In [ ]:
```
#1.Why does this train-test split fail?
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20, random_state=42) #test size is 20% of the dataset
# The train-test split fails because the variable 'X' and 'y' are not defined in the code snippet.
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[2], line 4
      1 #1.Why does this train-test split fail?
      2 from sklearn.model_selection import train_test_split
----> 4 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20, random_state=42)

NameError: name 'X' is not defined
```

In [3]:
```
#2.from sklearn.tree import DecisionTreeClassifier

X = [[1, 2], [3, 4], [5, 6]]
y = [0, 1, 0]

model = DecisionTreeClassifier(max_depth=5, random_state=42)  # max_depth is set to 5, which is a valid value
model.fit(X, y)
print(model.score(X, y))
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[3], line 6
      3 X = [[1, 2], [3, 4], [5, 6]]
      4 y = [0, 1, 0]
----> 6 model = DecisionTreeClassifier(max_depth=5, random_state=42)  # max_depth is set to 5, which is a valid value
      7 model.fit(X, y)
      8 print(model.score(X, y))

NameError: name 'DecisionTreeClassifier' is not defined
```

In [ ]:
```
#3.from sklearn.cluster import KMeans
from sklearn.cluster import KMeans
import numpy as np

X = np.array([[1, 2], [3, 4], [5, 6]])

model = KMeans(n_clusters=2)
model.fit(X)
```

Out[ ]:
```
KMeans(n_clusters=2)
```

In [ ]:
```
#4.
import pandas as pd

df = pd.DataFrame({'A': [1, 2, 3]})
print(df)
```

```
   A
0  1
1  2
2  3
```

In [6]:
```
#5. from sklearn.tree import DecisionTreeClassifier  # Fixed typo

from sklearn.tree import DecisionTreeClassifier  # Fixed typo

model = DecisionTreeClassifier()
```

In [ ]:
```python
#6.
from sklearn.tree import DecisionTreeClassifier

model = DecisionTreeClassifier(max_depth=3)
# The max_depth parameter is set to 3, which is a valid value for the DecisionTreeClassifier.
```

In [8]:
```python
#7.
from sklearn.linear_model import LinearRegression

X = [[1], [2], [3], [4]]
y = [2, 4, 6, 8]

model = LinearRegression()
model.fit(X, y)

print(model.coef_)  # Correct attribute name
```

```
[2.]
```

In [9]:
```python
#8.
from sklearn.neighbors import KNeighborsClassifier
import numpy as np

X_train = np.array([[1, 2], [3, 4], [5, 6]])
y_train = [0, 1, 0]

model = KNeighborsClassifier(n_neighbors=1)  # Set to a valid value >= 1
model.fit(X_train, y_train)
```

Out[9]:

▾ KNeighborsClassifier

KNeighborsClassifier(n_neighbors=1)

In [10]:
```python
#9.
import pandas as pd

df = pd.DataFrame({'A': [10, 20, 30], 'B': [1, 2, 3]})
print(df.iloc[:, 0])  # Correct: iloc requires integer indexes
```

```
0    10
1    20
2    30
Name: A, dtype: int64
```

In [11]:
```python
#10.
from sklearn.ensemble import RandomForestClassifier

X_train = [[1, 2], [3, 4], [5, 6]]
y_train = [0, 1, 0]

model = RandomForestClassifier()
model.fit(X_train, y_train)

probs = model.predict_proba([[2, 3]])[0]  # Incorrect index
print(probs)
```

```
[0.75 0.25]
```

In [ ]:

In [ ]: