- Is there a significant difference in the median value of houses bounded by the Charles river or not?
- Is there a difference in median values of houses of each proportion of owner-occupied units built before 1940?
- Can we conclude that there is no relationship between Nitric oxide concentrations and the proportion of non-retail business acres per town?
- What is the impact of an additional weighted distance to the five Boston employment centres on the median value of owner-occupied homes?

Data Dictionary

Field	Description
CRIM	per capita crime rate by town
ZN	proportion of residential land zoned for lots over 25,000 sq.ft.
INDUS	proportion of non-retail business acres per town.
CHAS	Charles River dummy variable (1 if tract bounds river; 0 otherwise)
NOX	nitric oxides concentration (parts per 10 million)
RM	average number of rooms per dwelling
AGE	proportion of owner-occupied units built prior to 1940
DIS	weighted distances to five Boston employment centres
RAD	index of accessibility to radial highways
TAX	full-value property-tax rate per 10,000
PTRATIO	pupil-teacher ratio by town
LSTAT	lower status of the population
MEDV	Median value of owner-occupied homes in 1000s

Import Libraries

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import statsmodels.api as sm
import datetime
import scipy.stats

%matplotlib inline
%autosave 60
sns.set_style('dark')
sns.set(font_scale=1.2)
```

```python
import warnings
warnings.filterwarnings('ignore')

pd.set_option('display.max_columns',None)
pd.set_option('display.width', 1000)

np.random.seed(0)
np.set_printoptions(suppress=True)
```

```
Autosaving every 60 seconds
```

In [2]:
```python
df = pd.read_csv("boston_housing.csv")
```

In [3]:
```python
df
```

Out[3]:

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRAT
0	0.00632	18.0	2.31	0	0.538	6.575	65.2	4.0900	1	296	1!
1	0.02731	0.0	7.07	0	0.469	6.421	78.9	4.9671	2	242	1
2	0.02729	0.0	7.07	0	0.469	7.185	61.1	4.9671	2	242	1
3	0.03237	0.0	2.18	0	0.458	6.998	45.8	6.0622	3	222	1
4	0.06905	0.0	2.18	0	0.458	7.147	54.2	6.0622	3	222	1
...	...	...	...	...	...	...	...	...	...	...	
501	0.06263	0.0	11.93	0	0.573	6.593	69.1	2.4786	1	273	2
502	0.04527	0.0	11.93	0	0.573	6.120	76.7	2.2875	1	273	2
503	0.06076	0.0	11.93	0	0.573	6.976	91.0	2.1675	1	273	2
