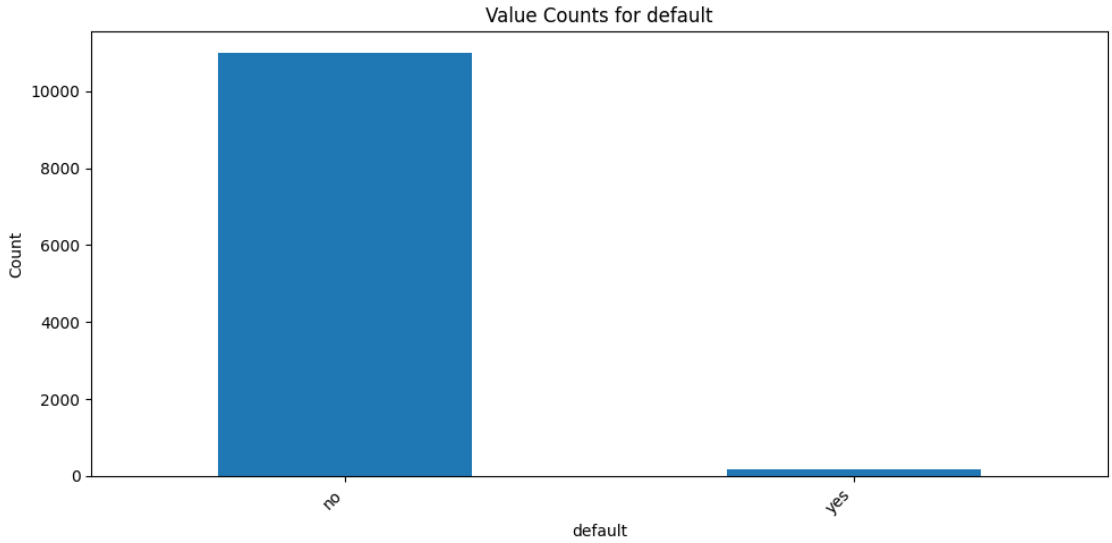
- **Sebagian besar nasabah yang melakukan deposito memiliki pendidikan terakhir tingkat menengah (`secondary`)**. Ini menunjukkan bahwa kelompok dengan pendidikan menengah memiliki minat yang lebih tinggi untuk menabung atau memiliki kemampuan finansial yang lebih baik.
- **Nasabah dengan pendidikan tinggi (`tertiary`) juga cukup banyak yang melakukan deposito**. Ini menunjukkan bahwa kelompok ini memiliki kesadaran akan pentingnya perencanaan keuangan jangka panjang.
- **Nasabah dengan pendidikan dasar (`primary`) dan yang tidak diketahui tingkat pendidikannya (`unknown`) merupakan kelompok yang lebih kecil yang melakukan deposito**. Ini bisa disebabkan oleh beberapa faktor, seperti:
  - **Kondisi finansial**: Nasabah dengan pendidikan dasar mungkin memiliki keterbatasan finansial atau lebih memprioritaskan kebutuhan sehari-hari.
  - **Akses informasi**: Nasabah dengan pendidikan yang lebih rendah mungkin memiliki akses informasi yang lebih terbatas tentang produk keuangan.

Apa artinya bagi bank?

- **Fokus pada nasabah dengan pendidikan menengah**: Bank bisa lebih fokus pada segmentasi pasar ini dengan menawarkan produk dan layanan yang sesuai dengan kebutuhan mereka, seperti produk investasi yang mudah dipahami atau layanan konsultasi keuangan.
- **Jangan mengabaikan nasabah dengan pendidikan tinggi**: Kelompok ini memiliki potensi untuk menjadi nasabah dengan nilai tambah yang tinggi. Bank bisa menawarkan produk yang lebih kompleks dan layanan yang lebih personal.
- **Perhatikan nasabah dengan pendidikan dasar dan yang tidak diketahui**: Bank bisa memberikan edukasi finansial yang lebih intensif kepada kelompok ini untuk meningkatkan kesadaran mereka tentang pentingnya menabung dan berinvestasi.

4. Kolom `default`:

```
# Kolom 'default'
plt.figure(figsize=(10, 5))
data['default'].value_counts().plot(kind='bar')
plt.title('Value Counts for default')
plt.xlabel('default')
plt.ylabel('Count')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```



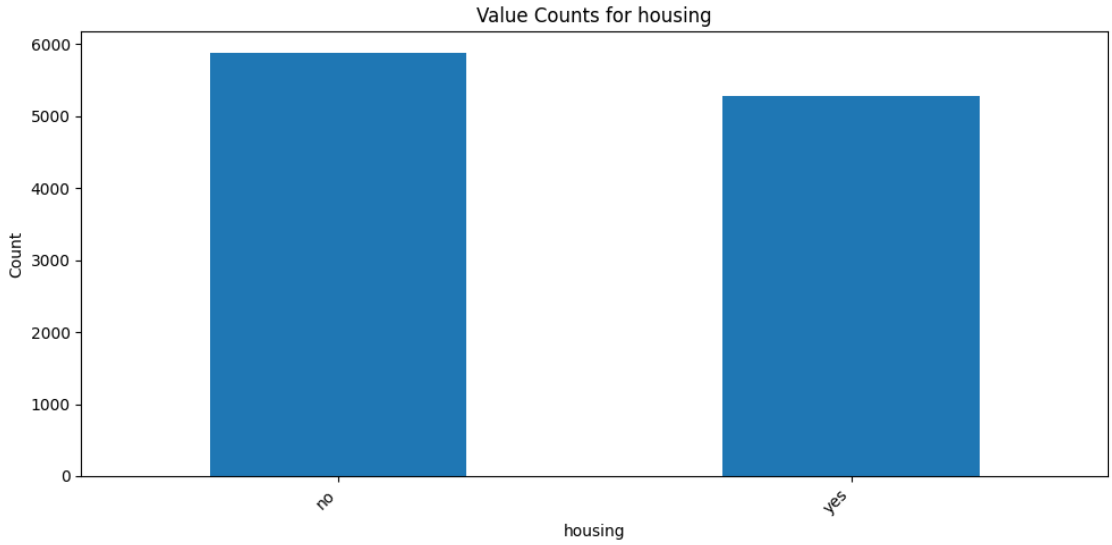
- **Sebagian besar nasabah tidak mengalami default**. Ini menunjukkan bahwa secara umum, nasabah yang ada dalam dataset ini memiliki rekam jejak pembayaran yang baik.
- **Hanya sebagian kecil nasabah yang mengalami default**. Ini berarti bahwa tingkat ketidakmampuan membayar atau gagal bayar di antara nasabah ini relatif rendah.

Apa artinya bagi bank?

- **Portofolio kredit yang sehat**: Jumlah nasabah yang mengalami default yang rendah menunjukkan bahwa bank memiliki portofolio kredit yang relatif sehat. Ini berarti bahwa bank memiliki risiko kredit yang lebih rendah.
- **Potensi untuk meningkatkan penyaluran kredit**: Dengan tingkat default yang rendah, bank dapat mempertimbangkan untuk meningkatkan penyaluran kredit kepada nasabah baru, terutama jika mereka memenuhi kriteria kredit yang baik.
- **Perlu adanya pemantauan terhadap nasabah yang berpotensi mengalami default**: Meskipun jumlahnya kecil, nasabah yang mengalami default tetap perlu diperhatikan. Bank perlu melakukan analisis lebih lanjut untuk mengidentifikasi faktor-faktor yang menyebabkan terjadinya default dan mengambil tindakan preventif untuk mencegah terjadinya default pada nasabah lainnya.

5. Kolom `housing`:

```
# Kolom 'housing'
plt.figure(figsize=(10, 5))
data['housing'].value_counts().plot(kind='bar')
plt.title('Value Counts for housing')
plt.xlabel('housing')
plt.ylabel('Count')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```



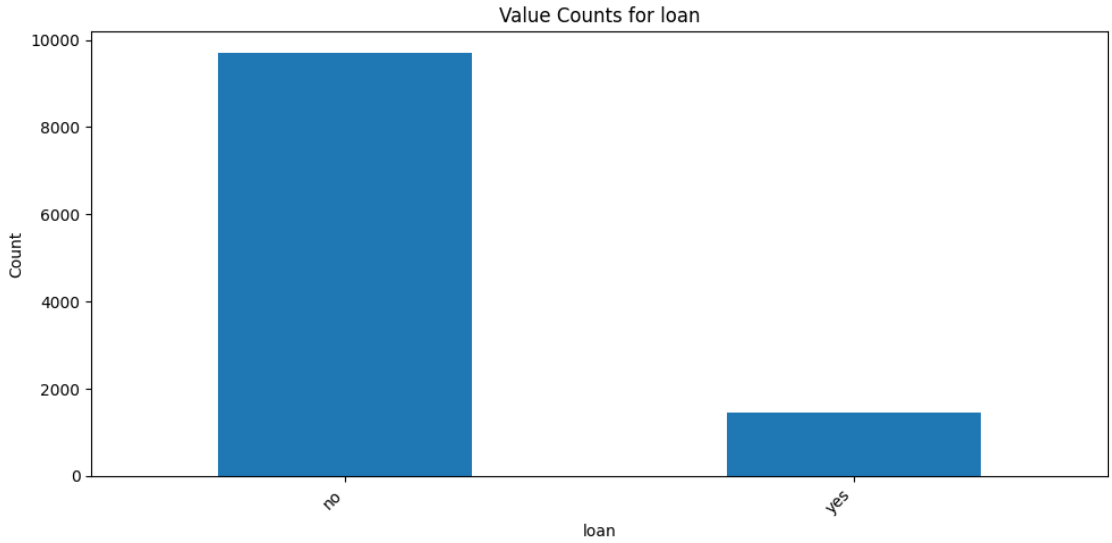
- **Sebagian besar nasabah tidak memiliki pinjaman perumahan**. Ini menunjukkan bahwa banyak nasabah yang belum memiliki kewajiban utang untuk perumahan.
- **Jumlah nasabah yang memiliki pinjaman perumahan juga cukup signifikan**. Ini menunjukkan bahwa ada cukup banyak nasabah yang memiliki beban utang untuk perumahan.

Apa artinya bagi bank?

- **Potensi pasar yang besar**: Nasabah yang tidak memiliki pinjaman perumahan bisa menjadi target potensial untuk produk kredit perumahan. Bank bisa menawarkan berbagai jenis produk kredit perumahan dengan suku bunga yang menarik dan persyaratan yang mudah.
- **Perlu hati-hati dalam memberikan kredit perumahan**: Nasabah yang sudah memiliki pinjaman perumahan perlu diperhatikan dengan baik. Bank perlu melakukan analisis yang cermat terhadap kemampuan pembayaran mereka sebelum memberikan kredit tambahan.
- **Peluang untuk bundling produk**: Bank bisa menawarkan produk bundling, misalnya gabungan antara deposito dan kredit perumahan, untuk memberikan nilai tambah kepada nasabah.

6. Kolom `loan`:

```
# Kolom 'loan'
plt.figure(figsize=(10, 5))
data['loan'].value_counts().plot(kind='bar')
plt.title('Value Counts for loan')
plt.xlabel('loan')
plt.ylabel('Count')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```



- **Sebagian besar nasabah tidak memiliki pinjaman pribadi**. Artinya, banyak nasabah yang tidak punya utang pribadi selain mungkin utang rumah.
- **Hanya sebagian kecil nasabah yang memiliki pinjaman pribadi**. Ini berarti tidak banyak nasabah yang punya utang pribadi tambahan.

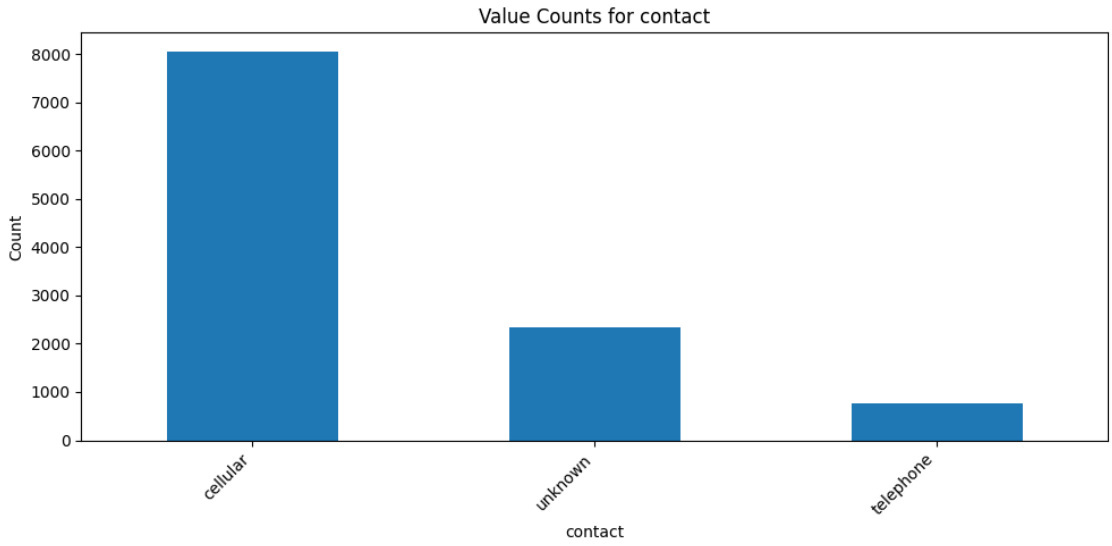
Apa artinya bagi bank?

- **Peluang besar untuk tawarkan pinjaman**: Karena banyak nasabah yang tidak punya utang, bank bisa coba tawarkan produk pinjaman pribadi dengan suku bunga menarik dan syarat yang mudah.
- **Perlu hati-hati saat memberikan pinjaman**: Bagi nasabah yang sudah punya pinjaman, bank perlu teliti sebelum memberi pinjaman tambahan untuk memastikan mereka mampu membayar.
- **Peluang untuk gabungkan produk**: Bank bisa tawarkan paket produk, misalnya gabungkan deposito dengan pinjaman pribadi, untuk menarik minat nasabah.

7. Kolom `contact`:

```
# Kolom 'contact'
plt.figure(figsize=(10, 5))
data['contact'].value_counts().plot(kind='bar')
plt.title('Value Counts for contact')
```

```
plt.xlabel('contact')
plt.ylabel('Count')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```



Grafik di atas menunjukkan bagaimana bank menghubungi nasabah untuk menawarkan produk deposito.

Apa yang bisa kita lihat dari grafik ini?

