Data Collection and Preprocessing Phase
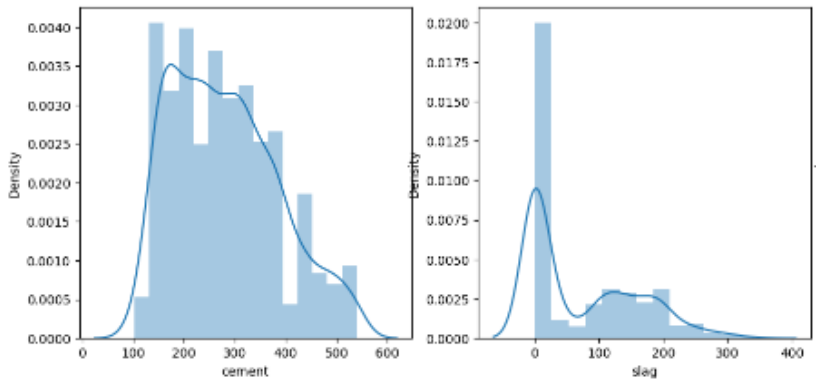
Date	15 March 2024
Team ID	SWTID1720113374
Project Title	Predicting Compressive Strength Of Concrete Using Machine Learning
Maximum Marks	6 Marks

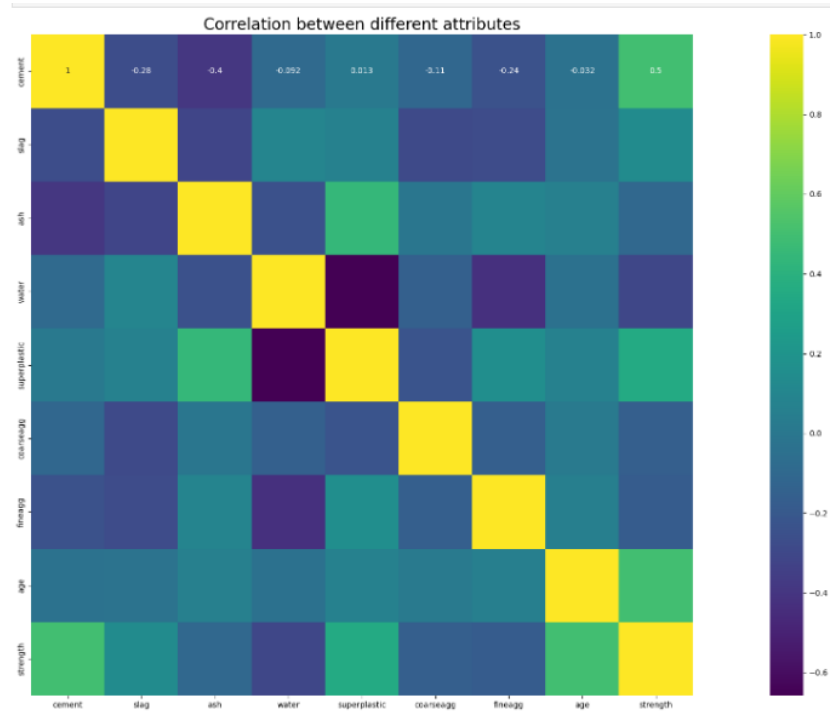**Data Exploration and Preprocessing Template**

