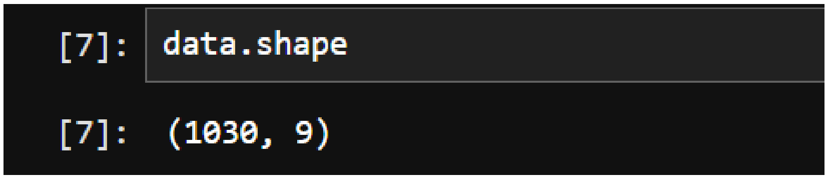
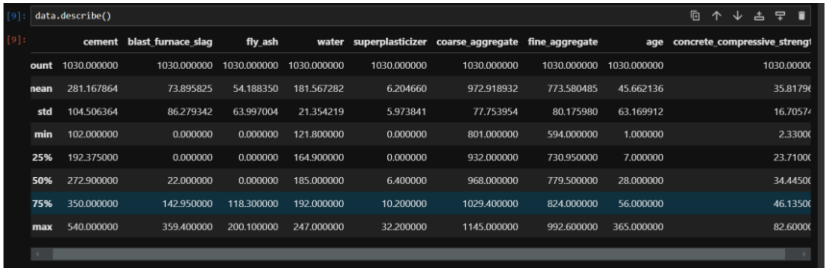
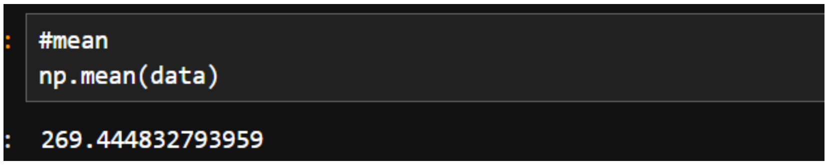
Data Collection and Preprocessing Phase

Date	9 July 2024
Team ID	SWTID1720162737
Project Title	Predicting Compressive Strength Of Concrete Using Machine Learning.
Maximum Marks	6 Marks

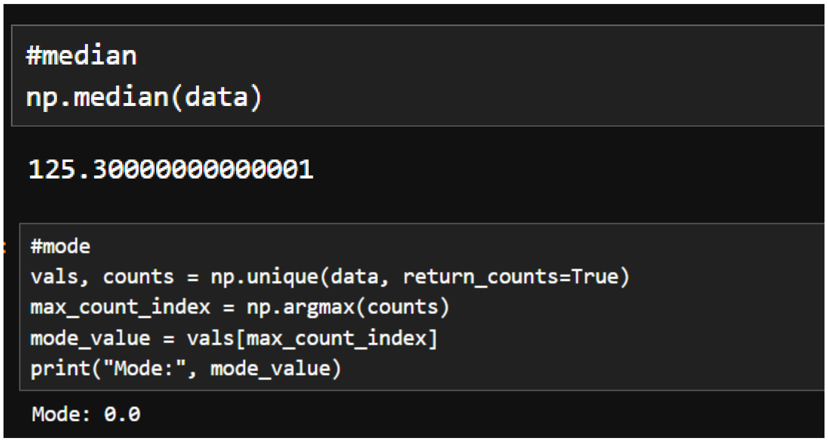
Data Exploration and Preprocessing Template

Identifies data sources, assesses quality issues like missing values and duplicates, and implements resolution plans to ensure accurate and reliable analysis.

Section	Description
Data Overview	**Dimensions:**  [7]: data.shape [7]: (1030, 9) **Descriptive statistics:**  [9]: data.describe()

Univariate Analysis	
	: #mean
	np.mean(data)
	: 269.444832793959

```
[9]:       cement  blast_furnace_slag      fly_ash        water  superplasticizer  coarse_aggregate  fine_aggregate          age  concrete_compressive_strengt
ount  1030.000000         1030.000000  1030.000000  1030.000000       1030.000000       1030.000000     1030.000000  1030.000000                    1030.0000(
nean   281.167864           73.895825    54.188350   181.567282          6.204660        972.918932      773.580485    45.662136                      35.8179(
 std   104.506364           86.279342    63.997004    21.354219          5.973841         77.753954       80.175980    63.169912                      16.7057(
 min   102.000000            0.000000     0.000000   121.800000          0.000000        801.000000      594.000000     1.000000                       2.3300(
 25%   192.375000            0.000000     0.000000   164.900000          0.000000        932.000000      730.950000     7.000000                      23.7100(
 50%   272.900000           22.000000     0.000000   185.000000          6.400000        968.000000      779.500000    28.000000                      34.4450(
 75%   350.000000          142.950000   118.300000   192.000000         10.200000       1029.400000      824.000000    56.000000                      46.1350(
 max   540.000000          359.400000   200.100000   247.000000         32.200000       1145.000000      992.600000   365.000000                      82.6000(
```



```
#median
np.median(data)
```

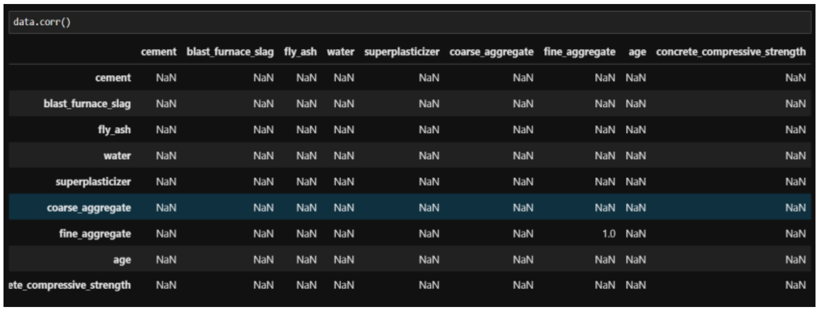
```
125.30000000000001
```

```
#mode
vals, counts = np.unique(data, return_counts=True)
max_count_index = np.argmax(counts)
mode_value = vals[max_count_index]
print("Mode:", mode_value)
```

```
Mode: 0.0
```

Bivariate Analysis

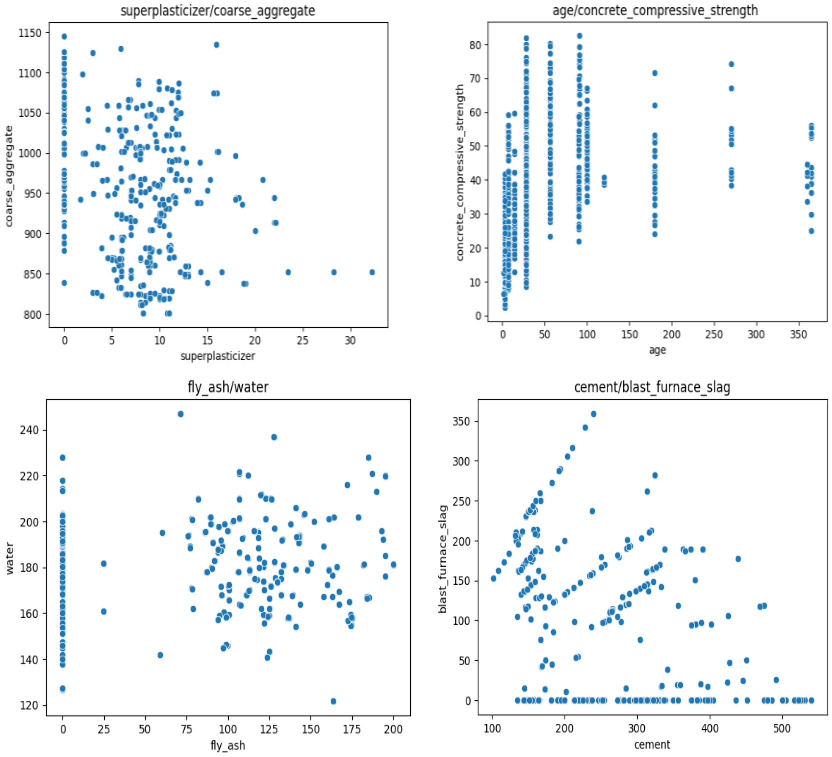
Correlation:

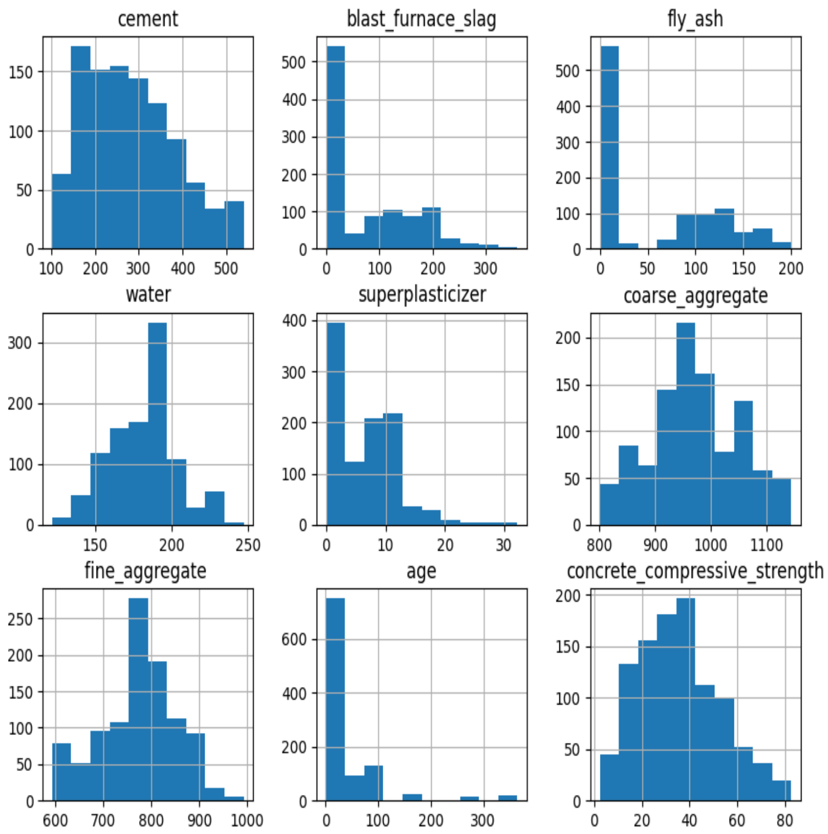


```
data.corr()
```

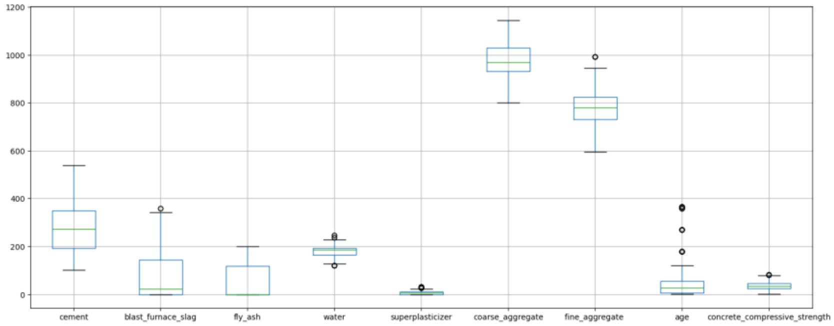
	cement	blast_furnace_slag	fly_ash	water	superplasticizer	coarse_aggregate	fine_aggregate	age	concrete_compressive_strength
cement	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
blast_furnace_slag	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
fly_ash	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
water	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
superplasticizer	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
coarse_aggregate	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
fine_aggregate	NaN	NaN	NaN	NaN	NaN	NaN	1.0	NaN	NaN
age	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
ete_compressive_strength	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

