```
import pandas as pd
import numpy as np

# Load dataset
df = pd.read_csv("train.csv")

# Drop irrelevant columns
columns_to_drop = [
    'id', 'description', 'name', 'thumbnail_url', 'zipcode',
'amenities',
    'first_review', 'last_review', 'host_since', 'host_response_rate'
]
df = df.drop(columns=columns_to_drop)

# Drop rows with missing values in key features
df = df.dropna(subset=['review_scores_rating', 'bathrooms',
'bedrooms', 'beds'])

# Convert log_price to actual price
df['price'] = df['log_price'].apply(lambda x: round(np.exp(x), 2))
df = df.drop(columns=['log_price'])

# Save cleaned data
df.to_csv("airbnb_cleaned.csv", index=False)

from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import OneHotEncoder
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline

# Select features and target
features = [
    'city', 'property_type', 'room_type', 'accommodates',
    'bathrooms', 'bedrooms', 'beds', 'review_scores_rating'
]
target = 'price'

X = df[features]
y = df[target]

# Define categorical and numeric columns
cat_cols = ['city', 'property_type', 'room_type']
num_cols = ['accommodates', 'bathrooms', 'bedrooms', 'beds',
'review_scores_rating']

# One-hot encoding for categoricals
preprocessor = ColumnTransformer([
    ('cat', OneHotEncoder(handle_unknown='ignore'), cat_cols)
], remainder='passthrough')
```

```
# Build pipeline with regression model
model = Pipeline(steps=[
    ('preprocessor', preprocessor),
    ('regressor', LinearRegression())
])

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

# Train model
model.fit(X_train, y_train)

# Predict on test set
predictions = model.predict(X_test)

# Save model predictions to CSV
output_df = X_test.copy()
output_df['predicted_price'] = predictions.round(2)
output_df.to_csv("airbnb_predictions.csv", index=False)

df_dashboard = df[['city', 'property_type', 'room_type',
'accommodates',
                   'bathrooms', 'bedrooms', 'beds',
'review_scores_rating', 'price']]
df_dashboard.to_csv("airbnb_tableau_dashboard.csv", index=False)
```