# COLLEGE CODE : 8100

# REGISTER NO :810021205057

**WEBSITE TRAFFIC ANALYSIS**

**DATA ANALYTICS WITH COGNOS : GROUP 5**

**PHASE : 3**

This phase involves in designing of the steps that defining in each phase of the previous documentation this involves importing necessary functions, data processing and so on in this phase we have to begin our project by loading and preprocessing the dataset.

The IBM suggests using the jupyter notebook for loading and preprocess the dataset:

Here for this project title we need to define the loading the libraries, understand the data and visualize the missing values.

For this certain inputs are defined for this project.in this phase each of the input

Codes of project is given below:

# untitled7

October 18, 2023

[ ]:

PHASE

3

[1]:

**import**

**pandas**

**as**

**pd**

**import**

**numpy**

**as**

**np**

**import**

**missingno**

**as**

**msno**

[2]:

df

=

pd

.

read\_csv(

'

daily-website-visitors.csv

'

)

[3]:

df

.

head()

[3]:

Row

Day Day.Of.Week

Date Page.Loads Unique.Visits \

1. 1 Sunday 1 9/14/2014 2,146 1,582
2. 2 Monday 2 9/15/2014 3,621 2,528
3. 3 Tuesday 3 9/16/2014 3,698 2,630
4. 4 Wednesday 4 9/17/2014 3,667 2,614
5. 5 Thursday 5 9/18/2014 3,316 2,366

First.Time.Visits Returning.Visits

1. 1,430 152
2. 2,297 231
3. 2,352 278
4. 2,327 287
5. 2,130 236

[4]:

df

.

tail()

[4]: Row Day Day.Of.Week Date Page.Loads Unique.Visits \

2162 2163 Saturday 7 8/15/2020 2,221 1,696 2163 2164 Sunday 1 8/16/2020 2,724 2,037

1. 2165 Monday 2 8/17/2020 3,456 2,638
2. 2166 Tuesday 3 8/18/2020 3,581 2,683
3. 2167 Wednesday 4 8/19/2020 2,064 1,564

First.Time.Visits Returning.Visits

1. 1,373 323
2. 1,686 351
3. 2,181 457

|  |  |
| --- | --- |
| 2165 2,184 | 499 |
| 2166 1,297 | 267 |

[5]:

df

.

shape

[5]: (2167, 8)

[6]:

df

.

info()

<class 'pandas.core.frame.DataFrame'> RangeIndex: 2167 entries, 0 to 2166 Data columns (total 8 columns):

|  |  |
| --- | --- |
| # Column Non-Null Count Dtype  --- ------ -------------- ----- | |
| 0 Row 2167 non-null | int64 |
| 1 Day 2167 non-null | object |
| 2 Day.Of.Week 2167 non-null | int64 |
| 3 Date 2167 non-null | object |
| 4 Page.Loads 2167 non-null | object |
| 5 Unique.Visits 2167 non-null | object |
| 6 First.Time.Visits 2167 non-null | object |
| 7 Returning.Visits 2167 non-null dtypes: int64(2), object(6) memory usage: 135.6+ KB | object |

[7]:

df

.

columns

.

values

[7]: array(['Row', 'Day', 'Day.Of.Week', 'Date', 'Page.Loads', 'Unique.Visits',

'First.Time.Visits', 'Returning.Visits'], dtype=object)

[8]:

df

.