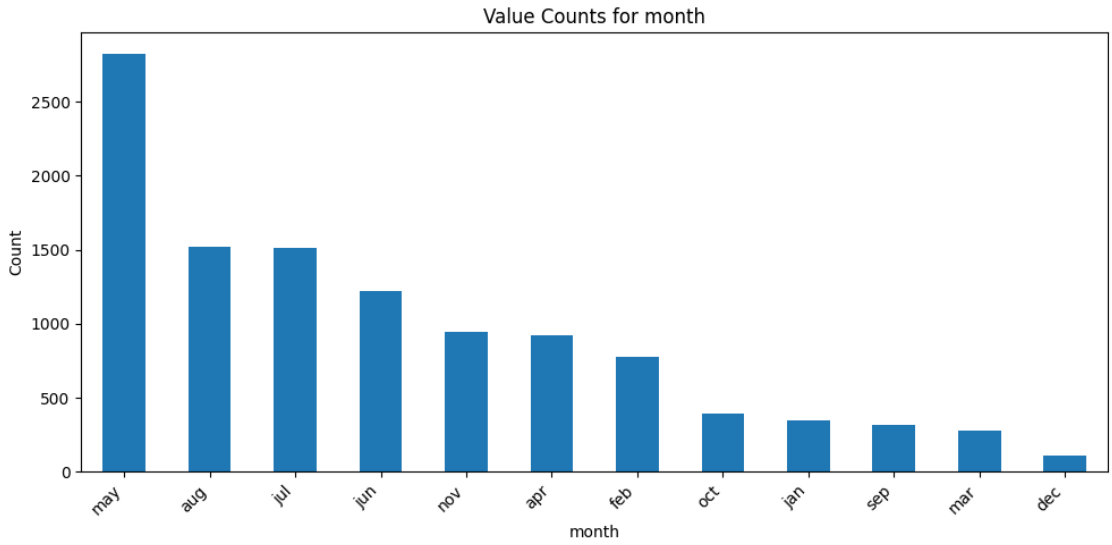
- **Sebagian besar nasabah dihubungi melalui telepon seluler (`cellular`)**. Ini menunjukkan bahwa bank lebih sering menggunakan telepon seluler sebagai sarana komunikasi untuk menjangkau nasabah.
- **Ada juga nasabah yang dihubungi melalui telepon rumah (`telephone`) dan ada juga yang cara kontaknya tidak diketahui (`unknown`)**. Ini menunjukkan bahwa bank menggunakan berbagai cara untuk menghubungi nasabah, namun telepon seluler adalah yang paling umum.

Apa artinya bagi bank?

- **Telepon seluler adalah saluran komunikasi yang paling efektif**. Karena sebagian besar nasabah dihubungi melalui telepon seluler, maka bank perlu memastikan bahwa kampanye pemasaran melalui telepon seluler berjalan efektif.
- **Perlu adanya diversifikasi saluran komunikasi**. Meskipun telepon seluler adalah saluran yang paling efektif, bank juga perlu mempertimbangkan saluran komunikasi lainnya seperti telepon rumah atau email untuk menjangkau segmen nasabah yang berbeda.
- **Pentingnya data yang akurat**. Untuk meningkatkan efektivitas kampanye pemasaran, bank perlu memastikan bahwa data kontak nasabah, terutama nomor telepon seluler, akurat dan up-to-date.

8. Kolom `month`:

```
# Kolom 'month'
plt.figure(figsize=(10, 5))
data['month'].value_counts().plot(kind='bar')
plt.title('Value Counts for month')
plt.xlabel('month')
plt.ylabel('Count')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```



Grafik di atas menunjukkan seberapa sering kampanye pemasaran bank dilakukan setiap bulannya.

Apa yang bisa kita lihat dari grafik ini?

- **Bulan Mei adalah bulan dengan frekuensi kampanye pemasaran tertinggi**. Ini artinya, kampanye pemasaran paling sering dilakukan pada bulan Mei.
- **Bulan Desember adalah bulan dengan frekuensi kampanye pemasaran terendah**. Ini menunjukkan bahwa kampanye pemasaran jarang dilakukan pada bulan Desember.
- **Secara umum, frekuensi kampanye pemasaran cenderung lebih tinggi pada bulan-bulan tertentu**. Ini mungkin karena ada faktor-faktor tertentu yang mempengaruhi keputusan bank untuk melakukan kampanye pemasaran pada bulan-bulan tersebut, seperti musim, liburan, atau event khusus.

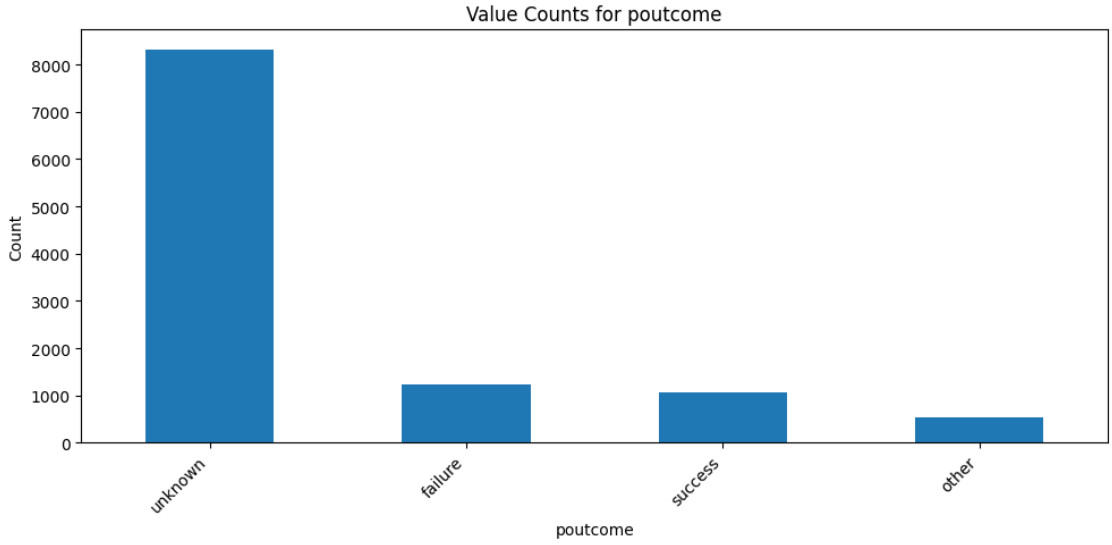
Apa artinya bagi bank?

- **Bulan Mei adalah waktu yang tepat untuk meningkatkan penjualan produk deposito**. Karena kampanye pemasaran paling sering dilakukan pada bulan Mei, maka ini adalah peluang yang baik bagi bank untuk meningkatkan penjualan produk deposito.

- **Perlu dievaluasi kembali frekuensi kampanye pada bulan Desember**. Karena kampanye pemasaran jarang dilakukan pada bulan Desember, maka bank perlu mengevaluasi kembali apakah ini adalah strategi yang tepat. Mungkin ada potensi yang terlewatkan untuk meningkatkan penjualan pada bulan Desember.
- **Perlu menganalisis faktor-faktor yang mempengaruhi frekuensi kampanye pemasaran**. Dengan memahami faktor-faktor yang mempengaruhi frekuensi kampanye pemasaran, bank dapat menyusun strategi pemasaran yang lebih efektif dan efisien.

9. Kolom `poutcome`:

```
# Kolom 'poutcome'
plt.figure(figsize=(10, 5))
data['poutcome'].value_counts().plot(kind='bar')
plt.title('Value Counts for poutcome')
plt.xlabel('poutcome')
plt.ylabel('Count')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```



Grafik di atas menunjukkan hasil dari kampanye pemasaran sebelumnya terhadap nasabah.

Apa yang bisa kita lihat dari grafik ini?

- **Sebagian besar hasil kampanye sebelumnya tidak diketahui (`unknown`)**. Ini berarti bank tidak memiliki data yang cukup mengenai hasil kampanye sebelumnya untuk sebagian besar nasabah.
- **Jumlah nasabah yang berhasil diajak berlangganan (`success`) dan gagal (`failure`) relatif sama**. Ini menunjukkan bahwa tingkat keberhasilan kampanye sebelumnya cukup seimbang.

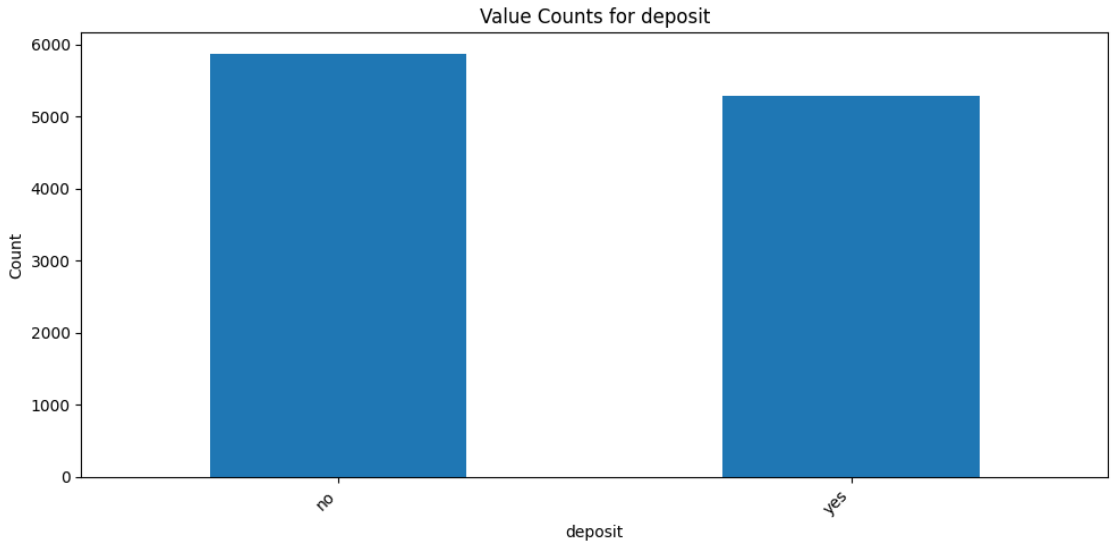
- **Hanya sebagian kecil nasabah yang memiliki kategori "`other`"**. Kategori ini mungkin mewakili hasil kampanye yang unik atau tidak termasuk dalam kategori utama.

Apa artinya bagi bank?

- **Perlu perbaikan dalam pelacakan hasil kampanye**. Bank perlu memperbaiki sistem pelacakan hasil kampanye agar dapat mengetahui dengan lebih baik hasil dari setiap kampanye yang dilakukan.
- **Hasil kampanye sebelumnya dapat menjadi acuan untuk kampanye selanjutnya**. Dengan mengetahui hasil kampanye sebelumnya, bank dapat menyusun strategi kampanye yang lebih efektif.
- **Perlu adanya analisis lebih lanjut terhadap kategori "`other`"**. Kategori "other" perlu dianalisis lebih lanjut untuk memahami apa yang menyebabkan hasil yang berbeda tersebut.

10. Kolom `deposit`:

```
# Kolom 'deposit'
plt.figure(figsize=(10, 5))
data['deposit'].value_counts().plot(kind='bar')
plt.title('Value Counts for deposit')
plt.xlabel('deposit')
plt.ylabel('Count')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```



Grafik di atas menunjukkan jumlah nasabah yang memutuskan untuk membuka deposito dan yang tidak.

Apa yang bisa kita lihat dari grafik ini?

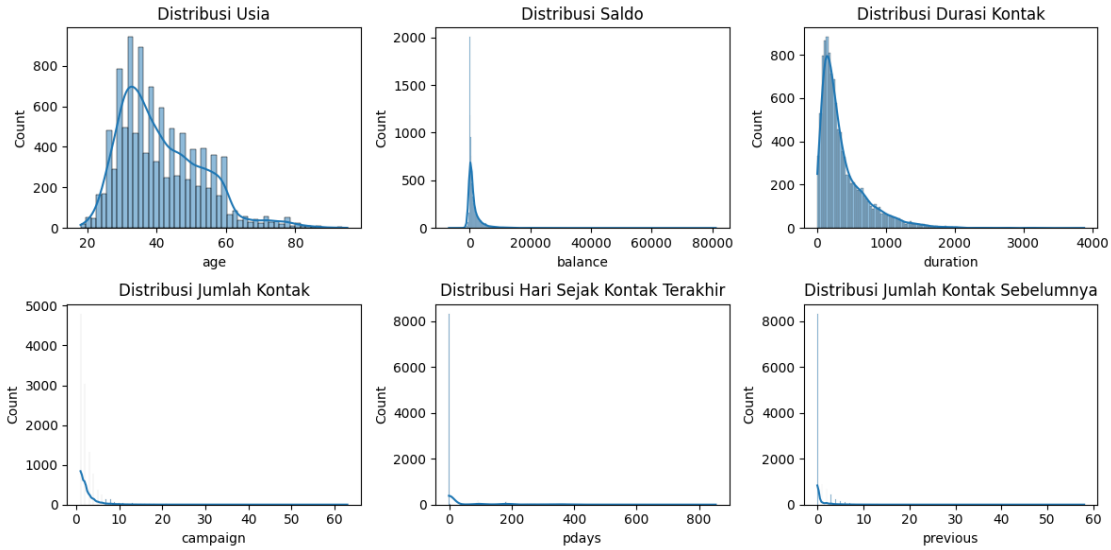
- **Jumlah nasabah yang tidak membuka deposito lebih banyak dibandingkan yang membuka deposito**. Ini artinya, sebagian besar nasabah yang dihubungi oleh bank tidak tertarik untuk membuka deposito.
- **Namun, jumlah nasabah yang membuka deposito juga cukup signifikan**. Ini menunjukkan bahwa kampanye pemasaran bank cukup berhasil dalam menarik minat sebagian nasabah untuk membuka deposito.

Apa artinya bagi bank?

- **Tingkat keberhasilan kampanye masih bisa ditingkatkan**. Meskipun ada nasabah yang membuka deposito, namun jumlahnya masih bisa ditingkatkan lagi.
- **Perlu dilakukan analisis lebih lanjut untuk mengetahui faktor-faktor yang mempengaruhi keputusan nasabah**. Dengan memahami faktor-faktor ini, bank dapat menyusun strategi pemasaran yang lebih efektif.
- **Perlu adanya segmentasi nasabah**. Tidak semua nasabah memiliki karakteristik yang sama. Dengan melakukan segmentasi nasabah, bank dapat memberikan penawaran yang lebih relevan dan personal kepada setiap segmen.

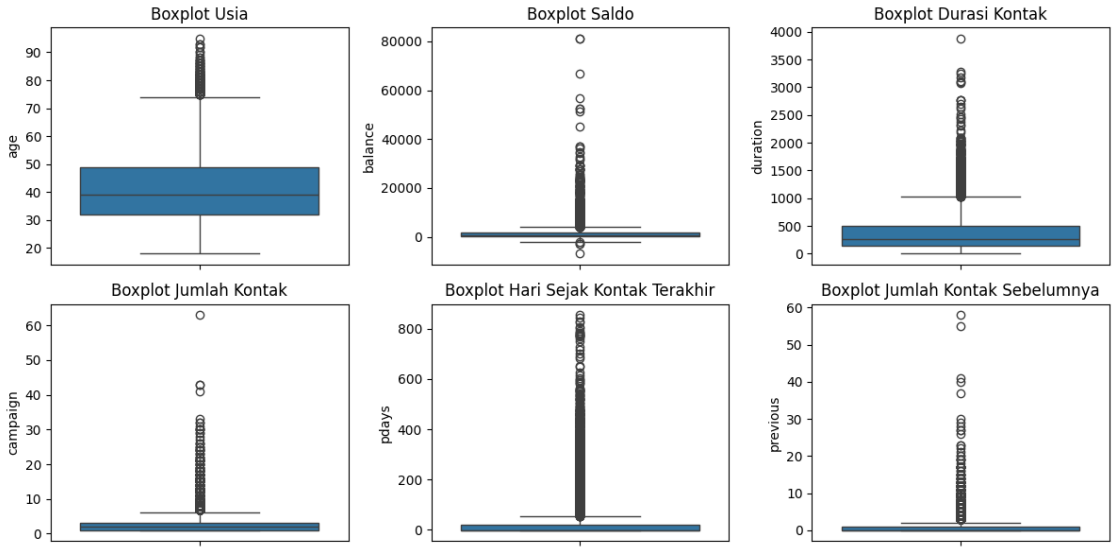
Distribusi variabel numerik

```
plt.figure(figsize=(12, 6))
plt.subplot(2, 3, 1)
sns.histplot(data['age'], kde=True)
plt.title('Distribusi Usia')
plt.subplot(2, 3, 2)
sns.histplot(data['balance'], kde=True)
plt.title('Distribusi Saldo')
plt.subplot(2, 3, 3)
sns.histplot(data['duration'], kde=True)
plt.title('Distribusi Durasi Kontak')
plt.subplot(2, 3, 4)
sns.histplot(data['campaign'], kde=True)
plt.title('Distribusi Jumlah Kontak')
plt.subplot(2, 3, 5)
sns.histplot(data['pdays'], kde=True)
plt.title('Distribusi Hari Sejak Kontak Terakhir')
plt.subplot(2, 3, 6)
sns.histplot(data['previous'], kde=True)
plt.title('Distribusi Jumlah Kontak Sebelumnya')
plt.tight_layout()
plt.show()
```



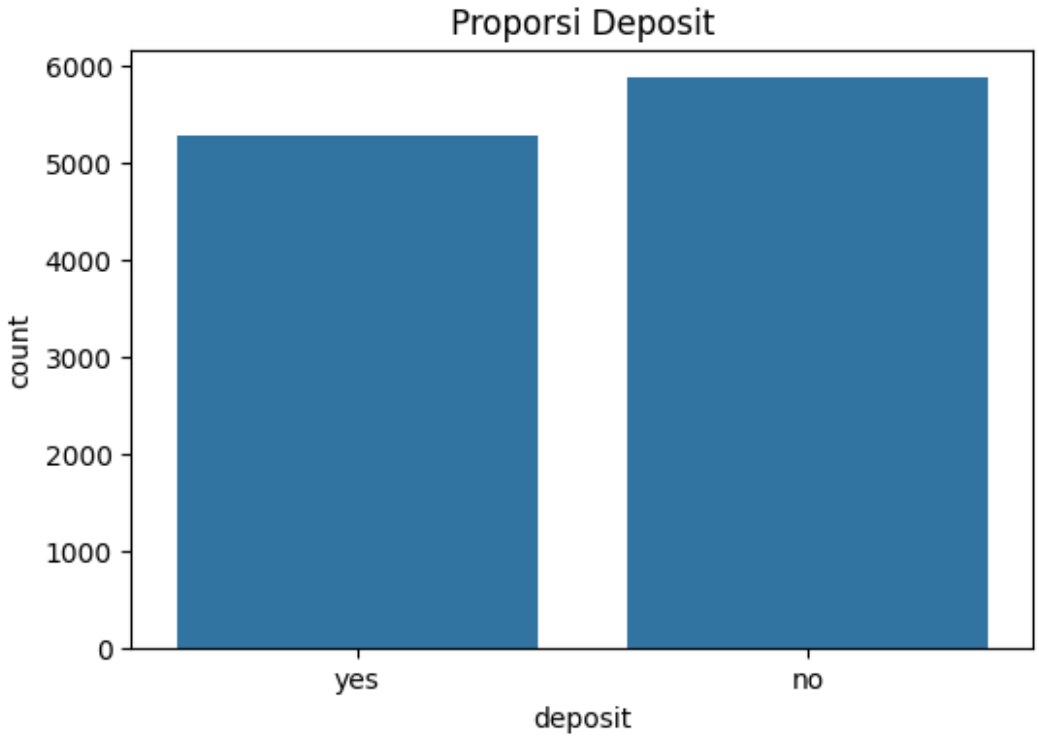
Boxplot untuk melihat outlier

```
plt.figure(figsize=(12, 6))
plt.subplot(2, 3, 1)
sns.boxplot(y=data['age'])
plt.title('Boxplot Usia')
plt.subplot(2, 3, 2)
sns.boxplot(y=data['balance'])
plt.title('Boxplot Saldo')
plt.subplot(2, 3, 3)
sns.boxplot(y=data['duration'])
plt.title('Boxplot Durasi Kontak')
plt.subplot(2, 3, 4)
sns.boxplot(y=data['campaign'])
plt.title('Boxplot Jumlah Kontak')
plt.subplot(2, 3, 5)
sns.boxplot(y=data['pdays'])
plt.title('Boxplot Hari Sejak Kontak Terakhir')
plt.subplot(2, 3, 6)
sns.boxplot(y=data['previous'])
plt.title('Boxplot Jumlah Kontak Sebelumnya')
plt.tight_layout()
plt.show()
```



Proporsi kelas target ('deposit')

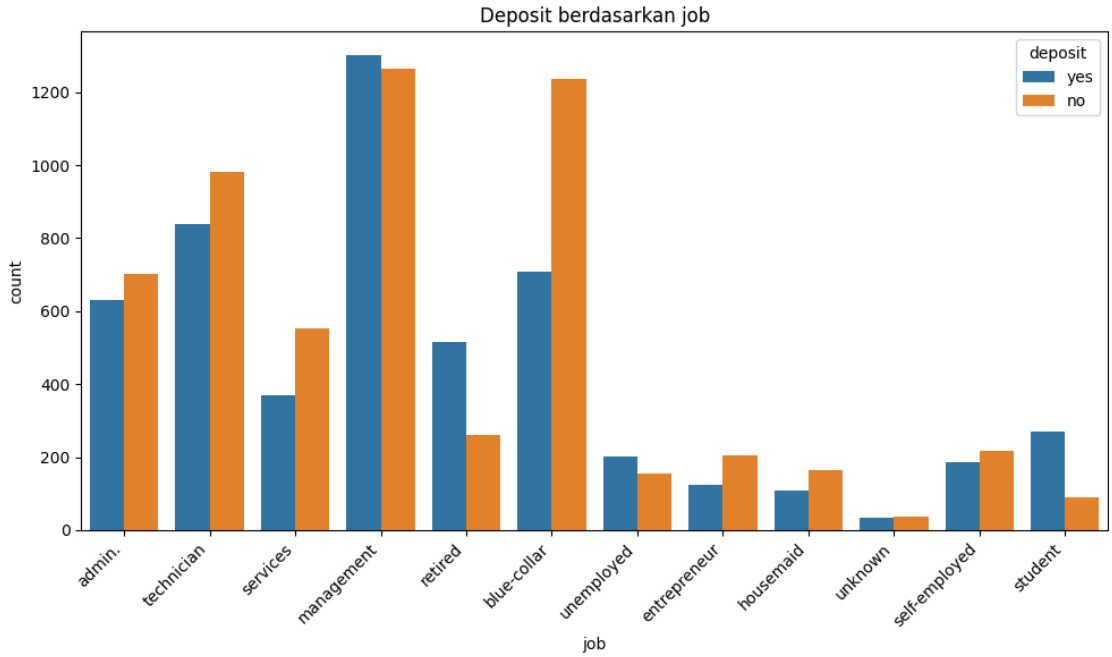
```
plt.figure(figsize=(6, 4))
sns.countplot(x=data['deposit'])
plt.title('Proporsi Deposit')
plt.show()
```

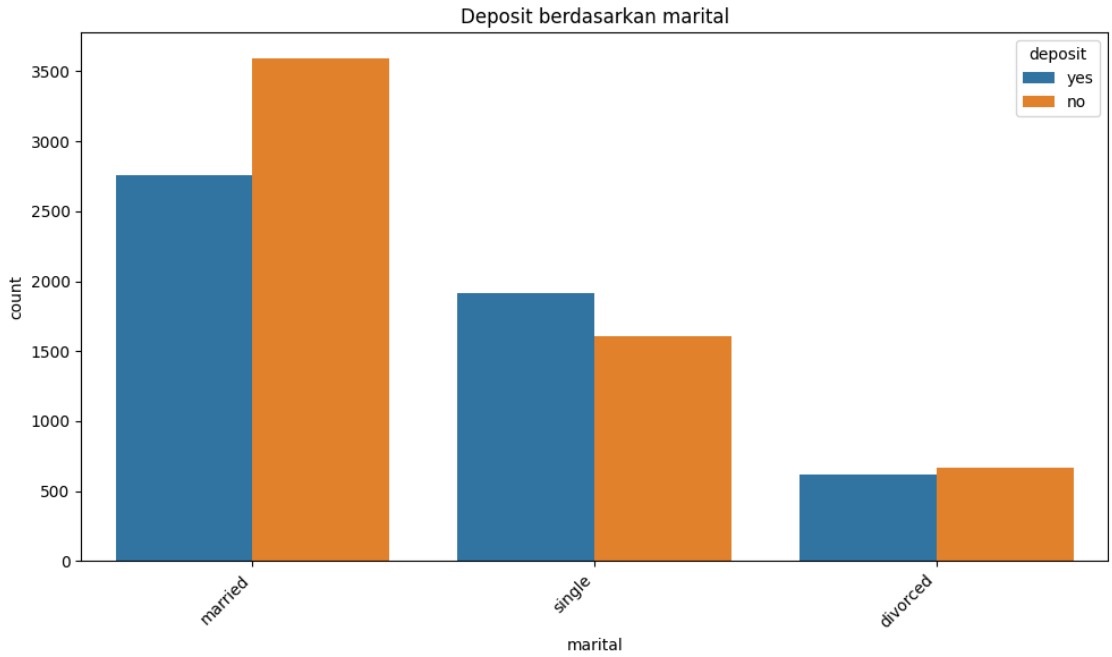


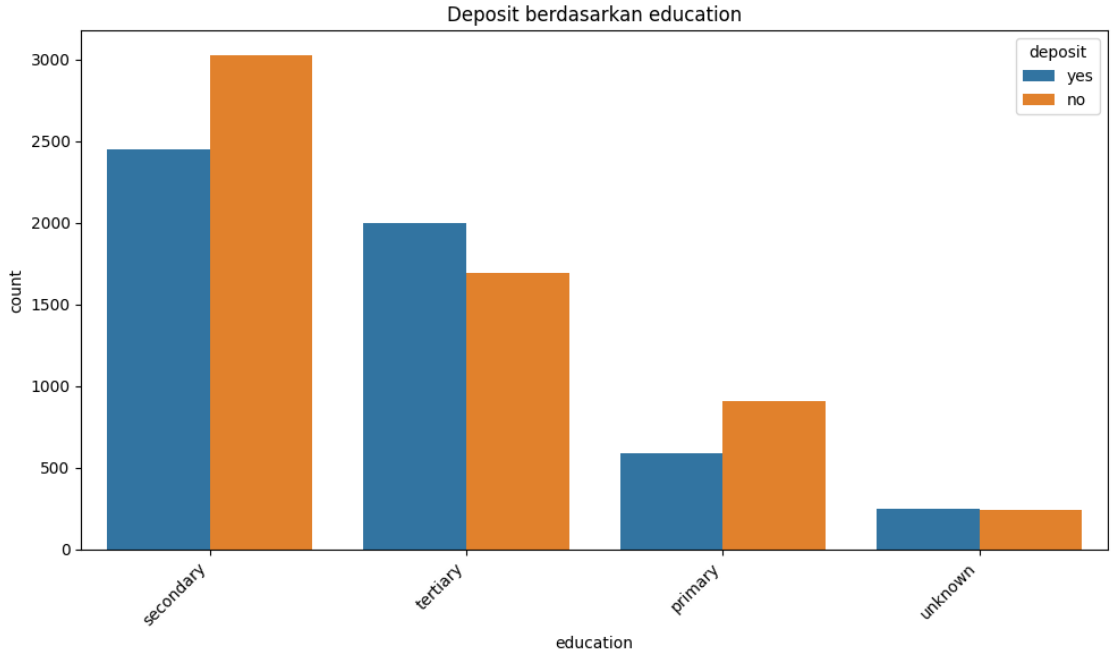
Hubungan antara variabel kategorikal dan target

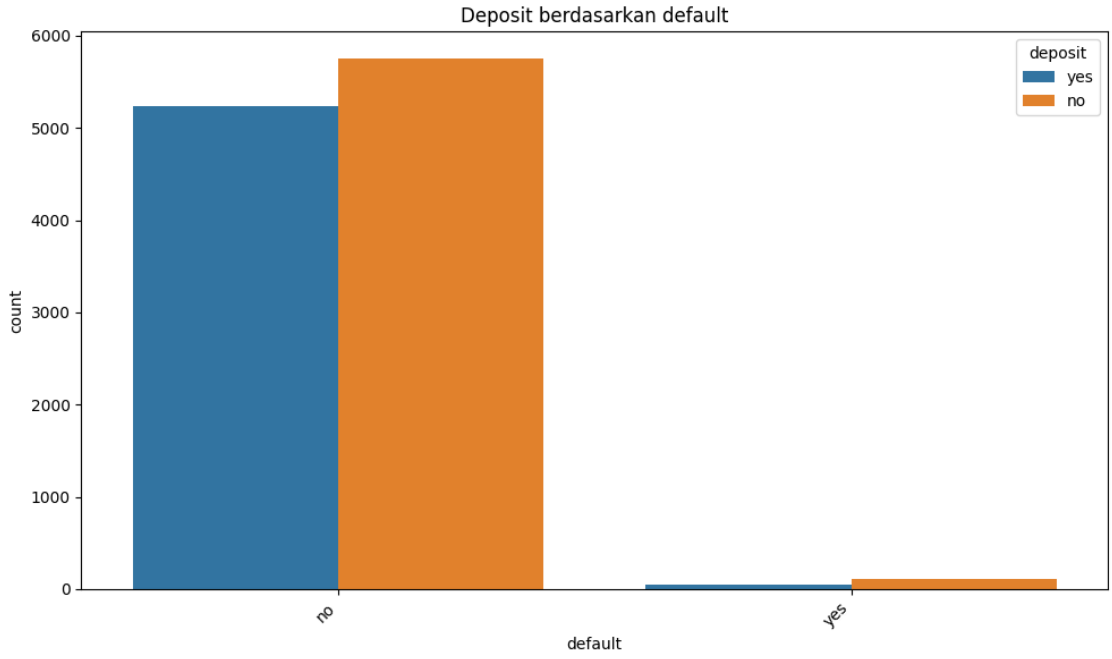
```
categorical_cols_eda = ['job', 'marital', 'education', 'default', 'housing',
'loan', 'contact', 'month', 'poutcome'] # Kolom kategorikal sebelum di-encode
for col in categorical_cols_eda:
```