504	0.10959	0.0	11.93	0	0.573	6.794	89.3	2.3889	1	273	2
505	0.04741	0.0	11.93	0	0.573	6.030	80.8	2.5050	1	273	2

506 rows × 13 columns

Data Analysis

In [4]:
```python
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 506 entries, 0 to 505
Data columns (total 13 columns):
 #   Column   Non-Null Count  Dtype  
---  ------   --------------  -----  
 0   CRIM     506 non-null    float64
 1   ZN       506 non-null    float64
 2   INDUS    506 non-null    float64
 3   CHAS     506 non-null    int64  
 4   NOX      506 non-null    float64
 5   RM       506 non-null    float64
 6   AGE      506 non-null    float64
 7   DIS      506 non-null    float64
 8   RAD      506 non-null    int64  
 9   TAX      506 non-null    int64  
 10  PTRATIO  506 non-null    float64
 11  LSTAT    506 non-null    float64
 12  MEDV     506 non-null    float64
dtypes: float64(10), int64(3)
memory usage: 51.5 KB
```

In [5]:
```
df.describe()
```

Out[5]:

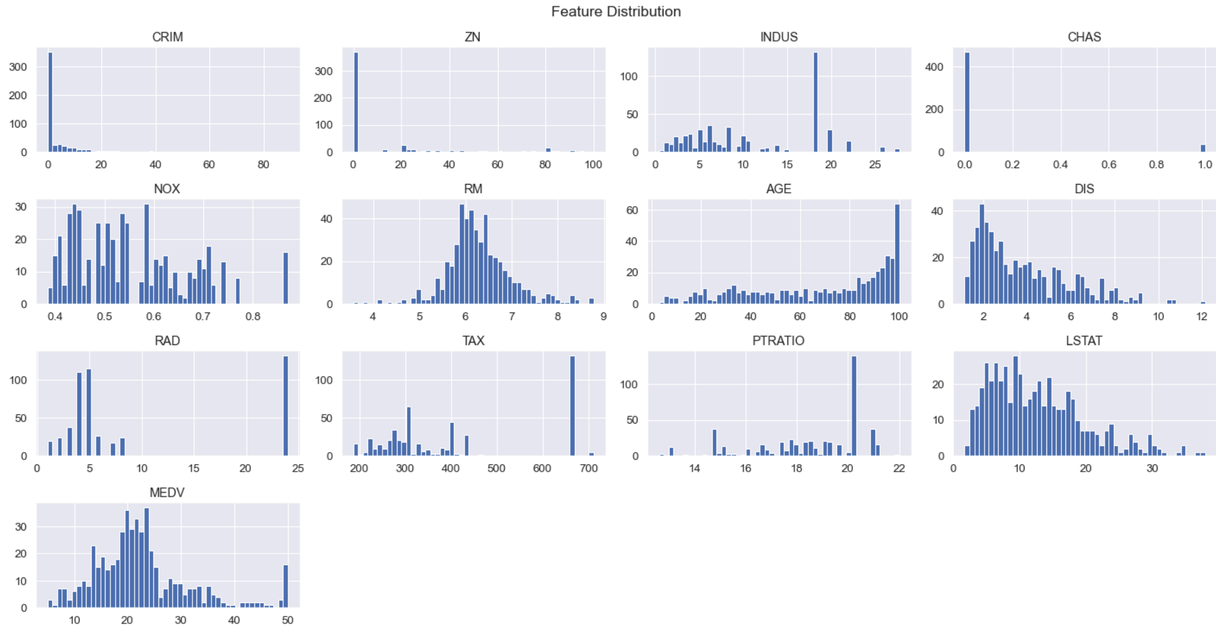
	CRIM	ZN	INDUS	CHAS	NOX	RM
count	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000
mean	3.613524	11.363636	11.136779	0.069170	0.554695	6.284634
std	8.601545	23.322453	6.860353	0.253994	0.115878	0.702617