dtypes

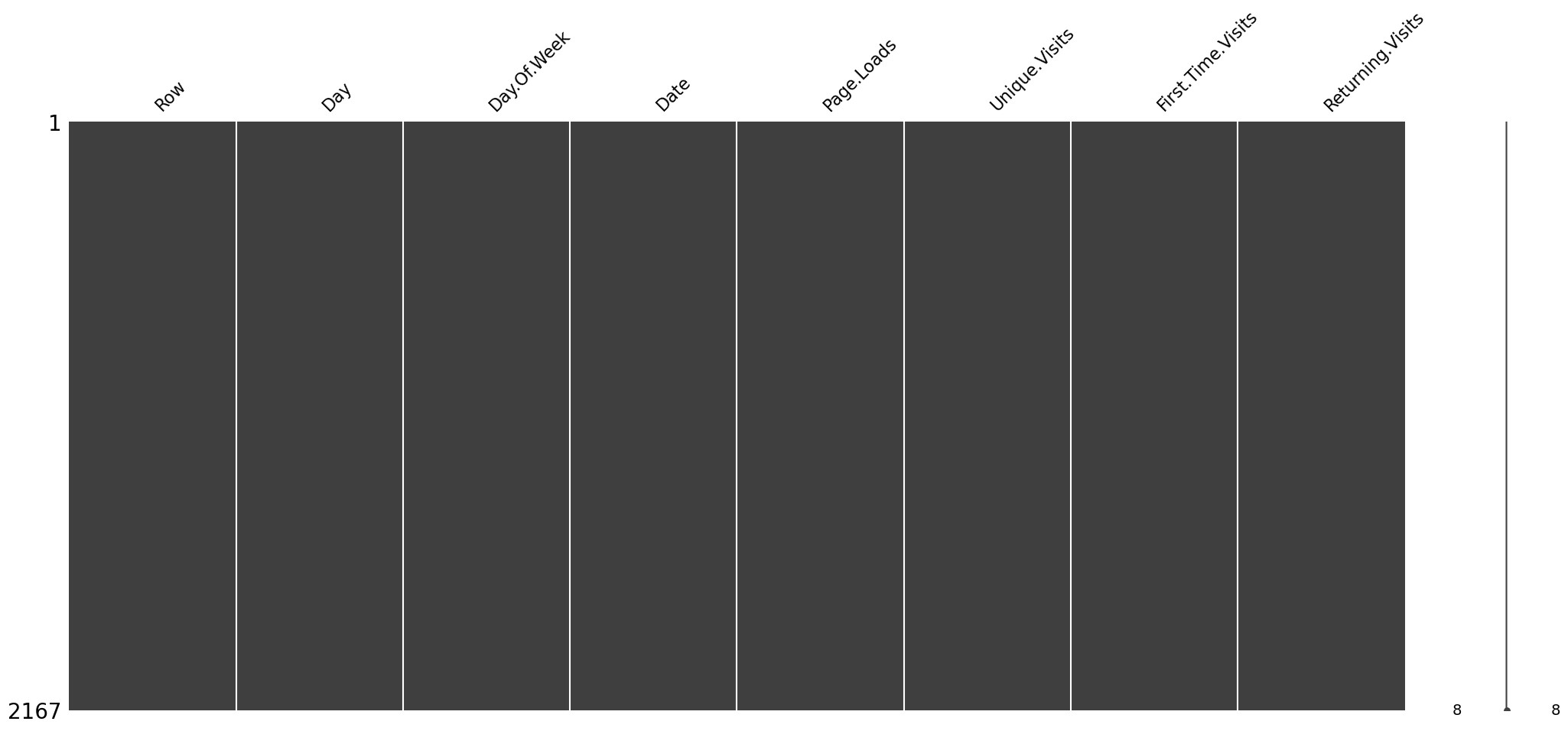
|  |  |
| --- | --- |
| [8]: Row | int64 |
| Day | object |
| Day.Of.Week | int64 |
| Date | object |
| Page.Loads | object |
| Unique.Visits | object |
| First.Time.Visits | object |
| Returning.Visits dtype: object | object |

[9]:

msno

.

matrix(df);



[10]:

df

=

df

.

drop([

'

Unique.Visits

'

]

,axis

=

1

)

df

.

head()

[10]: Row Day Day.Of.Week Date Page.Loads First.Time.Visits \

1. 1 Sunday 1 9/14/2014 2,146 1,430
2. 2 Monday 2 9/15/2014 3,621 2,297
3. 3 Tuesday 3 9/16/2014 3,698 2,352
4. 4 Wednesday 4 9/17/2014 3,667 2,327
5. 5 Thursday 5 9/18/2014 3,316 2,130

Returning.Visits

1. 152
2. 231
3. 278
4. 287
5. 236

[11]:

df

.

isnull()

[11]: Row Day Day.Of.Week Date Page.Loads First.Time.Visits \

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0 False False |  | False False | False |  | False |
| 1 False False |  | False False | False |  | False |
| 2 False False |  | False False | False |  | False |
| 3 False False |  | False False | False |  | False |
| 4 False False |  | False False | False |  | False |
| … … … | … | … … |  | … |  |
| 2162 False False |  | False False | False |  | False |
| 2163 False False |  | False False | False |  | False |
| 2164 False False |  | False False | False |  | False |
| 2165 False False | | False False | False | | False |
| 2166 False False | | False False | False | | False |

Returning.Visits

1. False
2. False
3. False
4. False
5. False

… …

1. False
2. False
3. False
4. False
5. False

[2167 rows x 7 columns]

[12]:

df

.

isnull()

.

sum()

|  |  |
| --- | --- |
| [12]: Row | 0 |
| Day | 0 |
| Day.Of.Week | 0 |
| Date | 0 |
| Page.Loads | 0 |
| First.Time.Visits | 0 |
| Returning.Visits dtype: int64 | 0 |

[13]:

df[

'

Row

'

]

=

pd

.

to\_numeric(df

.