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	

```
data.dtypes

cement          float64
slag            float64
ash             float64
water           float64
superplastic    float64
coarseagg       float64
fineagg         float64
age               int64
strength        float64
dtype: object

data.shape

(1030, 9)

data.info

<bound method DataFrame.info of      cement   slag    ash   water  superplastic  coarseagg  fineagg  age  \
0     540.0    0.0    0.0  162.0           2.5     1040.0    676.0   28
1     540.0    0.0    0.0  162.0           2.5     1055.0    676.0   28
2     332.5  142.5    0.0  228.0           0.0      932.0    594.0  270
3     332.5  142.5    0.0  228.0           0.0      932.0    594.0  365
4     198.6  132.4    0.0  192.0           0.0      978.4    825.5  360
...     ...    ...    ...    ...           ...        ...      ...  ...
1025  276.4  116.0   90.3  179.6           8.9      870.1    768.3   28
1026  322.2    0.0  115.6  196.0          10.4      817.9    813.4   28
1027  148.5  139.4  108.6  192.7           6.1      892.4    780.0   28
1028  159.1  186.7    0.0  175.6          11.3      989.6    788.9   28
1029  260.9  100.5   78.3  200.6           8.6      864.5    761.5   28

      strength
0        79.99
1        61.89
2        40.27
3        41.05
```
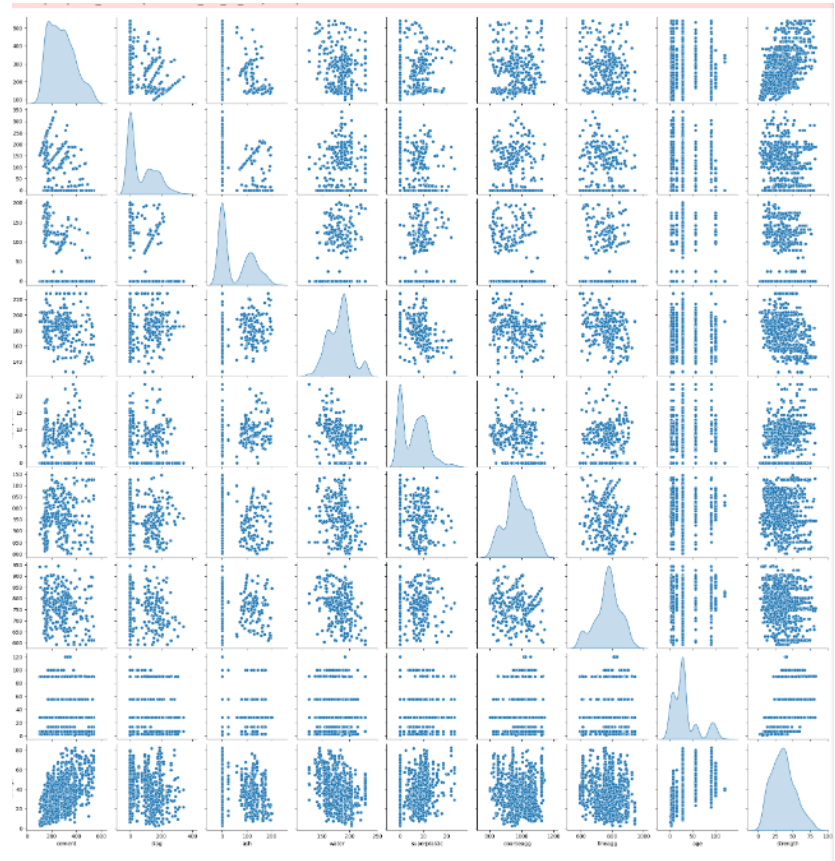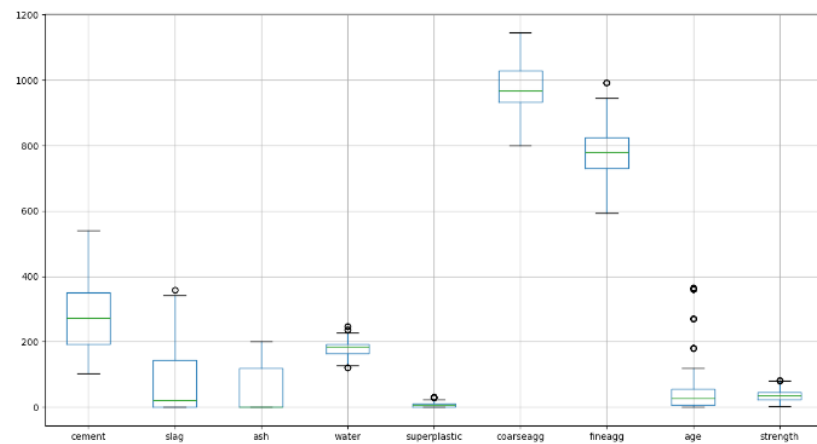
Univariate Analysis	
Bivariate Analysis	

Multivariate Analysis	
Outliers and Anomalies	

Data Preprocessing Code Screenshots

Loading Data	**Reading The Dataset** `data=pd.read_csv("concrete_data.csv")`

Handling Missing Data	No missing attributes

Data Transformation	**Scaling ¶**
	
	``python
	from sklearn.preprocessing import StandardScaler
	
	scaler = StandardScaler()
	
	# Fit on training set only.
	scaler.fit(X_train)
	
	# Apply transform to both the training set and the test set.
	X_train_transformed = scaler.transform(X_train)
	X_test_transoformed = scaler.transform(X_test)
	``

Feature Engineering	Attached the codes in final submission.
Save Processed Data	Done