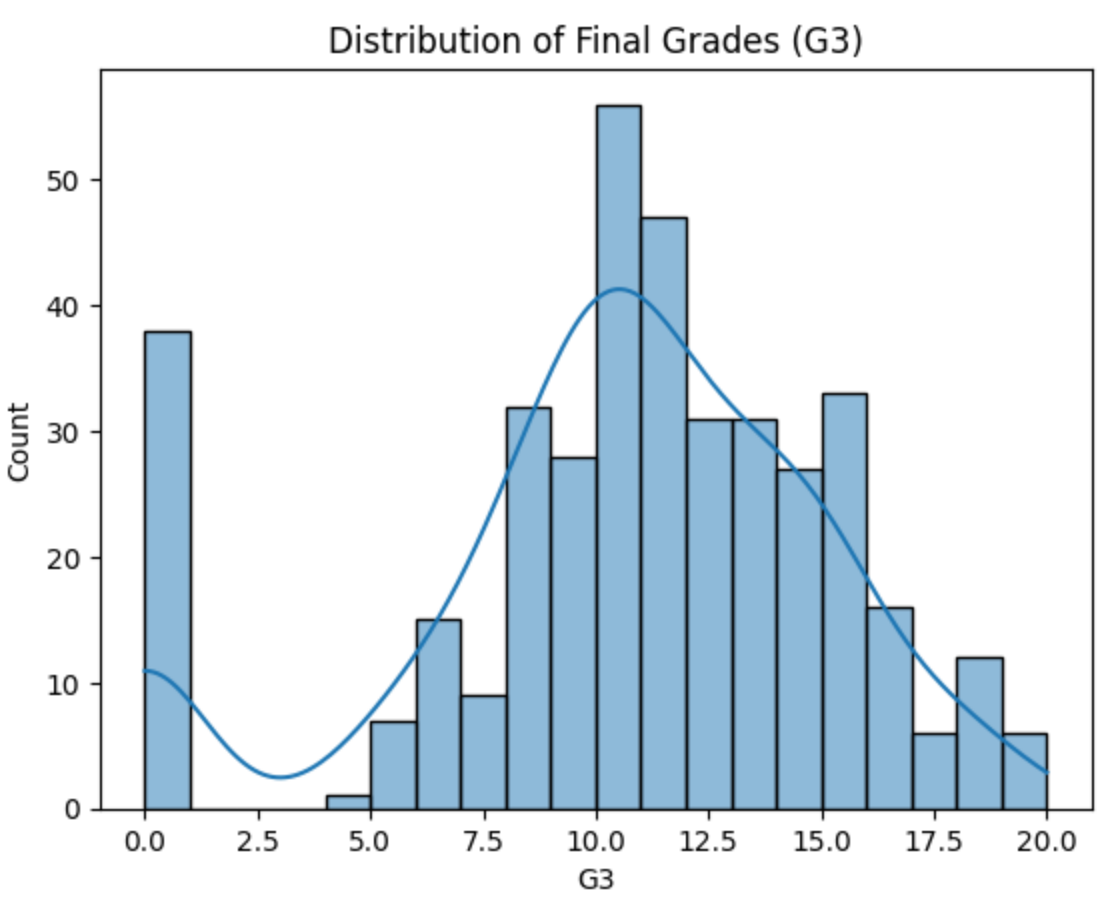
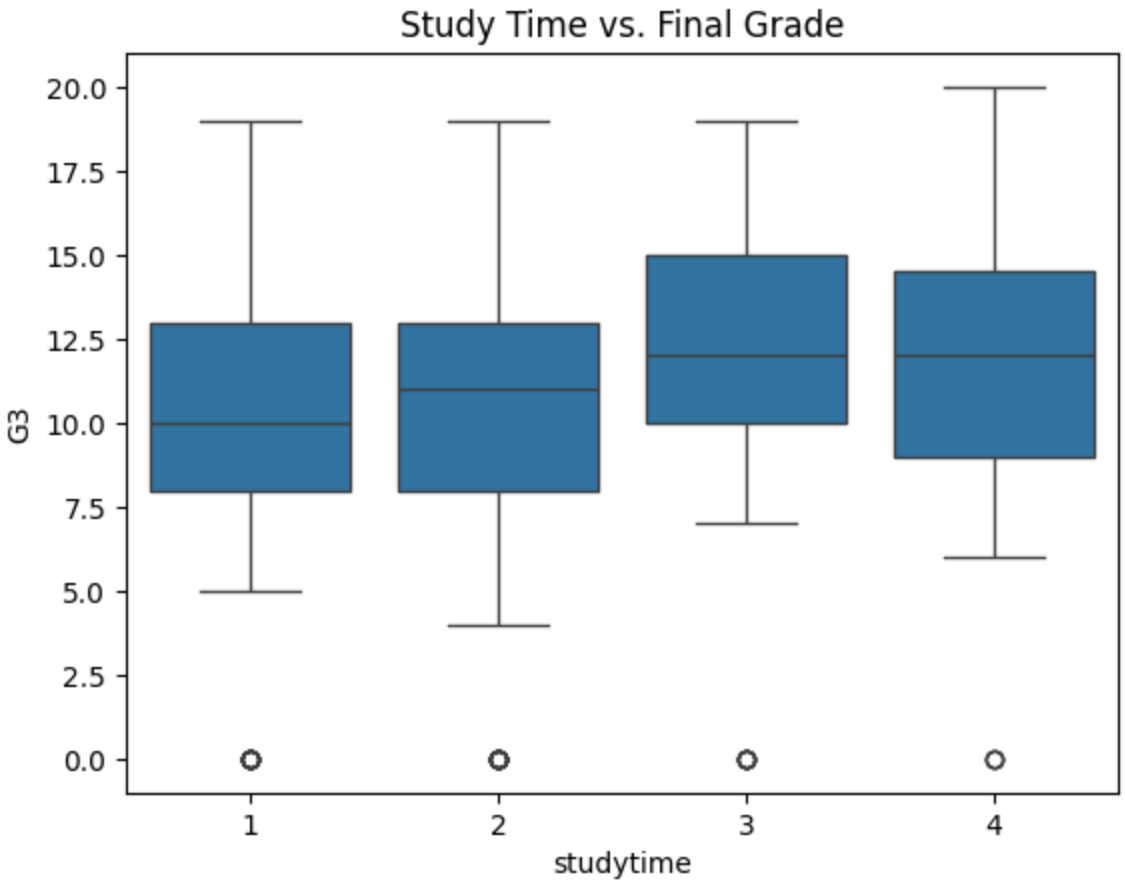
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Load data
df = pd.read_csv('student-mat.csv', sep=';')
print(df.head())

# Check missing values
print(df.isnull().sum())  # No missing values in this dataset
```

Show hidden output

```
#Target Distribution
sns.histplot(df['G3'], bins=20, kde=True)
plt.title('Distribution of Final Grades (G3)')
plt.show()
```




```
#Feature vs. Grade
sns.boxplot(x='studytime', y='G3', data=df)
plt.title('Study Time vs. Final Grade')
plt.show()
```




```
#Encode Categorical Variables
from sklearn.preprocessing import LabelEncoder

df['parent_edu_encoded'] = LabelEncoder().fit_transform(df['Medu'] + df['Fedu'])  # Combine


#Feature Selection
features = ['studytime', 'absences', 'parent_edu_encoded', 'failures', 'Dalc', 'Walc']  # [
X = df[features]
y = df['G3']


#Building The Model(Linear Regression)

df['study_failures'] = df['studytime'] * df['failures']  # Combines study time and past f
df['high_alcohol'] = (df['Dalc'] + df['Walc'] > 6).astype(int)  # 1 if high alcohol consu
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_absolute_error, r2_score

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

# Train
model = LinearRegression()
model.fit(X_train, y_train)
```

```
model.fit(X_train, y_train)

# Evaluate
y_pred = model.predict(X_test)
print(f"MAE: {mean_absolute_error(y_test, y_pred):.2f}")
print(f"R²: {r2_score(y_test, y_pred):.2f}")  # Aim for >0.3
```

```
MAE: 3.52
R²: 0.06
```

```
#Explainability?
import shap

# Explain Random Forest
explainer = shap.TreeExplainer(rf)
shap_values = explainer.shap_values(X_test)

# Plot feature importance
shap.summary_plot(shap_values, X_test, feature_names=features)
```