```
import pandas as pd
import numpy as np

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import OneHotEncoder
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
from sklearn.impute import SimpleImputer
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error
```

```
train_df = pd.read_csv("train.csv")
test_df  = pd.read_csv("test.csv")
```

```
y = train_df["SalePrice"]
X = train_df.drop(["SalePrice"], axis=1)
```

```
numeric_features = X.select_dtypes(include=["int64", "float64"]).columns
categorical_features = X.select_dtypes(include=["object"]).columns
```

```
numeric_transformer = Pipeline(steps=[
    ("imputer", SimpleImputer(strategy="median"))
])
```

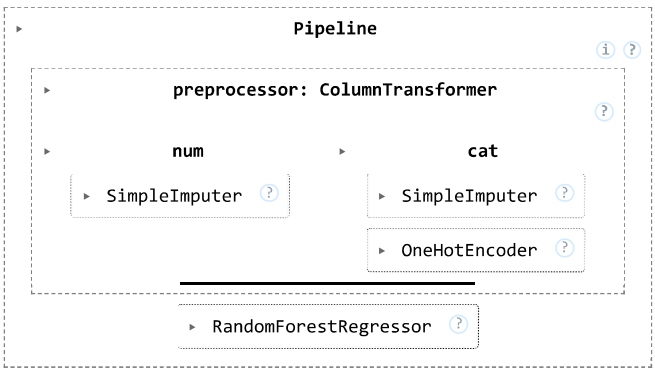
```
categorical_transformer = Pipeline(steps=[
    ("imputer", SimpleImputer(strategy="most_frequent")),
    ("onehot", OneHotEncoder(handle_unknown="ignore"))
])
```

```
preprocessor = ColumnTransformer(
    transformers=[
        ("num", numeric_transformer, numeric_features),
        ("cat", categorical_transformer, categorical_features)
    ]
)
```

```
model = RandomForestRegressor(
    n_estimators=300,
    random_state=42,
    n_jobs=-1
)

pipeline = Pipeline(steps=[
    ("preprocessor", preprocessor),
    ("model", model)
])
```

```
pipeline.fit(X, y)
```




```
X_train, X_val, y_train, y_val = train_test_split(
    X, y, test_size=0.2, random_state=42
)
```

```
pipeline.fit(X_train, y_train)
val_preds = pipeline.predict(X_val)

rmse = np.sqrt(mean_squared_error(y_val, val_preds))
print("Validation RMSE:", rmse)
```

```
Validation RMSE: 28554.982874658388
```

```
test_preds = pipeline.predict(test_df)
```

```
submission = pd.DataFrame({
    "Id": test_df["Id"],
    "SalePrice": test_preds
})

submission.to_csv("submission.csv", index=False)
```

```
from google.colab import drive
drive.mount('/content/drive')
```