min	0.006320	0.000000	0.460000	0.000000	0.385000	3.561000
25%	0.082045	0.000000	5.190000	0.000000	0.449000	5.885500
50%	0.256510	0.000000	9.690000	0.000000	0.538000	6.208500
75%	3.677082	12.500000	18.100000	0.000000	0.624000	6.623500
max	88.976200	100.000000	27.740000	1.000000	0.871000	8.780000

In [6]:
```
df.columns
```

Out[6]:
```
Index(['CRIM', 'ZN', 'INDUS', 'CHAS', 'NOX', 'RM', 'AGE', 'DIS', 'RAD', 'TA
X', 'PTRATIO', 'LSTAT', 'MEDV'], dtype='object')
```

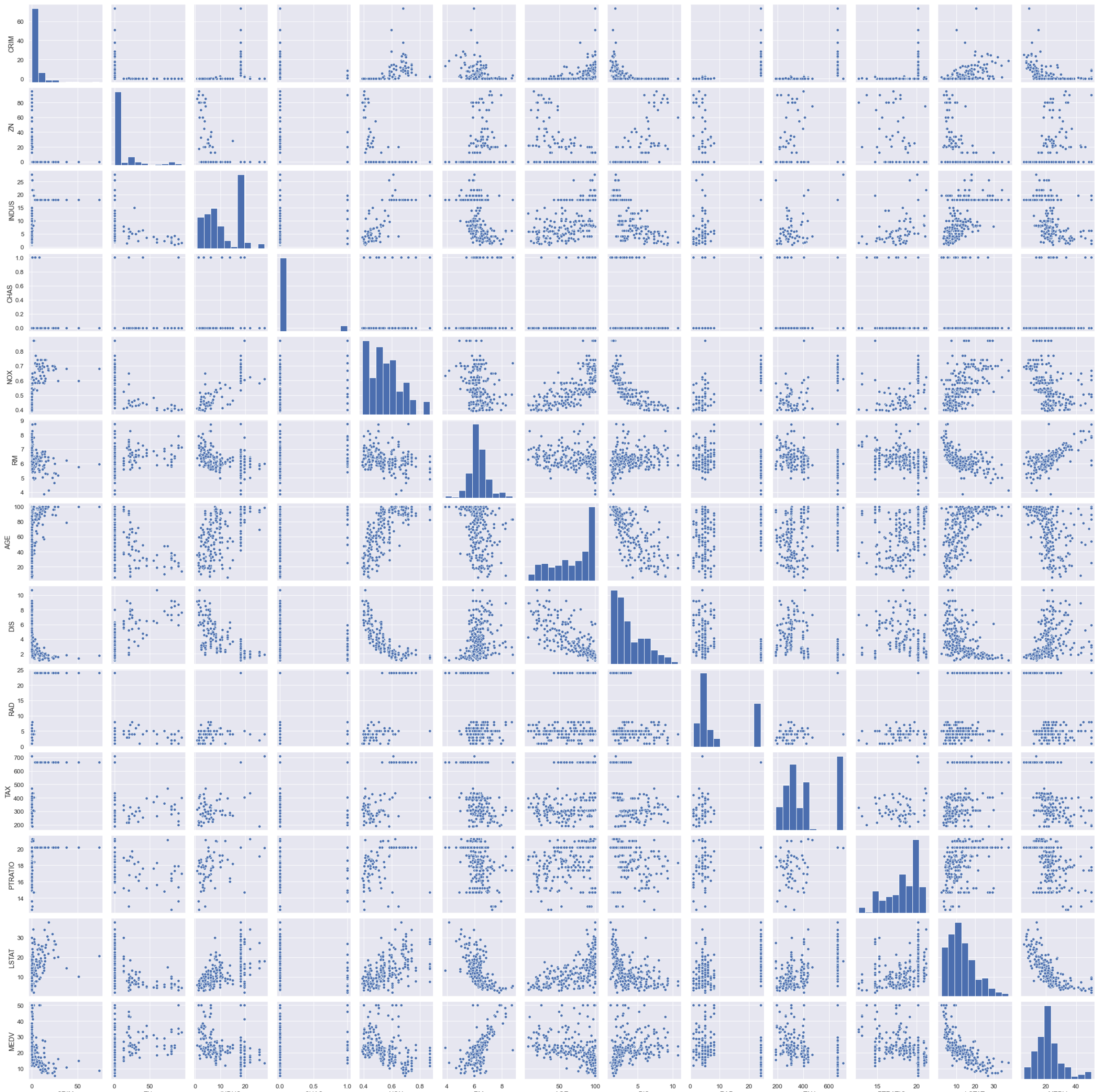
Data Visualization

In [7]:
```
df.hist(bins=50, figsize=(20,10))
plt.suptitle('Feature Distribution', x=0.5, y=1.02, ha='center', fontsize='l
plt.tight_layout()
plt.show()
```



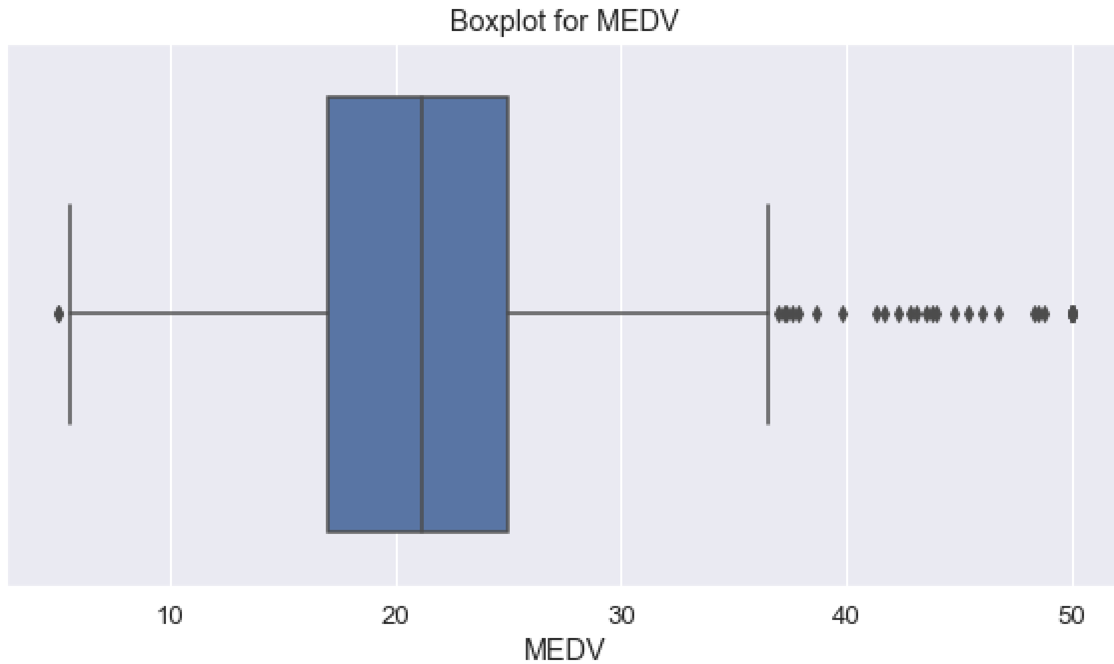
In [8]:
```
plt.figure(figsize=(20,20))
plt.suptitle('Pairplots of features', x=0.5, y=1.02, ha='center', fontsize='
sns.pairplot(df.sample(250))
plt.show()
```

```
<Figure size 1440x1440 with 0 Axes>
```



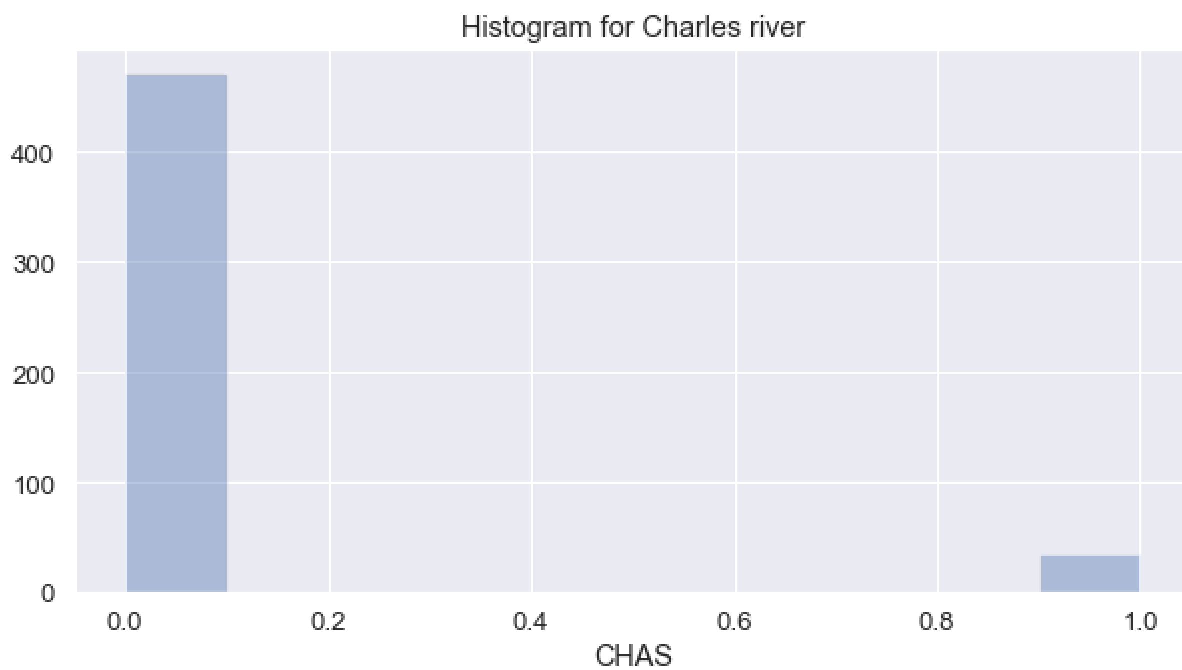
Task 4

In [9]:
```
plt.figure(figsize=(10,5))
sns.boxplot(x=df.MEDV)
plt.title("Boxplot for MEDV")
plt.show()
```



Note: Outliers after third quartile.

In [10]:
```
plt.figure(figsize=(10,5))
sns.distplot(a=df.CHAS,bins=10, kde=False)
plt.title("Histogram for Charles river")
plt.show()
```



In [11]:
```
df.loc[(df["AGE"] <= 35),'age_group'] = '35 years and younger'
df.loc[(df["AGE"] > 35) & (df["AGE"]<70),'age_group'] = 'between 35 and 70 y
df.loc[(df["AGE"] >= 70),'age_group'] = '70 years and older'
```

In [12]:
```
df
```

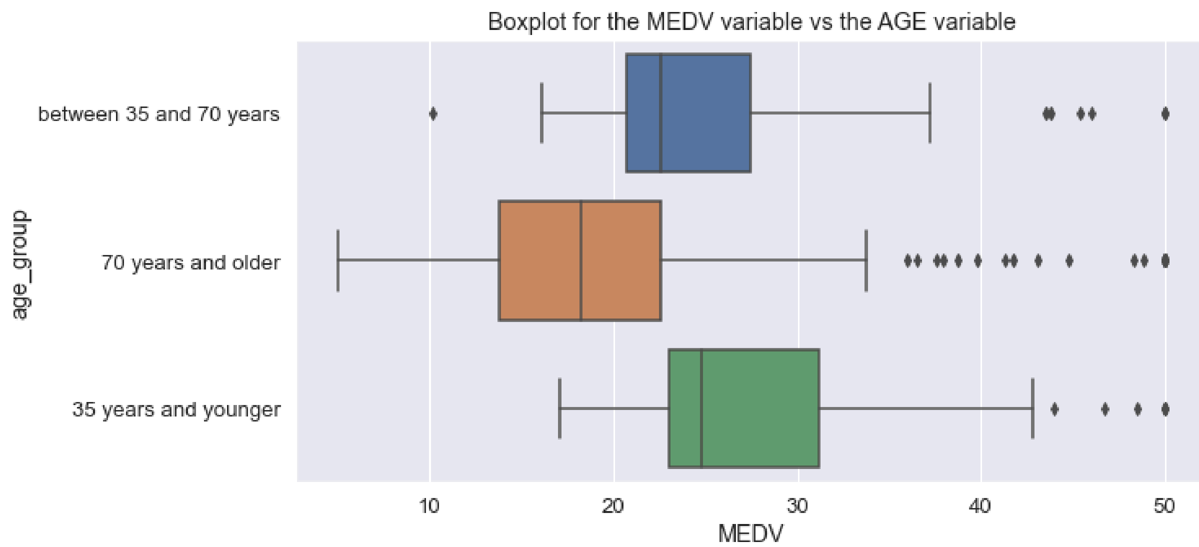
Out[12]:

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRAT
0	0.00632	18.0	2.31	0	0.538	6.575	65.2	4.0900	1	296	15
1	0.02731	0.0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17
2	0.02729	0.0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17
3	0.03237	0.0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18
4	0.06905	0.0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18
...	...	...	...	...	...	...	...	...	...	...	
501	0.06263	0.0	11.93	0	0.573	6.593	69.1	2.4786	1	273	21
502	0.04527	0.0	11.93	0	0.573	6.120	76.7	2.2875	1	273	21
503	0.06076	0.0	11.93	0	0.573	6.976	91.0	2.1675	1	273	21
504	0.10959	0.0	11.93	0	0.573	6.794	89.3	2.3889	1	273	21
505	0.04741	0.0	11.93	0	0.573	6.030	80.8	2.5050	1	273	21

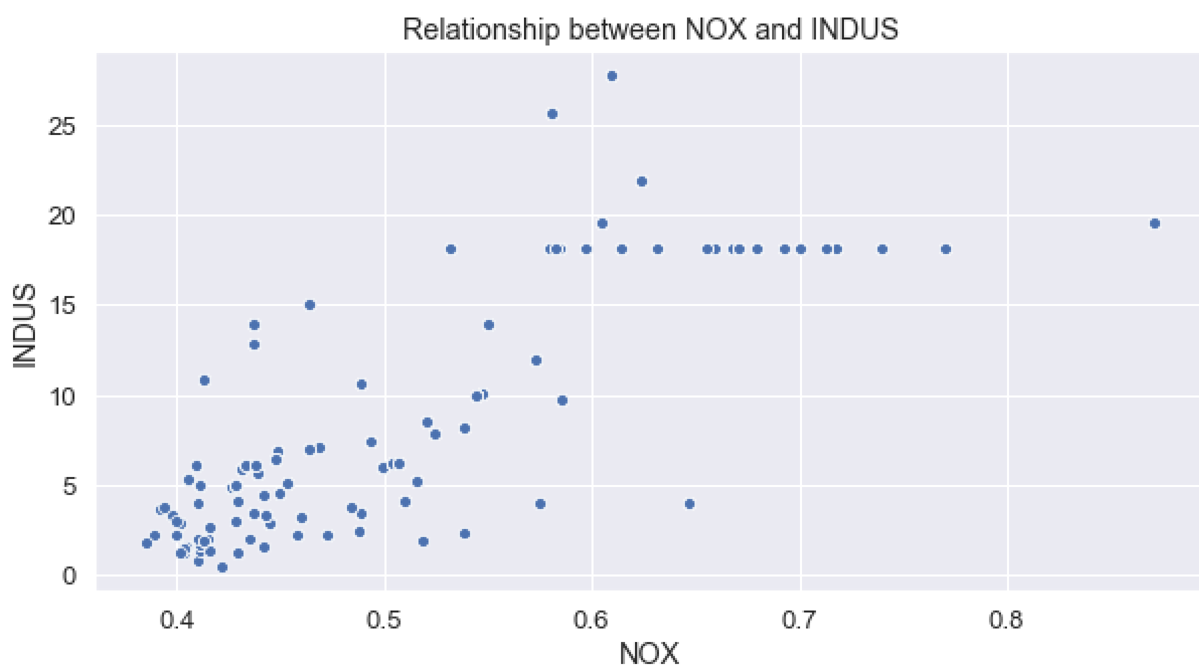
506 rows × 14 columns

In [13]:

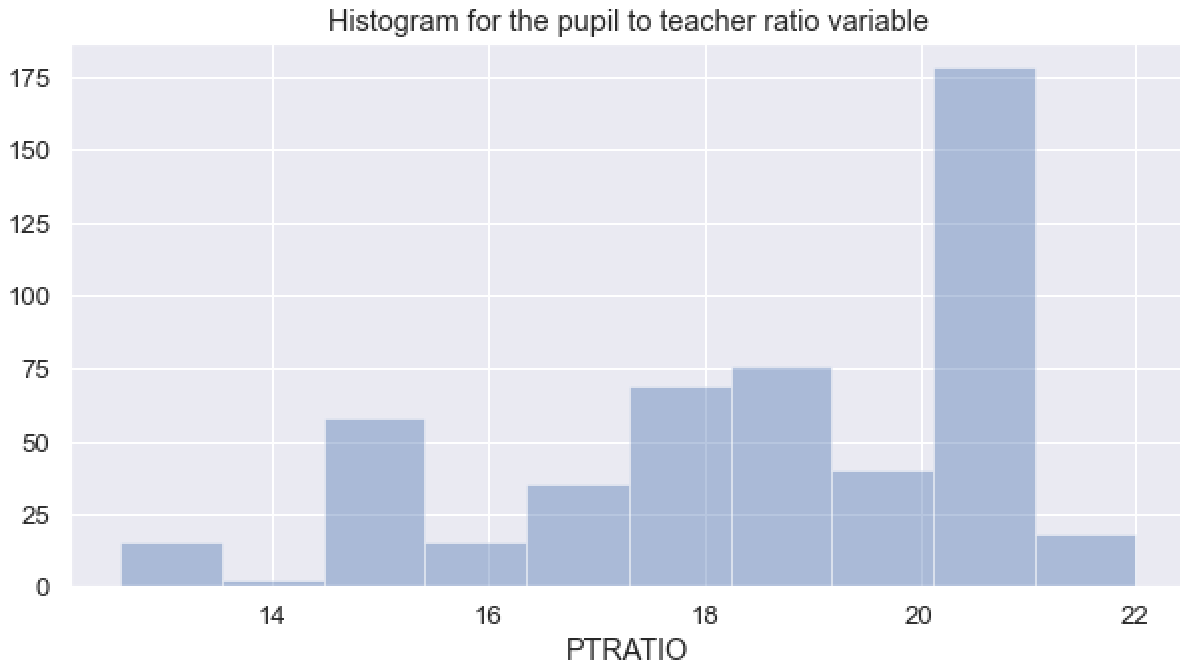
```
plt.figure(figsize=(10,5))
sns.boxplot(x=df.MEDV, y=df.age_group, data=df)
plt.title("Boxplot for the MEDV variable vs the AGE variable")
plt.show()
```



In [14]:
```
plt.figure(figsize=(10,5))
sns.scatterplot(x=df.NOX, y=df.INDUS, data=df)
plt.title("Relationship between NOX and INDUS")
plt.show()
```



In [15]:
```
plt.figure(figsize=(10,5))
sns.distplot(a=df.PTRATIO,bins=10, kde=False)
plt.title("Histogram for the pupil to teacher ratio variable")
plt.show()
```



Note: Pupil to teacher ratio is highest at 20-21 range.

Task 5

In [16]:
```
df
```

Out[16]:

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRAT
0	0.00632	18.0	2.31	0	0.538	6.575	65.2	4.0900	1	296	1!
1	0.02731	0.0	7.07	0	0.469	6.421	78.9	4.9671	2	242	1'
2	0.02729	0.0	7.07	0	0.469	7.185	61.1	4.9671	2	242	1'
3	0.03237	0.0	2.18	0	0.458	6.998	45.8	6.0622	3	222	1ε
4	0.06905	0.0	2.18	0	0.458	7.147	54.2	6.0622	3	222	1ε
...	...	...	...	...	...	...	...	...	...	...	
501	0.06263	0.0	11.93	0	0.573	6.593	69.1	2.4786	1	273	2:
502	0.04527	0.0	11.93	0	0.573	6.120	76.7	2.2875	1	273	2:
503	0.06076	0.0	11.93	0	0.573	6.976	91.0	2.1675	1	273	2:
504	0.10959	0.0	11.93	0	0.573	6.794	89.3	2.3889	1	273	2:
505	0.04741	0.0	11.93	0	0.573	6.030	80.8	2.5050	1	273	2:

506 rows × 14 columns

Is there a significant difference in median value of houses bounded by the Charles river or not? (T-test for independent samples)

Null Hypothesis(H_0): Both average MEDV are the same

Alternative Hypothesis(H_1): Both average MEDV are NOT the same

In [17]:
```
df["CHAS"].value_counts()
```

Out[17]:
```
0    471
1     35
Name: CHAS, dtype: int64
```

In [18]:
```
a = df[df["CHAS"] == 0]["MEDV"]
a
```

Out[18]:
```
0      24.0
1      21.6
2      34.7
3      33.4
4      36.2
       ...
501    22.4
502    20.6
503    23.9
504    22.0
505    11.9
Name: MEDV, Length: 471, dtype: float64
```

In [19]:
```
b = df[df["CHAS"] == 1]["MEDV"]
b
```

Out[19]:
```
142    13.4
152    15.3
154    17.0
155    15.6
160    27.0
162    50.0
163    50.0
208    24.4
209    20.0
210    21.7
211    19.3
212    22.4
216    23.3
218    21.5
219    23.0
220    26.7
221    21.7
222    27.5
234    29.0
236    25.1
269    20.7
273    35.2
274    32.4
276    33.2
277    33.1
282    46.0
283    50.0
356    17.8
357    21.7
358    22.7
363    16.8
364    21.9
369    50.0
370    50.0
372    50.0
Name: MEDV, dtype: float64
```

In [20]:
```
scipy.stats.ttest_ind(a,b,axis=0,equal_var=True)
```

Out[20]:
```
Ttest_indResult(statistic=-3.996437466090509, pvalue=7.390623170519905e-05)
```

Since p-value more than alpha value of 0.05, we failed to reject null hypothesis since there is NO statistical significance.

Is there a difference in Median values of houses (MEDV) for each proportion of owner occupied units built prior to 1940 (AGE)? (ANOVA)

In [21]:
```
df["AGE"].value_counts()
```

Out[21]:
```
100.0    43
96.0      4
98.2      4
95.4      4
97.9      4
         ..
47.6      1
92.7      1
13.9      1
58.4      1
40.1      1
Name: AGE, Length: 356, dtype: int64
```

In [22]:
```
df.loc[(df["AGE"] <= 35),'age_group'] = '35 years and younger'
df.loc[(df["AGE"] > 35) & (df["AGE"]<70),'age_group'] = 'between 35 and 70 y
df.loc[(df["AGE"] >= 70),'age_group'] = '70 years and older'
```

In [23]:
```
df
```

Out[23]:

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRAT
0	0.00632	18.0	2.31	0	0.538	6.575	65.2	4.0900	1	296	15
1	0.02731	0.0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17
2	0.02729	0.0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17
3	0.03237	0.0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18
4	0.06905	0.0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18
...	...	...	...	...	...	...	...	...	...	...	
501	0.06263	0.0	11.93	0	0.573	6.593	69.1	2.4786	1	273	21
502	0.04527	0.0	11.93	0	0.573	6.120	76.7	2.2875	1	273	21
503	0.06076	0.0	11.93	0	0.573	6.976	91.0	2.1675	1	273	21
504	0.10959	0.0	11.93	0	0.573	6.794	89.3	2.3889	1	273	21
505	0.04741	0.0	11.93	0	0.573	6.030	80.8	2.5050	1	273	21

506 rows × 14 columns

State the hypothesis

- $H_0 : \mu\_1 = \mu\_2 = \mu\_3$ (the three population means are equal)
- H_1 : At least one of the means differ

In [24]:
```
low = df[df["age_group"] == '35 years and younger']["MEDV"]
mid = df[df["age_group"] == 'between 35 and 70 years']["MEDV"]
high = df[df["age_group"] == '70 years and older']["MEDV"]
```

In [25]:
```
f_stats, p_value = scipy.stats.f_oneway(low,mid,high,axis=0)
```

In [26]:
```
print("F-Statistic={0}, P-value={1}".format(f_stats,p_value))
```

```
F-Statistic=36.40764999196599, P-value=1.7105011022702984e-15
```

Since p-value more than alpha value of 0.05, we failed to reject null hypothesis since there is NO statistical significance.

Can we conclude that there is no relationship between Nitric oxide concentrations and proportion of non-retail business acres per town? (Pearson Correlation)

State the hypothesis

- H_0 : NOX is not correlated with INDUS
- H_1 : NOX is correlated with INDUS

In [27]:
```
pearson,p_value = scipy.stats.pearsonr(df["NOX"],df["INDUS"])
```

In [28]:
```
print("Pearson Coefficient value={0}, P-value={1}".format(pearson,p_value))
```

```
Pearson Coefficient value=0.7636514469209154, P-value=7.913361061236894e-98
```

Since the p-value (Sig. (2-tailed) < 0.05, we reject the Null hypothesis and conclude that there exists a relationship between Nitric Oxide and non-retail business acres per town.

What is the impact of an additional weighted distance to the five Boston employment centres on the median value of owner occupied homes? (Regression analysis)

State Hypothesis

Null Hypothesis: weighted distances to five Boston employment centres are not related to median value

Alternative Hypothesis: weighted distances to five Boston employment centres are related to median value

In [29]:
```
df.columns
```

Out[29]:
```
Index(['CRIM', 'ZN', 'INDUS', 'CHAS', 'NOX', 'RM', 'AGE', 'DIS', 'RAD', 'TA
X', 'PTRATIO', 'LSTAT', 'MEDV', 'age_group'], dtype='object')
```

In [30]:
```
y = df['MEDV']
x = df['DIS']
```

In [31]:
```
x = sm.add_constant(x)
```

In [32]:
```
results = sm.OLS(y,x).fit()
```

In [33]:
```
results.summary()
```

Out[33]:

OLS Regression Results

Dep. Variable:	MEDV	**R-squared:**	0.062
Model:	OLS	**Adj. R-squared:**	0.061
Method:	Least Squares	**F-statistic:**	33.58
Date:	Tue, 03 Nov 2020	**Prob (F-statistic):**	1.21e-08
Time:	10:00:54	**Log-Likelihood:**	-1823.9
No. Observations:	506	**AIC:**	3652.
Df Residuals:	504	**BIC:**	3660.
Df Model:	1		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
const	18.3901	0.817	22.499	0.000	16.784	19.996
DIS	1.0916	0.188	5.795	0.000	0.722	1.462

Omnibus:	139.779	**Durbin-Watson:**	0.570
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	305.104
Skew:	1.466	**Prob(JB):**	5.59e-67
Kurtosis:	5.424	**Cond. No.**	9.32

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

In [34]:
```
np.sqrt(0.062)
```

Out[34]: 0.24899799195977465

The square root of R-squared is 0.25: implies weak correlation

Correlation

In [35]:
```
df.corr()
```

Out[35]:

	CRIM	ZN	INDUS	CHAS	NOX	RM	
CRIM	1.000000	-0.200469	0.406583	-0.055892	0.420972	-0.219247	0.352
ZN	-0.200469	1.000000	-0.533828	-0.042697	-0.516604	0.311991	-0.569
INDUS	0.406583	-0.533828	1.000000	0.062938	0.763651	-0.391676	0.644
CHAS	-0.055892	-0.042697	0.062938	1.000000	0.091203	0.091251	0.086
NOX	0.420972	-0.516604	0.763651	0.091203	1.000000	-0.302188	0.731
RM	-0.219247	0.311991	-0.391676	0.091251	-0.302188	1.000000	-0.240
AGE	0.352734	-0.569537	0.644779	0.086518	0.731470	-0.240265	1.000
DIS	-0.379670	0.664408	-0.708027	-0.099176	-0.769230	0.205246	-0.747
RAD	0.625505	-0.311948	0.595129	-0.007368	0.611441	-0.209847	0.456
TAX	0.582764	-0.314563	0.720760	-0.035587	0.668023	-0.292048	0.506
PTRATIO	0.289946	-0.391679	0.383248	-0.121515	0.188933	-0.355501	0.261
LSTAT	0.455621	-0.412995	0.603800	-0.053929	0.590879	-0.613808	0.602
MEDV	-0.388305	0.360445	-0.483725	0.175260	-0.427321	0.695360	-0.376

In [36]:

```
plt.figure(figsize=(16,9))
sns.heatmap(df.corr(),cmap="coolwarm",annot=True,fmt='.2f',linewidths=2, cba
plt.show()
```