Row,errors

=

'

coerce

'

)

df

.

isnull()

.

sum()

|  |  |
| --- | --- |
| [13]: Row | 0 |
| Day | 0 |
| Day.Of.Week | 0 |
| Date | 0 |
| Page.Loads | 0 |
| First.Time.Visits | 0 |
| Returning.Visits dtype: int64 | 0 |

[14]:

df[np

.

isnan(df[

'

Row

'

])]

[14]: Empty DataFrame

Columns: [Row, Day, Day.Of.Week, Date, Page.Loads, First.Time.Visits,

Returning.Visits]

Index: []

[15]:

df

.

fillna(df[

'

Row

'

]

.

mean())

[15]: Row Day Day.Of.Week Date Page.Loads First.Time.Visits \

1. 1 Sunday 1 9/14/2014 2,146 1,430
2. 2 Monday 2 9/15/2014 3,621 2,297
3. 3 Tuesday 3 9/16/2014 3,698 2,352
4. 4 Wednesday 4 9/17/2014 3,667 2,327
5. 5 Thursday 5 9/18/2014 3,316 2,130

… … … … … … …

1. 2163 Saturday 7 8/15/2020 2,221 1,373
2. 2164 Sunday 1 8/16/2020 2,724 1,686
3. 2165 Monday 2 8/17/2020 3,456 2,181
4. 2166 Tuesday 3 8/18/2020 3,581 2,184
5. 2167 Wednesday 4 8/19/2020 2,064 1,297

Returning.Visits

1. 152
2. 231
3. 278
4. 287
5. 236

… …

1. 323
2. 351
3. 457
4. 499
5. 267

[2167 rows x 7 columns]

[16]:

df[

"

Date

"

]

=

pd

.

to\_datetime(df[

"

Date

"

]

,

format

=

"

%

m/

**%d**

/

%

Y

"

)

print

(

df

.

info())

<class 'pandas.core.frame.DataFrame'> RangeIndex: 2167 entries, 0 to 2166 Data columns (total 7 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

1. Row 2167 non-null int64
2. Day 2167 non-null object
3. Day.Of.Week 2167 non-null int64
4. Date 2167 non-null datetime64[ns]
5. Page.Loads 2167 non-null object 5 First.Time.Visits 2167 non-null object 6 Returning.Visits 2167 non-null object dtypes: datetime64[ns](1), int64(2), object(4) memory usage: 118.6+ KB

None

[17]:

df

.

isnull()

.

sum()

|  |  |
| --- | --- |
| [17]: Row | 0 |
| Day | 0 |
| Day.Of.Week | 0 |
| Date | 0 |
| Page.Loads | 0 |
| First.Time.Visits | 0 |
| Returning.Visits dtype: int64 | 0 |

[18]: df["Returning.Visits"]=df['Returning.Visits'].map({0:"no", 1: "yes"}) df.head()

[18]: Row Day Day.Of.Week Date Page.Loads First.Time.Visits \

1. 1 Sunday 1 2014-09-14 2,146 1,430
2. 2 Monday 2 2014-09-15 3,621 2,297
3. 3 Tuesday 3 2014-09-16 3,698 2,352
4. 4 Wednesday 4 2014-09-17 3,667 2,327
5. 5 Thursday 5 2014-09-18 3,316 2,130

Returning.Visits

1. NaN
2. NaN
3. NaN
4. NaN
5. NaN

[19]: df["Returning.Visits"].describe(include=['object','bool'])

|  |  |
| --- | --- |
| [19]: count | 0 |
| unique | 0 |
| top | NaN |
| freq | NaN |

Name: Returning.Visits, dtype: object

[20]:

df[df[

'

Row

'

]

==

0

]

.

index

[20]: Int64Index([], dtype='int64')

[21]: numerical\_cols = ['Row','First.Time.Visits','Returning.Visits'] df[numerical\_cols].describe()

[21]: Row count 2167.000000

|  |  |
| --- | --- |
| mean | 1084.000000 |
| std | 625.703338 |
| min | 1.000000 |
| 25% | 542.500000 |
| 50% | 1084.000000 |
| 75% | 1625.500000 |
| max | 2167.000000 |

[ ]:

Accuracy :0.1726268

Consistency:23.28393