```python
In [2]:   import pandas as pd
          import matplotlib.pyplot as plt
          from sklearn.linear_model import LinearRegression
          from sklearn.model_selection import train_test_split
```
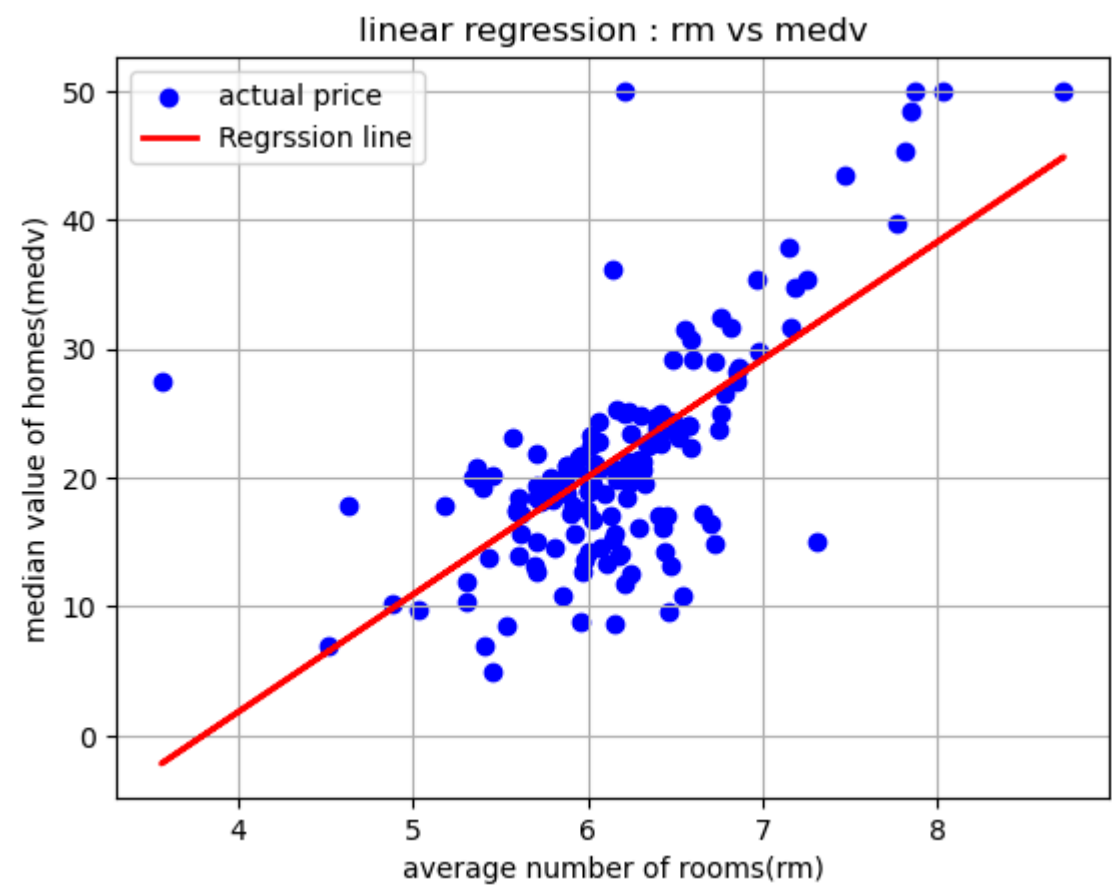
```python
In [9]:   data=pd.read_csv("BostonHousing.csv")
          X=data[['rm']]
          y=data['medv']
          X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.3,random_state=42)
          model=LinearRegression()
          model.fit(X_train,y_train)
          y_pred=model.predict(X_test)
```

```python
In [10]:  plt.scatter(X_test,y_test,color='blue',label='actual price')
          plt.plot(X_test,y_pred,color='red',linewidth=2,label='Regrssion line')
          plt.xlabel("average number of rooms(rm)")
          plt.ylabel("median value of homes(medv)")
          plt.title("linear regression : rm vs medv")
          plt.legend()
          plt.grid(True)
          plt.show()
```
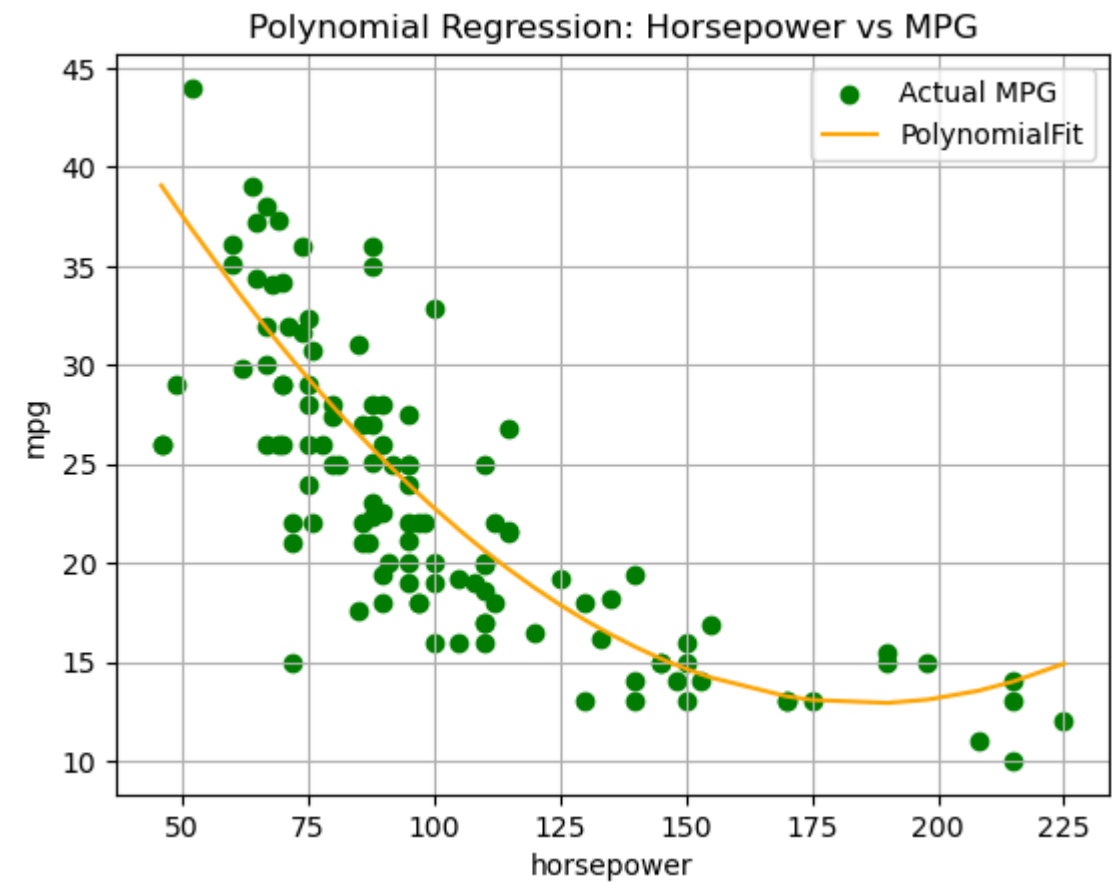


```python
In [15]:  import pandas as pd
          import numpy as np
          import matplotlib.pyplot as plt
          from sklearn.linear_model import LinearRegression
          from sklearn.preprocessing import PolynomialFeatures
          from sklearn.model_selection import train_test_split
```

```python
In [16]:  data=pd.read_csv("mpg.csv")
          data.replace({'horsepower': {'?': np.nan}}, inplace=True)
          data.dropna(subset=['horsepower'],inplace=True)
          data['horsepower']=data['horsepower'].astype(float)
```

```python
In [17]:  X=data[['horsepower']]
          y=data['mpg']
          X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.3,random_state=42)
          poly=PolynomialFeatures(degree=2)
          X_train_poly=poly.fit_transform(X_train)
          X_test_poly=poly.transform(X_test)
          model=LinearRegression()
          model.fit(X_train_poly,y_train)
          y_pred=model.predict(X_test_poly)
```

```python
In [18]:  plt.scatter(X_test, y_test, color='green', label='Actual MPG')
          sorted_idx = X_test['horsepower'].argsort()
          plt.plot(X_test.iloc[sorted_idx], y_pred[sorted_idx], color='orange', label='PolynomialFit')
          plt.xlabel("horsepower")
          plt.ylabel("mpg")
          plt.title("Polynomial Regression: Horsepower vs MPG")
          plt.legend()
          plt.grid(True)
          plt.show()
```



```
In [ ]:
```