

```
# Import libraries
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn import metrics
import numpy as np
import pandas as pd
```

```
# NOTE: Make sure df is your cleaned DataFrame and contains:
# - 'price' column as target
# - encoded categorical features (e.g., using pd.get_dummies)
```

```
# 1. One-Hot Encode categorical columns
df_model = pd.get_dummies(df, columns=['location', 'size_type', 'availability', 'area_type'], drop_first=True)
```

```
# 2. Separate features and target
X = df_model.drop('price', axis=1)
y = df_model['price']
```

```
# 3. Split the data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
# 4. Build and train the model
model = LinearRegression()
model.fit(X_train, y_train)
```

5. Predict

```
y_pred = model.predict(X_test)
```

6. Evaluate using RMSE

```
rmse = np.sqrt(metrics.mean_squared_error(y_test, y_pred))
```

```
print("RMSE (Root Mean Squared Error):", rmse)
```

Optional: Compare predicted vs actual

```
predicted = pd.DataFrame({'Actual Price': y_test.values, 'Predicted Price': y_pred})
```

```
print(predicted.head())
```