**Task 2: End-to-End ML Pipeline with Scikit-learn Pipeline API**

**Pipelining**

```
#  Importing Required Libraries
import pandas as pd
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.pipeline import Pipeline
from sklearn.compose import ColumnTransformer
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.ensemble import RandomForestClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, classification_report
import joblib

# Load Dataset
url = "https://raw.githubusercontent.com/blastchar/telco-churn/master/WA_Fn-UseC_-Telco-Customer-Churn.csv"
df = pd.read_csv(url)

#  Basic Cleaning
df = df[df["TotalCharges"] != " "]  # remove blank rows
df["TotalCharges"] = df["TotalCharges"].astype(float)
df.drop("customerID", axis=1, inplace=True)

#  Target and Features
X = df.drop("Churn", axis=1)
y = df["Churn"].map({"Yes": 1, "No": 0})

#  Column Separation
numeric_cols = X.select_dtypes(include=['float64', 'int64']).columns.tolist()
categorical_cols = X.select_dtypes(include=['object']).columns.tolist()

#  Preprocessing Pipeline
numeric_transformer = Pipeline(steps=[
    ('scaler', StandardScaler())
])
categorical_transformer = Pipeline(steps=[
    ('encoder', OneHotEncoder(handle_unknown='ignore'))
])
preprocessor = ColumnTransformer(
    transformers=[
        ('num', numeric_transformer, numeric_cols),
        ('cat', categorical_transformer, categorical_cols)
    ]
)

# Full Pipeline with Model (RandomForest First)
model_pipeline = Pipeline(steps=[
    ('preprocessor', preprocessor),
    ('classifier', RandomForestClassifier(random_state=42))
])

#  Hyperparameter Tuning with GridSearchCV
param_grid = {
    'classifier__n_estimators': [50, 100],
    'classifier__max_depth': [5, 10]
}
grid_search = GridSearchCV(model_pipeline, param_grid, cv=3, scoring='accuracy')

#  Train-Test Split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

#  Fit Model
grid_search.fit(X_train, y_train)

#  Predict and Evaluate
y_pred = grid_search.predict(X_test)
print("Best Parameters:", grid_search.best_params_)
print("Accuracy:", accuracy_score(y_test, y_pred))
print(classification_report(y_test
```



```
#  Save the Final Model
joblib.dump(grid_search.best_estimator_, 'churn_model_pipeline.pkl')
```

```
import pandas as pd
from io import StringIO

#  Step 1: Sample Telco Churn Dataset as CSV string
data = """
customerID,gender,SeniorCitizen,Partner,Dependents,tenure,PhoneService,InternetService,Contract,MonthlyCharges,Churn
7590-VHVEG,Female,0,Yes,No,1,No,DSL,Month-to-month,29.85,No
5575-GNVDE,Male,0,No,No,34,Yes,DSL,One year,56.95,No
3668-QPYBK,Male,0,No,No,2,Yes,DSL,Month-to-month,53.85,Yes
7795-CFOCW,Male,0,No,No,45,No,DSL,One year,42.30,No
9237-HQITU,Female,0,No,No,2,Yes,Fiber optic,Month-to-month,70.70,Yes
"""

# Step 2: Load it using StringIO
df = pd.read_csv(StringIO(data))
print(df.head())

#  Step 3: Encode target
df['Churn'] = df['Churn'].map({'Yes': 1, 'No': 0})

# Step 4: Encode categorical columns simply
df['gender'] = df['gender'].map({'Male': 1, 'Female': 0})
df['Partner'] = df['Partner'].map({'Yes': 1, 'No': 0})
df['Dependents'] = df['Dependents'].map({'Yes': 1, 'No': 0})
df['PhoneService'] = df['PhoneService'].map({'Yes': 1, 'No': 0})

#  Step 5: Drop unnecessary for now
df = df.drop(['customerID', 'InternetService', 'Contract'], axis=1)

#  Step 6: Train a simple model
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score

X = df.drop('Churn', axis=1)
y = df['Churn']

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

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

y_pred = model.predict(X_test)

print("Accuracy:", accuracy_score(y_test, y_pred))
```

```
    customerID  gender  SeniorCitizen Partner Dependents  tenure PhoneService  \
0  7590-VHVEG  Female              0     Yes         No       1           No
1  5575-GNVDE    Male              0      No         No      34          Yes
2  3668-QPYBK    Male              0      No         No       2          Yes
3  7795-CFOCW    Male              0      No         No      45           No
4  9237-HQITU  Female              0      No         No       2          Yes

      InternetService        Contract  MonthlyCharges Churn
0             DSL  Month-to-month           29.85    No
1             DSL        One year           56.95    No
2             DSL  Month-to-month           53.85   Yes
3             DSL        One year           42.30    No
4     Fiber optic  Month-to-month           70.70   Yes
Accuracy: 0.0
```