Scatter plots:

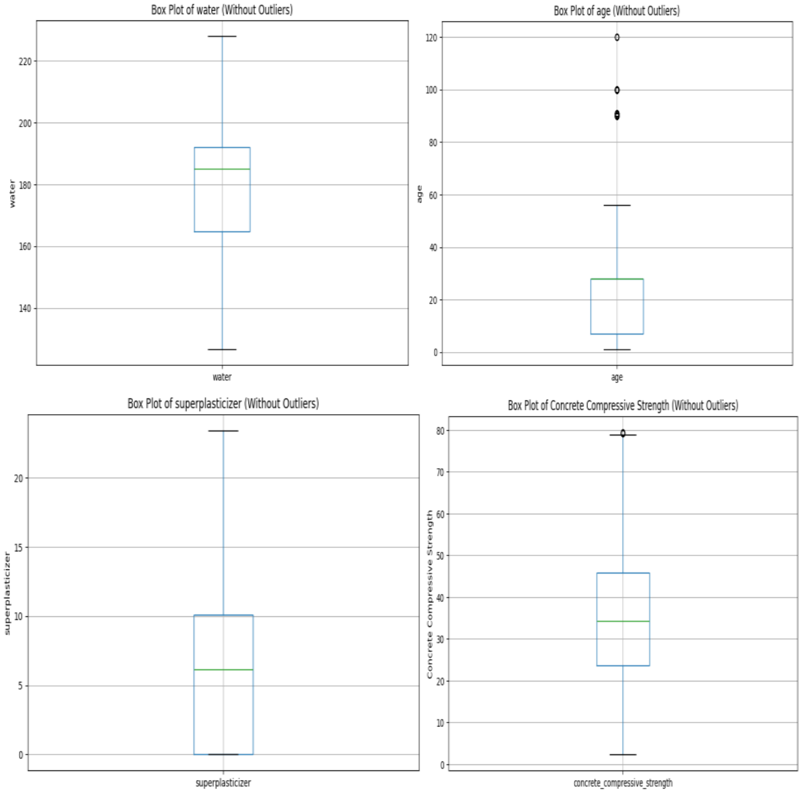


Multivariate Analysis	

<table>
<tr>
<td>Outliers and Anomalies</td>
<td>
With Outliers

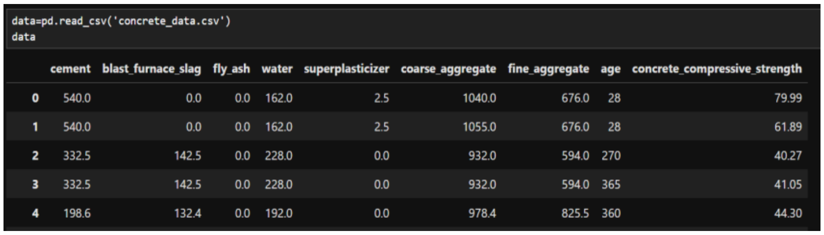


Without outliers:



</td>
</tr>
<tr>
<td colspan="2"><b>Data Preprocessing Code Screenshots</b></td>
</tr>
</table>

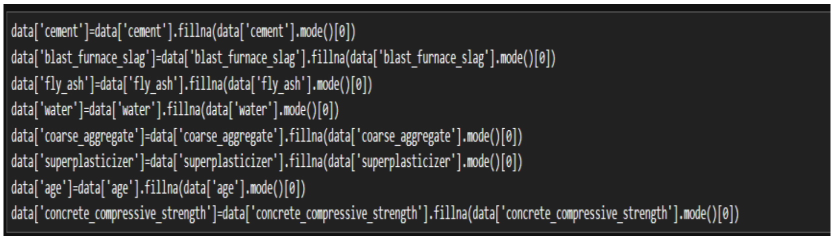
<table>
<tr>
<td>Loading Data</td>
<td>



```
data=pd.read_csv('concrete_data.csv')
data
```

```
   cement  blast_furnace_slag  fly_ash  water  superplasticizer  coarse_aggregate  fine_aggregate  age  concrete_compressive_strength
0   540.0                 0.0      0.0  162.0               2.5            1040.0           676.0   28                          79.99
1   540.0                 0.0      0.0  162.0               2.5            1055.0           676.0   28                          61.89
2   332.5               142.5      0.0  228.0               0.0             932.0           594.0  270                          40.27
3   332.5               142.5      0.0  228.0               0.0             932.0           594.0  365                          41.05
4   198.6               132.4      0.0  192.0               0.0             978.4           825.5  360                          44.30
```

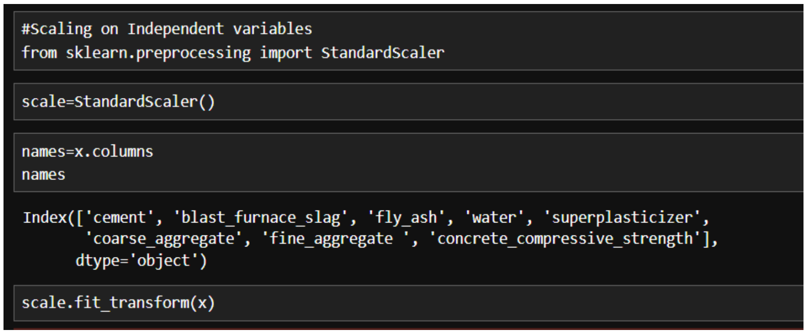
</td>
</tr>
<tr>
<td>Handling Missing Data</td>
<td>



```
data['cement']=data['cement'].fillna(data['cement'].mode()[0])
data['blast_furnace_slag']=data['blast_furnace_slag'].fillna(data['blast_furnace_slag'].mode()[0])
data['fly_ash']=data['fly_ash'].fillna(data['fly_ash'].mode()[0])
data['water']=data['water'].fillna(data['water'].mode()[0])
data['coarse_aggregate']=data['coarse_aggregate'].fillna(data['coarse_aggregate'].mode()[0])
data['superplasticizer']=data['superplasticizer'].fillna(data['superplasticizer'].mode()[0])
data['age']=data['age'].fillna(data['age'].mode()[0])
data['concrete_compressive_strength']=data['concrete_compressive_strength'].fillna(data['concrete_compressive_strength'].mode()[0])
```

But it is not necessary cause we don’t have any missed values in the data set.

</td>
</tr>
<tr>
<td>Data Transformation</td>
<td>



```
#Scaling on Independent variables
from sklearn.preprocessing import StandardScaler
```

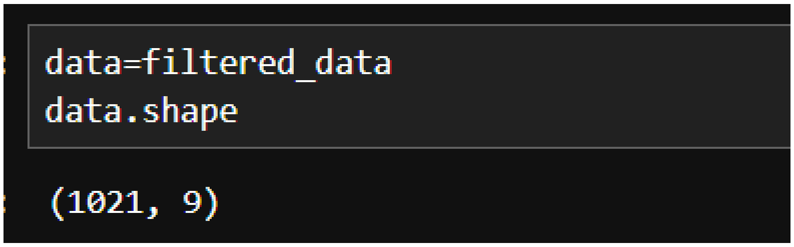
```
scale=StandardScaler()
```

```
names=x.columns
names
```

```
Index(['cement', 'blast_furnace_slag', 'fly_ash', 'water', 'superplasticizer',
       'coarse_aggregate', 'fine_aggregate ', 'concrete_compressive_strength'],
      dtype='object')
```

```
scale.fit_transform(x)
```

</td>
</tr>
<tr>
<td>Feature Engineering</td>
<td>Attached the codes in final submission.</td>
</tr>
<tr>
<td>Save Processed Data</td>
<td>



```
data=filtered_data
data.shape
```

```
(1021, 9)
```

</td>
</tr>
</table>