```
!pip install pandas scikit-learn joblib
```

```
Requirement already satisfied: pandas in
/usr/local/lib/python3.12/dist-packages (2.2.2)
Requirement already satisfied: scikit-learn in
/usr/local/lib/python3.12/dist-packages (1.6.1)
Requirement already satisfied: joblib in
/usr/local/lib/python3.12/dist-packages (1.5.2)
Requirement already satisfied: numpy>=1.26.0 in
/usr/local/lib/python3.12/dist-packages (from pandas) (2.0.2)
Requirement already satisfied: python-dateutil>=2.8.2 in
/usr/local/lib/python3.12/dist-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in
/usr/local/lib/python3.12/dist-packages (from pandas) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in
/usr/local/lib/python3.12/dist-packages (from pandas) (2025.2)
Requirement already satisfied: scipy>=1.6.0 in
/usr/local/lib/python3.12/dist-packages (from scikit-learn) (1.16.3)
Requirement already satisfied: threadpoolctl>=3.1.0 in
/usr/local/lib/python3.12/dist-packages (from scikit-learn) (3.6.0)
Requirement already satisfied: six>=1.5 in
/usr/local/lib/python3.12/dist-packages (from python-dateutil>=2.8.2-
>pandas) (1.17.0)
```

```
import pandas as pd
import joblib

from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score

data = {
    "age": [25, 30, 45, 35, 50, 23, 40, 60],
    "salary": [40000, 50000, 80000, 60000, 90000, 35000, 70000,
100000],
    "experience": [1, 3, 10, 5, 15, 1, 8, 20],
    "result": [0, 0, 1, 1, 1, 0, 1, 1]
}

df = pd.DataFrame(data)
df
```


```
{"summary":"{\n  \"name\": \"df\",\n  \"rows\": 8,\n  \"fields\": [\n
{\n      \"column\": \"age\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 12,\n        \"min\": 23,\n
\"max\": 60,\n        \"num_unique_values\": 8,\n        \"samples\":
[\n          30,\n          23,\n          25\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"salary\",\n      \"properties\":
{\n        \"dtype\": \"number\",\n        \"std\": 23518,\n
```

\"min\": 35000,\n   \"max\": 100000,\n
\"num_unique_values\": 8,\n   \"samples\": [\n   50000,\n
35000,\n   40000\n   ],\n   \"semantic_type\": \"\",\
n   \"description\": \"\"\n   }\n   },\n   {\n
\"column\": \"experience\",\n   \"properties\": {\n
\"dtype\": \"number\",\n   \"std\": 6,\n   \"min\": 1,\n
\"max\": 20,\n   \"num_unique_values\": 7,\n   \"samples\":
[\n   1,\n   3,\n   8\n   ],\n
\"semantic_type\": \"\",\n   \"description\": \"\"\n   }\
n   },\n   {\n   \"column\": \"result\",\n   \"properties\":
{\n   \"dtype\": \"number\",\n   \"std\": 0,\n
\"min\": 0,\n   \"max\": 1,\n   \"num_unique_values\": 2,\n
\"samples\": [\n   1,\n   0\n   ],\n
\"semantic_type\": \"\",\n   \"description\": \"\"\n   }\
n   }\n]\n}","type":"dataframe","variable_name":"df"}

```
X = df.drop("result", axis=1)
y = df["result"]

X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)

model = RandomForestClassifier(
    n_estimators=100,
    random_state=42
)

model.fit(X_train, y_train)
```

```
RandomForestClassifier(random_state=42)
```

```
y_pred = model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)

print("Model Accuracy:", accuracy)
```

```
Model Accuracy: 1.0
```

```
sample_input = [[28, 45000, 2]]  # age, salary, experience
prediction = model.predict(sample_input)

print("Prediction:", prediction)
```

```
Prediction: [0]
```

```
/usr/local/lib/python3.12/dist-packages/sklearn/utils/
validation.py:2739: UserWarning: X does not have valid feature names,
but RandomForestClassifier was fitted with feature names
  warnings.warn(
```

```
joblib.dump(model, "random_forest_model.pkl")
print("Model saved as random_forest_model.pkl")
```

```
Model saved as random_forest_model.pkl
```

```
from google.colab import files
files.download("random_forest_model.pkl")
```

```
<IPython.core.display.Javascript object>
```

```
<IPython.core.display.Javascript object>
```