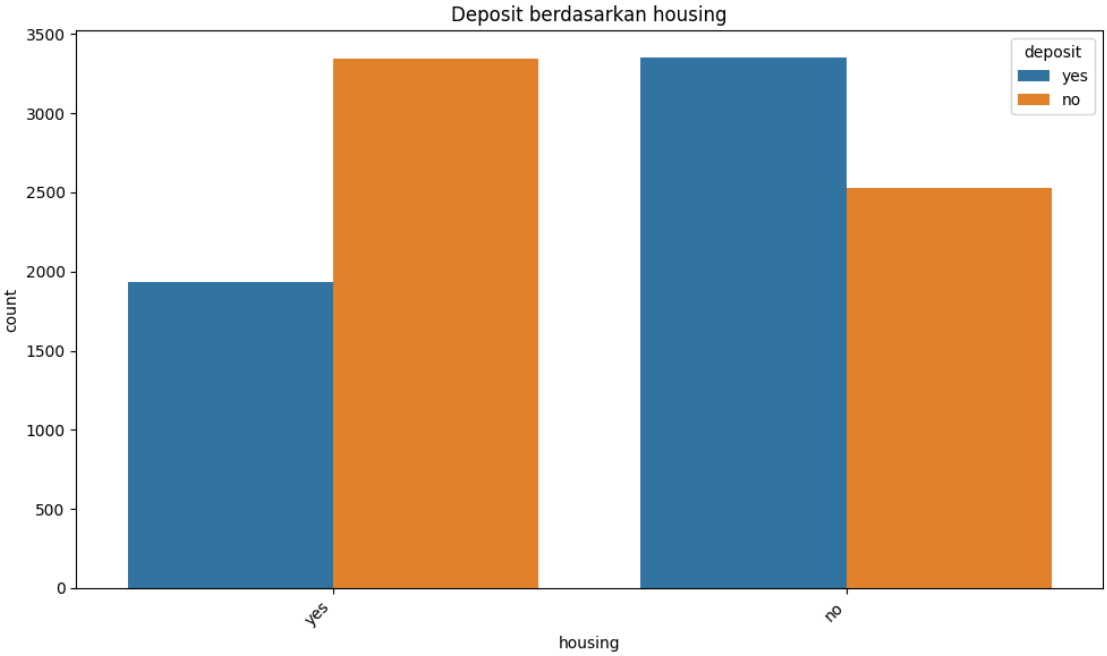
```
    plt.figure(figsize=(10, 6))
    sns.countplot(x=data[col], hue=data['deposit'])
    plt.title(f'Deposit berdasarkan {col}')
    plt.xticks(rotation=45, ha='right')  # Rotasi label sumbu x agar tidak
tumpang tindih
    plt.tight_layout()
    plt.show()
```

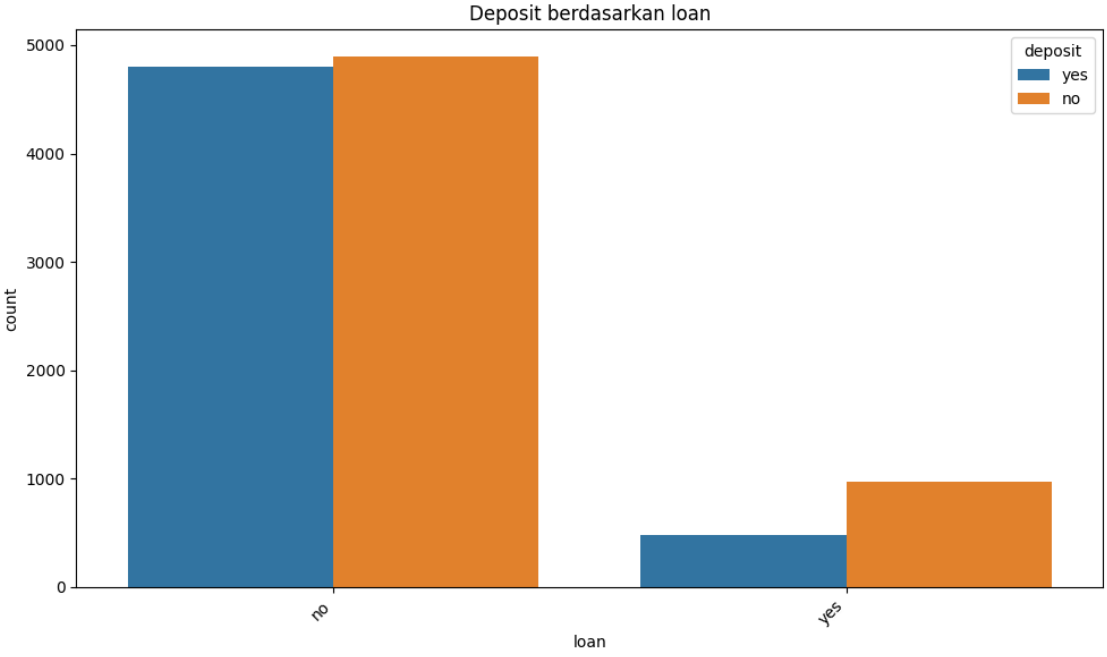


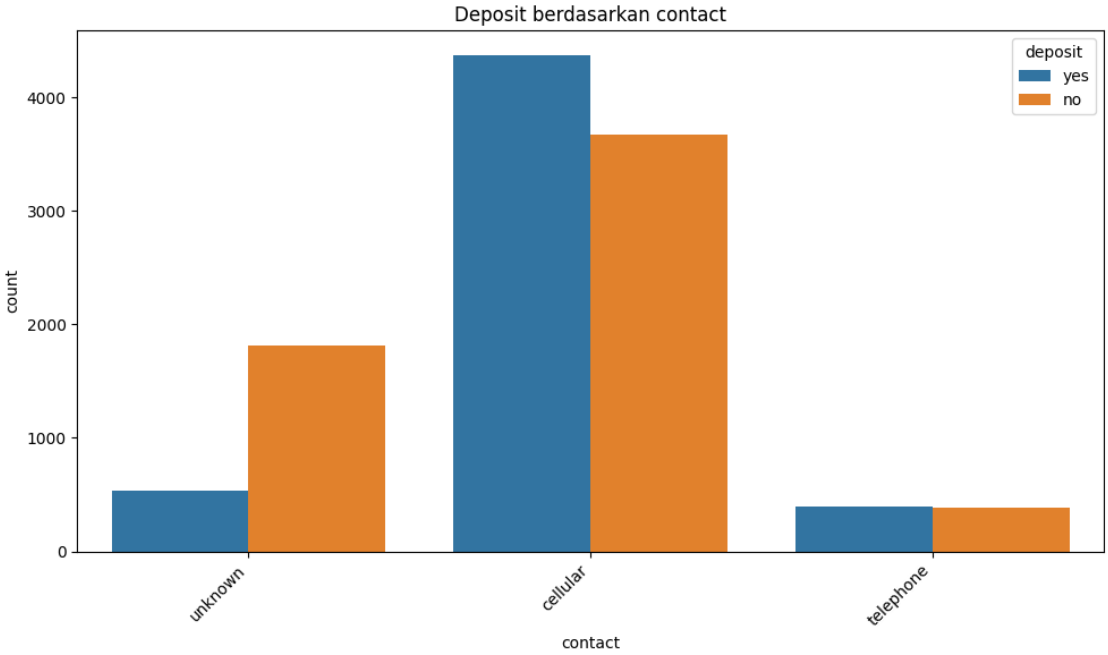


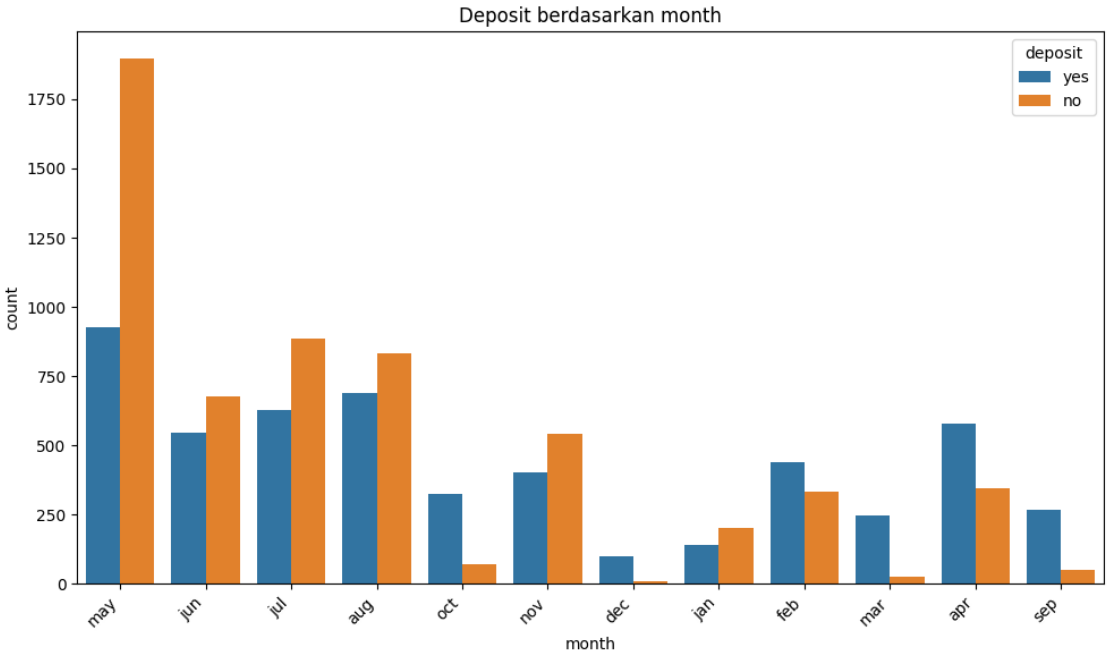

Deposit berdasarkan education
deposit
yes
no
count
3000
2500
2000
1500
1000
500
0
secondary
tertiary
primary
unknown
education

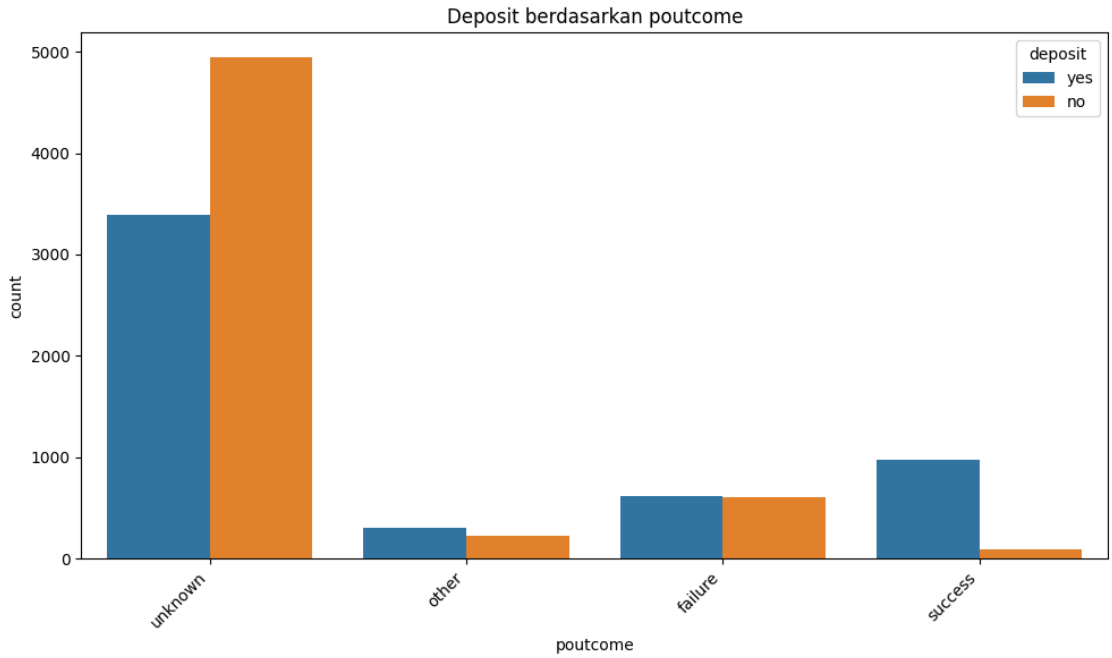

Deposit berdasarkan default
deposit
yes
no
count
6000
5000
4000
3000
2000
1000
0
no
yes
default


Deposit berdasarkan housing
deposit
yes
no
3500
3000
2500
2000
1500
1000
500
0
count
yes
no
housing


Deposit berdasarkan loan
deposit
yes
no
5000
4000
3000
2000
1000
0
count
no
yes
loan


Deposit berdasarkan contact
deposit
yes
no
count
4000
3000
2000
1000
0
unknown
cellular
telephone
contact


Deposit berdasarkan month
deposit
yes
no
count
1750
1500
1250
1000
750
500
250
0
may
jun
jul
aug
oct
nov
dec
jan
feb
mar
apr
sep
month



3. Feature Engineering

a. Membuat fitur interaksi

```
# Contoh: Kombinasi 'age' dan 'job'
data['age_job'] = data['age'].astype(str) + '_' + data['job']
encoder_age_job = OneHotEncoder(sparse_output=False, handle_unknown='ignore')
encoded_age_job = encoder_age_job.fit_transform(data[['age_job']])
encoded_age_job_df = pd.DataFrame(encoded_age_job,
columns=encoder_age_job.get_feature_names_out(['age_job']))
data = data.drop(columns=['age_job'])  # Hapus kolom asli
data = pd.concat([data, encoded_age_job_df], axis=1)
```

b. Encoding Fitur Kategorikal

Konversi variabel kategorikal menjadi numerik menggunakan One-Hot Encoding

```
categorical_cols = ['job', 'marital', 'education', 'default', 'housing',
'loan', 'contact', 'month', 'poutcome']
encoder = OneHotEncoder(sparse_output=False, handle_unknown='ignore')
#sparse=false agar output tidak sparse matrix, handle_unknown='ignore' agar
data baru
encoded_data = encoder.fit_transform(data[categorical_cols])
encoded_df = pd.DataFrame(encoded_data,
columns=encoder.get_feature_names_out(categorical_cols))
data = data.drop(categorical_cols, axis=1) # Hapus kolom kategorikal asli
data = pd.concat([data, encoded_df], axis=1) # Gabungkan data dengan hasil
encoding
```

Mengubah variabel target 'deposit' menjadi numerik (0 dan 1)

```
data['deposit'] = data['deposit'].map({'yes': 1, 'no': 0})
```

Standarisasi data numerik (opsional, tapi direkomendasikan untuk beberapa model)

```
numerical_cols = ['age', 'balance', 'duration', 'campaign', 'pdays',
'previous']
scaler = StandardScaler()
data[numerical_cols] = scaler.fit_transform(data[numerical_cols])
```

c. Binning untuk variabel numerik

```
# Contoh: Binning 'age' menjadi 3 kategori
bins = [0, 30, 60, 100]
labels = ['Muda', 'Dewasa', 'Lansia']
data['age_group'] = pd.cut(data['age'], bins=bins, labels=labels)
data = pd.get_dummies(data, columns=['age_group'], prefix=['age_group']) #
One-hot encoding untuk 'age_group'
```

d. Membuat fitur baru

```
# Contoh: Total jumlah interaksi (campaign + previous)
data['total_interactions'] = data['campaign'] + data['previous']
```