```
from sklearn.preprocessing import OneHotEncoder, StandardScaler
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
import pandas as pd
import numpy as np

# Load the dataset
file_path = "C:\\Users\\Rishita Tehlan\\Downloads\\Labour Training
Dataset.csv"  # Replace with your dataset path
data = pd.read_csv(file_path)

# Features and target
features = ['Age', 'Eduacation', 'Race', 'Hisp', 'MaritalStatus',
'Nodeg', 'Earnings_1974', 'Earnings_1975']
target = 'Earnings_1978'

# Split the data
X = data[features]
y = data[target]

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

# Preprocessing
categorical_features = ['Eduacation', 'Race', 'Hisp', 'MaritalStatus',
'Nodeg']
numeric_features = ['Age', 'Earnings_1974', 'Earnings_1975']

preprocessor = ColumnTransformer(
    transformers=[
        ('num', StandardScaler(), numeric_features),
        ('cat', OneHotEncoder(handle_unknown='ignore'),
categorical_features)
    ]
)

# Model pipeline
model = Pipeline(steps=[
    ('preprocessor', preprocessor),
    ('regressor', LinearRegression())
])

# Train the model
model.fit(X_train, y_train)

# Example prediction
example_data = pd.DataFrame([{
    'Age': 30,
```

```
    'Eduacation': 'high_school',  # Replace with a valid category from
the dataset
    'Race': 'not_black',         # Replace with a valid category from
the dataset
    'Hisp': 'no',                # Replace with a valid category from
the dataset
    'MaritalStatus': 'married',  # Replace with a valid category from
the dataset
    'Nodeg': 'yes',              # Replace with a valid category from
the dataset
    'Earnings_1974': 25000,
    'Earnings_1975': 27000
}])

example_prediction = model.predict(example_data)
print(f"Predicted Earnings for 1978: {example_prediction[0]:.2f}")
```

```
Predicted Earnings for 1978: 24251.36
```