In [90]:
```python
from mpl_toolkits.mplot3d import Axes3D
from sklearn.preprocessing import StandardScaler
from mpl_toolkits.mplot3d import Axes3D
from sklearn.preprocessing import StandardScaler
import matplotlib.pyplot as plt # plotting
import numpy as np # linear algebra
import matplotlib.pyplot as plt # plotting
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import seaborn as sns
```

In [91]:
```python
path = "bank (1).csv"

df = pd.read_csv(path)

df.head(5)
```

Out[91]:

	age	job	marital	education	default	balance	housing	loan	contact	day	month	duration	campaign	pdays	previous	poutcome	deposit
0	59	admin.	married	secondary	no	2343	yes	no	unknown	5	may	1042	1	-1	0	unknown	yes
1	56	admin.	married	secondary	no	45	no	no	unknown	5	may	1467	1	-1	0	unknown	yes
2	41	technician	married	secondary	no	1270	yes	no	unknown	5	may	1389	1	-1	0	unknown	yes
3	55	services	married	secondary	no	2476	yes	no	unknown	5	may	579	1	-1	0	unknown	yes
4	54	admin.	married	tertiary	no	184	no	no	unknown	5	may	673	2	-1	0	unknown	yes

In [92]:
```python
df.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
```

In [93]:
```python
## Check the nulls
df.isna().sum()
```

Out[93]:
```
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
```

In [94]:
```python
df.dropna(how='any',inplace=True)
print(df)
```

```
       age          job  marital  education default  balance housing loan  \
0       59       admin.  married  secondary      no     2343     yes   no
1       56       admin.  married  secondary      no       45      no   no
2       41   technician  married  secondary      no     1270     yes   no
3       55     services  married  secondary      no     2476     yes   no
4       54       admin.  married   tertiary      no      184      no   no
...    ...          ...      ...        ...     ...      ...     ...  ...
11157   33  blue-collar   single    primary      no        1     yes   no
11158   39     services  married  secondary      no      733      no   no
11159   32   technician   single  secondary      no       29      no   no
11160   43   technician  married  secondary      no        0      no  yes
11161   34   technician  married  secondary      no        0      no   no

        contact  day month  duration  campaign  pdays  previous poutcome  \
0       unknown    5   may      1042         1     -1         0  unknown
1       unknown    5   may      1467         1     -1         0  unknown
2       unknown    5   may      1389         1     -1         0  unknown
3       unknown    5   may       579         1     -1         0  unknown
4       unknown    5   may       673         2     -1         0  unknown
...         ...  ...   ...       ...       ...    ...       ...      ...
11157  cellular   20   apr       257         1     -1         0  unknown
11158   unknown   16   jun        83         4     -1         0  unknown
11159  cellular   19   aug       156         2     -1         0  unknown
11160  cellular    8   may         9         2    172         5  failure
11161  cellular    9   jul       628         1     -1         0  unknown

      deposit
0         yes
1         yes
2         yes
3         yes
4         yes
...       ...
11157      no
11158      no
11159      no
11160      no
11161      no

[11162 rows x 17 columns]
```

In [95]:
```python
df.shape
```

Out[95]:
```
(11162, 17)
```

In [96]:
```python
df.columns
```

Out[96]:
```
Index(['age', 'job', 'marital', 'education', 'default', 'balance', 'housing',
       'loan', 'contact', 'day', 'month', 'duration', 'campaign', 'pdays',
       'previous', 'poutcome', 'deposit'],
      dtype='object')
```

In [97]:
```python
df.dtypes
```

Out[97]:
```
age           int64
job          object
marital      object
education    object
default      object
balance       int64
housing      object
loan         object
contact      object
day           int64
month        object
duration      int64
campaign      int64
pdays         int64
previous      int64
poutcome     object
deposit      object
dtype: object
```

In [98]:
```python
df.isnull()
```

Out[98]:

	age	job	marital	education	default	balance	housing	loan	contact	day	month	duration	campaign	pdays	previous	poutcome	deposit
0	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
11157	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
11158	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
11159	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
11160	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
11161	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False

11162 rows × 17 columns

In [99]:
```python
df.isnull().any()
```

Out[99]:
```
age          False
job          False
marital      False
education    False
default      False
balance      False
housing      False
loan         False
contact      False
day          False
month        False
duration     False
campaign     False
pdays        False
previous     False
poutcome     False
deposit      False
dtype: bool
```

In [100…
```python
df.duplicated()
```

Out[100]:
```
0        False
1        False
2        False
3        False
4        False
         ...
11157    False
11158    False
11159    False
11160    False
11161    False
Length: 11162, dtype: bool
```

In [101…
```python
df.duplicated().sum()
```

Out[101]:
```
0
```

In [102…
```python
df.describe()
```

Out[102]:

	age	balance	day	duration	campaign	pdays	previous
count	11162.000000	11162.000000	11162.000000	11162.000000	11162.000000	11162.000000	11162.000000
mean	41.231948	1528.538524	15.658036	371.993818	2.508421	51.330407	0.832557
std	11.913369	3225.413326	8.420740	347.128386	2.722077	108.758282	2.292007
min	18.000000	-6847.000000	1.000000	2.000000	1.000000	-1.000000	0.000000
25%	32.000000	122.000000	8.000000	138.000000	1.000000	-1.000000	0.000000
50%	39.000000	550.000000	15.000000	255.000000	2.000000	-1.000000	0.000000
75%	49.000000	1708.000000	22.000000	496.000000	3.000000	20.750000	1.000000
max	95.000000	81204.000000	31.000000	3881.000000	63.000000	854.000000	58.000000

In [103…
```python
df.skew()
```

```
C:\Users\Sutharsahana\AppData\Local\Temp\ipykernel_6812\1665899112.py:1: FutureWarning: The default value of numeric_only in DataFrame.skew is deprecated. In a future versio
n, it will default to False. In addition, specifying 'numeric_only=None' is deprecated. Select only valid columns or specify the value of numeric_only to silence this warnin
g.
  df.skew()
```

Out[103]:
```
age          0.862780
balance      8.224619
day          0.111330
duration     2.143695
campaign     5.545578
pdays        2.449986
previous     7.335298
dtype: float64
```

In [104…
```python
from sklearn.preprocessing import LabelEncoder
Numerics=LabelEncoder()

#data['label']=Numerics.fit_transform(data['label'])
df['job']=Numerics.fit_transform(df['job'])
df['marital']=Numerics.fit_transform(df['marital'])
df['education']=Numerics.fit_transform(df['education'])
df['default']=Numerics.fit_transform(df['default'])
df['housing']=Numerics.fit_transform(df['housing'])
df['loan']=Numerics.fit_transform(df['loan'])
df['contact']=Numerics.fit_transform(df['contact'])
df['month']=Numerics.fit_transform(df['month'])
df['poutcome']=Numerics.fit_transform(df['poutcome'])
df['deposit']=Numerics.fit_transform(df['deposit'])
```

In [105…
```python
import seaborn as sns  #seaborn
sns.pairplot(df,hue="day",diag_kind="hist")
```

Out[105]:
```
<seaborn.axisgrid.PairGrid at 0x1a2f7bb7640>
```

In [106…
```python
import seaborn as sns
import matplotlib.pyplot as plt
plt.figure(figsize=(12,8))
sns.heatmap(df.corr(),annot=True)
```

Out[106]:
```
<Axes: >
```

In [107…
```python
# Import the required libraries
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
```

In [108…
```python
X = df.drop(['deposit'], axis=1)

y = df['deposit']
print("okay")
```

```
okay
```

In [109…
```python
# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Create a decision tree classifier instance
clf = DecisionTreeClassifier()

# Train the classifier on the training data
clf.fit(X_train, y_train)

# Make predictions on the test set
y_pred = clf.predict(X_test)

# Calculate the accuracy of the model
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy:", accuracy);
```

```
Accuracy: 0.7604120017913122
```

In [ ]: