In [22]:
```
'''
Name: Tejaswini Jaywant Durge
Roll No: 331020
Sub: ML
Assignment No:2

'''
```

Out[22]: '\nName: Tejaswini Jaywant Durge\nRoll No: 331020\nSub: ML\nAssignment No:2\n\n'

In [6]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
```

In [7]:
```
# Load data
df = pd.read_csv('t:/ML_Prj/final_cleaned_traffic_data.csv')
df.head()
```

Out[7]:

	Weather	Road_Type	Time_of_Day	Traffic_Density	Speed_Limit	Number_of_Vehicles	Dı
0	Rainy	City Road	Morning	1.000000	100.0	5.0	
1	Clear	Rural Road	Night	1.001253	120.0	3.0	
2	Rainy	Highway	Evening	1.000000	60.0	4.0	
3	Clear	City Road	Afternoon	2.000000	60.0	3.0	
4	Rainy	Highway	Morning	1.000000	195.0	11.0	

In [8]:
```
# Select Driver_Age as X (independent) and Speed_Limit as y (dependent)
X = df['Driver_Age'].values.reshape(-1, 1)  # Independent variable
y = df['Speed_Limit'].values  # Dependent variable
```

In [9]:
```
y
```

```
Out[9]:  array([100., 120.,  60.,  60., 195., 120.,  60.,  60.,  60.,  30., 100.,
                 60.,  30.,  60.,  60.,  60., 100.,  60.,  50.,  60., 100.,  30.,
                 30.,  50., 100.,  60., 100.,  80.,  80.,  80.,  30., 100.,  60.,
                 60.,  60.,  50., 120., 100.,  60.,  50.,  60.,  60.,  50.,  60.,
                 60.,  30.,  30.,  60.,  50.,  60.,  80.,  60.,  60.,  80., 100.,
                 50., 100., 100.,  60.,  50., 200., 120.,  50.,  80.,  80.,  30.,
                 80.,  60., 206.,  30., 120.,  80.,  60.,  60.,  50.,  60.,  50.,
                 30.,  30.,  80., 100.,  60.,  80.,  30.,  80.,  50.,  60.,  80.,
                 60.,  60.,  50.,  60.,  50., 100.,  50.,  60.,  80.,  50.,  30.,
                 80.,  30., 178.,  60., 100.,  60.,  60.,  60.,  60.,  30.,  50.,
                 60.,  80.,  60.,  60.,  60.,  80.,  60.,  60., 208.,  50.,  80.,
                 50.,  60.,  80.,  60.,  50., 100., 120.,  50.,  50., 100.,  60.,
                100.,  50.,  60.,  50.,  60.,  80.,  80.,  50.,  60.,  30.,  60.,
                 60.,  80.,  80.,  80.,  30.,  80.,  60., 100., 100.,  50.,  80.,
                100.,  60.,  50., 213.,  50.,  60.,  60.,  80.,  80.,  30.,  30.,
                 30., 120., 120.,  80.,  60.,  80.,  60., 100.,  50.,  60.,  80.,
                100.,  50.,  60., 120.,  80.,  50.,  60.,  60.,  30.,  30.,  30.,
                100.,  60.,  60.,  60.,  80.,  50.,  60., 100.,  60.,  60.,  60.,
                 60., 190.,  60.,  60.,  80.,  60., 100.,  60., 196.,  60.,  60.,
                 80.,  60., 100.,  80.,  50.,  80.,  80.,  30.,  50.,  80.,  50.,
                188.,  80., 100.,  80., 100.,  50.,  60.,  60.,  50.,  30.,  30.,
                100.,  50.,  60., 195., 100., 194.,  60.,  80.,  60.,  80.,  60.,
                100.,  60.,  50.,  80., 100.,  80.,  80.,  50.,  50.,  50.,  60.,
                 60.,  60.,  60.,  60.,  60.,  80.,  80.,  50., 100.,  60.,  50.,
                120.,  50.,  50., 120.,  50., 100.,  30.,  50.,  50.,  80.,  60.,
                 30., 100.,  60.,  60.,  30.,  50.,  60.,  30.,  60., 100.,  30.,
                100.,  80.,  80., 100.,  50.,  50.,  80.,  60.,  30.,  50.,  80.,
                 60.,  80.,  50.,  50.,  50.,  60.,  60., 100.,  50.,  60.,  60.,
                100.,  60., 100.,  80., 100.,  80.,  60.,  50.,  60.,  30.,  60.,
                 60.,  30.,  50.,  60.,  50.,  50., 100.,  80.,  60.,  50.,  60.,
                 50., 100.,  60.,  60.,  30.,  60.,  50.,  60.,  60., 189.,  60.,
                 80.,  30.,  30.,  50.,  80.,  50.,  50.,  60.,  50.,  50.,  60.,
                 60.,  60.,  50., 100.,  60.,  30.,  60., 100.,  60., 100., 100.,
                 50.,  60.,  80.,  80.,  30.,  80.,  50.,  50.,  60., 100., 120.,
                 30.,  80.,  60.,  60., 100.,  60.,  60., 120.,  50.,  80.,  60.,
                 30.,  60., 100.,  60., 100.,  30.,  50.,  50., 100.,  30., 100.,
                 60., 100.,  60.,  60.,  80., 100., 100.,  80.,  60.,  60., 100.,
                 60.,  60.,  60.,  60.,  50., 120., 100., 100., 100.,  60.,  60.,
                 60.,  50.,  50.,  50.,  50.,  80., 100.,  50.,  80.,  60.,  50.,
                 80.,  60., 100., 100., 100.,  60.,  80.,  60.,  60.,  80., 193.,
                 60., 100.,  80., 100.,  60.,  80.,  60.,  50.,  50., 100., 120.,
                120.,  50.,  50.,  50., 100.,  80., 100., 100.,  80.,  50.,  80.,
                 80.,  50.,  60.,  60.,  60.,  80.,  60.,  50.,  50., 100., 120.,
                 60., 100.,  80.,  30.,  30.,  60.,  80.,  60.,  80.,  50., 100.,
                 80.,  60.,  50.,  30.,  60.,  50.,  50.,  60., 100.,  60.,  80.,
                 60., 100.,  50.,  60., 100.,  50., 189.,  30.,  60.,  50.,  30.,
                 60., 100.,  60.,  50.,  80., 100.,  60.,  60.,  60.,  80., 100.,
                 80.,  80.,  50.,  60.,  30.,  60.,  60., 120.,  60., 100.,  60.,
                 30.,  60.,  30., 100.,  60., 100., 185.,  60.,  50.,  60.,  80.,
                 50., 100.,  60.,  50.,  30.,  80., 199.,  60.,  60.,  50.,  50.,
                 80.,  50.,  60.,  50.,  50., 192.,  30.,  80., 120.,  50., 100.,
                 60.,  30.,  60., 100.,  80.,  80., 100.,  60.,  50.,  80.,  30.,
                100.,  30.,  50.,  60.,  80., 213., 100.,  60.,  30.,  50.,  50.,
                 60.,  30.,  50.,  30., 100.,  30., 100.,  50.,  80.,  50., 120.,
                 60., 100.,  30.,  50.,  60.,  50.,  60.,  80.,  30.,  30.,  60.,
                100.,  30.,  50.,  60., 100.,  60., 120.,  80.,  80., 100., 120.,
```

```
        80.,  80.,  50.,  60., 120.,  60.,  50.,  60.,  80.,  60.,  50.,
        30.,  80.,  60., 100.,  50.,  50.,  60., 100.,  60.,  80., 198.,
        60., 100., 192.,  80.,  60.,  30.,  60.,  80.,  80.,  50.,  50.,
        60.,  30.,  50.,  80.,  60.,  50.,  60.,  50.,  80.,  30.,  30.,
        30.,  50.,  60.,  80.,  80.,  60.,  80.,  50., 100.,  60.,  60.,
       120.,  80., 206.,  60.,  60.,  50.,  80.,  50.,  60.,  80.,  60.,
       212.,  60.,  30., 100.,  60.,  60.,  60.,  30.,  80.,  30.,  80.,
        30.,  50., 100., 181.,  60.,  30.,  60.,  50.,  80.,  60.,  80.,
        50.,  80.,  60.,  80.,  60.,  60.,  60.,  60.,  60., 100.,  60.,
        50.,  60.,  80.,  60.,  50.,  60.,  60.,  30.,  50.,  50.,  50.,
        60., 100.,  50.,  50.,  50.,  60., 100.,  80.,  60.,  60.,  50.,
       100.,  60.,  50.,  60.,  60.,  80.,  60.,  50.,  80.,  80.,  60.,
       100., 120.,  80., 100., 100., 120.,  60.,  60., 100.,  80.,  50.,
       100.,  50.,  80.,  30.,  60.,  80.,  50.,  80.,  80.,  50., 100.,
       100.,  80., 120.,  60.,  60.,  60.,  80.,  60.,  60.,  60., 100.,
        80.,  80.,  50., 100., 100., 100., 178.,  50.,  80., 100.,  30.,
        50.,  60.,  30.,  60.,  60.,  60.])
```

In [10]:
```python
# 2. Split the data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_sta
```

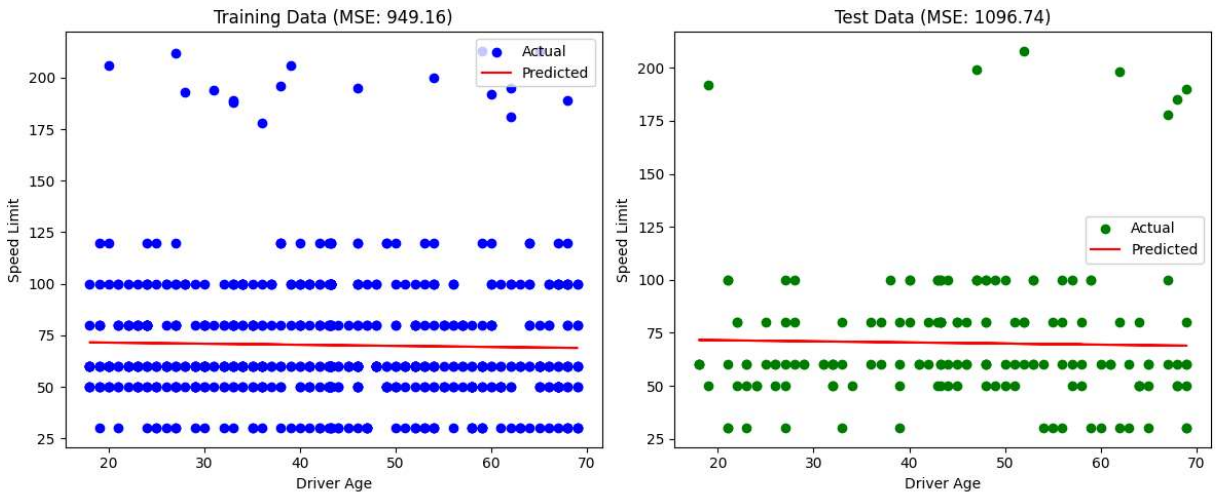
In [11]:
```python
# 3. Create and train linear regression model
model = LinearRegression()
model.fit(X_train, y_train)
```

Out[11]:
```
▾ LinearRegression
LinearRegression()
```

In [12]:
```python
## Make predictions
y_train_pred = model.predict(X_train)
y_test_pred = model.predict(X_test)

## Calculate MSE
train_mse = mean_squared_error(y_train, y_train_pred)
test_mse = mean_squared_error(y_test, y_test_pred)
r2 = r2_score(y_test, model.predict(X_test))
```

In [13]:
```python
## Plotting
plt.figure(figsize=(12, 5))

# Training data plot
plt.subplot(1, 2, 1)
plt.scatter(X_train, y_train, color='blue', label='Actual')
plt.plot(X_train, y_train_pred, color='red', label='Predicted')
plt.title(f'Training Data (MSE: {train_mse:.2f})')
plt.xlabel('Driver Age')
plt.ylabel('Speed Limit')
plt.legend()

# Test data plot
plt.subplot(1, 2, 2)
plt.scatter(X_test, y_test, color='green', label='Actual')
```

```
plt.plot(X_test, y_test_pred, color='red', label='Predicted')
plt.title(f'Test Data (MSE: {test_mse:.2f})')
plt.xlabel('Driver Age')
plt.ylabel('Speed Limit')
plt.legend()

plt.tight_layout()
plt.show()

# Print model details
print(f"\nRegression Line: y = {model.coef_[0]:.2f}x + {model.intercept_:.2f}")
print(f"Training MSE: {train_mse:.2f}")
print(f"Test MSE: {test_mse:.2f}")
print(f"R² Score: {r2:.4f}")
```




```
Regression Line: y = -0.05x + 72.54
Training MSE: 949.16
Test MSE: 1096.74
R² Score: -0.0058
```

lets check with other attributes

In [14]:
```
# Create figure for multiple plots
fig = plt.figure(figsize=(15, 12))

# 1. Driver Age vs Speed Limit with regression line
plt.subplot(2, 2, 1)
sns.regplot(data=df, x='Driver_Age', y='Speed_Limit',
            scatter_kws={'alpha':0.5, 'color':'darkblue'},
            line_kws={'color': 'red', 'linewidth': 2})
plt.title('Driver Age vs Speed Limit', fontsize=12)
plt.xlabel('Driver Age (years)')
plt.ylabel('Speed Limit (km/h)')
```

Out[14]:
```
Text(0, 0.5, 'Speed Limit (km/h)')
```

In [15]:

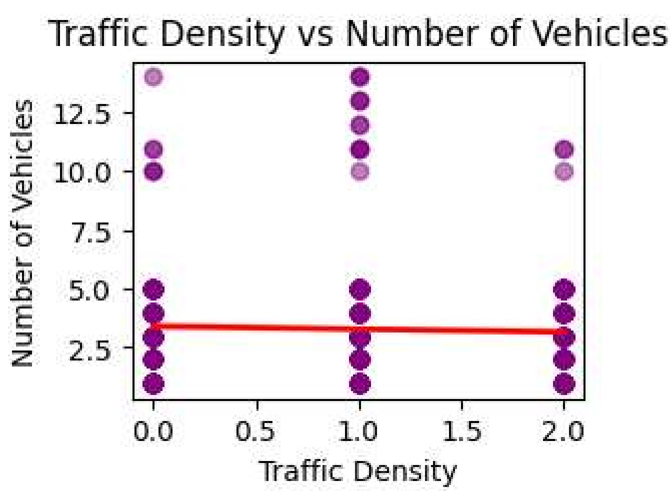
```
# 2. Driver Experience vs Speed Limit
plt.subplot(2, 2, 2)
sns.regplot(data=df, x='Driver_Experience', y='Speed_Limit',
            scatter_kws={'alpha':0.5, 'color':'darkgreen'},
            line_kws={'color': 'red', 'linewidth': 2})
plt.title('Driver Experience vs Speed Limit', fontsize=12)
plt.xlabel('Driver Experience (years)')
plt.ylabel('Speed Limit (km/h)')
```

Out[15]:

```
Text(0, 0.5, 'Speed Limit (km/h)')
```

In [16]:
```
# 3. Traffic Density vs Number of Vehicles
plt.subplot(2, 2, 3)
sns.regplot(data=df, x='Traffic_Density', y='Number_of_Vehicles',
            scatter_kws={'alpha':0.5, 'color':'purple'},
            line_kws={'color': 'red', 'linewidth': 2})
plt.title('Traffic Density vs Number of Vehicles', fontsize=12)
plt.xlabel('Traffic Density')
plt.ylabel('Number of Vehicles')
```

Out[16]:
```
Text(0, 0.5, 'Number of Vehicles')
```



In [ ]:

In [17]:
```
# Print regression metrics
print("\nRegression Analysis Results:")
print("-" * 80)

feature_pairs = [
    ('Driver_Age', 'Speed_Limit'),
    ('Driver_Experience', 'Speed_Limit'),
    ('Traffic_Density', 'Number_of_Vehicles')
]

for x_var, y_var in feature_pairs:
```

```
    # Prepare data
    X = df[x_var].values.reshape(-1, 1)
    y = df[y_var].values

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

    # Fit model
    model = LinearRegression()
    model.fit(X_train, y_train)

    # Calculate metrics
    train_mse = mean_squared_error(y_train, model.predict(X_train))
    test_mse = mean_squared_error(y_test, model.predict(X_test))
    r2 = r2_score(y_test, model.predict(X_test))

    print(f"\n{x_var} vs {y_var}:")
    print(f"Training MSE: {train_mse:.2f}")
    print(f"Test MSE: {test_mse:.2f}")
    print(f"R² Score: {r2:.4f}")
```

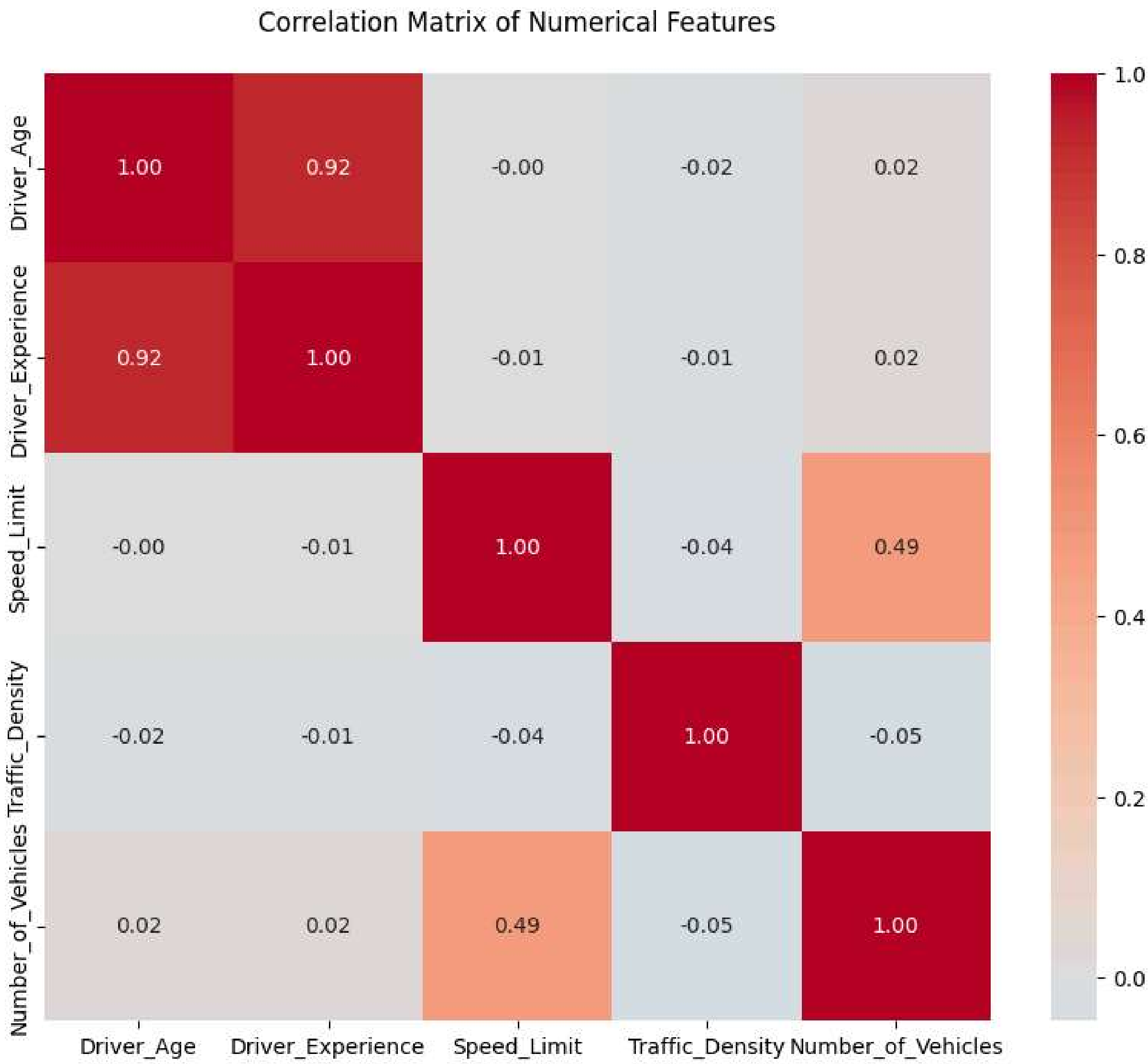
```
Regression Analysis Results:
--------------------------------------------------------------------------------

Driver_Age vs Speed_Limit:
Training MSE: 949.16
Test MSE: 1096.74
R² Score: -0.0058

Driver_Experience vs Speed_Limit:
Training MSE: 949.03
Test MSE: 1096.77
R² Score: -0.0058

Traffic_Density vs Number_of_Vehicles:
Training MSE: 3.58
Test MSE: 4.97
R² Score: -0.0079
```

In [ ]:

In [18]:

```
# Correlation Analysis
plt.figure(figsize=(10, 8))
correlation_vars = ['Driver_Age', 'Driver_Experience',
                   'Speed_Limit', 'Traffic_Density',
                   'Number_of_Vehicles']
correlation_matrix = df[correlation_vars].corr()

sns.heatmap(correlation_matrix,
            annot=True,
            cmap='coolwarm',
            center=0,
            fmt='.2f',
            square=True)
```

```
plt.title('Correlation Matrix of Numerical Features', pad=20)
plt.show()
```



In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: