

```
[2]: # Pandas is two dimensional datastructure

[3]: import pandas as pd
import numpy as np

[4]: print(dir(pd),end="")

['ArrowDtype', 'BooleanDtype', 'Categorical', 'CategoricalDtype', 'CategoricalIndex', 'DataFrame', 'DateOffset', 'DatetimeIndex', 'DatetimeTZDtype', 'ExcelFile', 'ExcelWriter', 'Flags', 'Float32Dtype', 'Float64Dtype', 'Grouper', 'HDFStore', 'Index', 'IndexSlice', 'Int16Dtype', 'Int32Dtype', 'Int64Dtype', 'Int8Dtype', 'Interval', 'IntervalDtype', 'IntervalIndex', 'MultiIndex', 'NA', 'NaN', 'NamedAgg', 'Period', 'PeriodDtype', 'PeriodIndex', 'RangeIndex', 'Series', 'SparseDtype', 'StringDtype', 'Timedelta', 'TimedeltaIndex', 'Timestamp', 'UInt16Dtype', 'UInt32Dtype', 'UInt64Dtype', 'UInt8Dtype', '_all__', '_builtins__', '_cached__', '_dot__', '_docformat__', '_file__', '_git_version__', '_loader__', '_name__', '_package__', '_path__', '_spec__', '_version__', '_built_with_meson', '_config', '_is_numpy_dev', '_libs', '_pandas_datetime_CAPI', '_pandas_parser_CAPI', '_testing', '_typing', '_version_meson', 'annotations', 'api', 'array', 'arrays', 'bdate_range', 'compat', 'concat', 'core', 'crosstab', 'cut', 'date_range', 'describe_option', 'errors', 'eval', 'factorize', 'from_dummies', 'get_dummies', 'get_option', 'infer_freq', 'interval_range', 'io', 'isna', 'isnull', 'json_normalize', 'lreshape', 'melt', 'merge', 'merge_asof', 'merge_ordered', 'notna', 'notnull', 'offsets', 'option_context', 'options', 'pandas', 'period_range', 'pivot', 'pivot_table', 'plotting', 'qcut', 'read_clipboard', 'read_csv', 'read_excel', 'read_feather', 'read_fwf', 'read_gbq', 'read_hdf', 'read_html', 'read_json', 'read_orc', 'read_parquet', 'read_pickle', 'read_sas', 'read_spss', 'read_sql', 'read_sql_query', 'read_sql_table', 'read_stata', 'read_table', 'read_xml', 'reset_option', 'set_eng_float_format', 'set_option', 'show_versions', 'test', 'testing', 'timedelta_range', 'to_datetime', 'to_numeric', 'to_pickle', 'to_timedelta', 'tseries', 'unique', 'util', 'value_counts', 'wide_to_long']

[5]: a=[[10,20,30,40],[50,60,70,80]]

[6]: a

[6]: [[10, 20, 30, 40], [50, 60, 70, 80]]

[7]: type(a)

[7]: list

[8]: pd.DataFrame(a)

[8]:   0  1  2  3
 0  10 20 30 40
 1  50 60 70 80

[9]: b=pd.Series([[10, 20, 30, 40], [50, 60, 70, 80]])

[10]: pd.DataFrame(b)

[10]:   0
 0  [10, 20, 30, 40]
 1  [50, 60, 70, 80]

[11]: c=np.array([[10, 20, 30, 40], [50, 60, 70, 80]])

[12]: c

[12]: array([[10, 20, 30, 40],
       [50, 60, 70, 80]])

[13]: pd.DataFrame(c)

[13]:   0  1  2  3
 0  10 20 30 40
 1  50 60 70 80

[14]: d = {"roll_no": ["101", "102", "103", "104"], "name": ["chandru", "rohit", "Ashish", "Devdata"], "maths": [72, 88, 56, 59], "science": [73, 89, 57, 60]}

[15]: d

[15]: {'roll_no': ['101', '102', '103', '104'], 'name': ['chandru', 'rohit', 'Ashish', 'Devdata'], 'maths': [72, 88, 56, 59], 'science': [73, 89, 57, 60]}

[16]: pd.DataFrame(d)

[16]:   roll_no    name  maths  science
 0      101  chandru     72      73
 1      102    rohit     88      89
 2      103    Ashish     56      57
 3      104   Devdata     59      60

[309]: #importing data from excel

data = pd.read_excel(r'C:\Users\CHANDRASHEKAR\Downloads\student_pbi.xlsx')

[311]: student = pd.DataFrame(data)

[57]: #select * from student
student

[57]:   ROLL_NO    NAME    DEPT  MATHS  SCIENCE  SOCIAL  KANNADA  ENGLISH  HINDI
 0      101  RAMESH  ANALYTICS     70      58      67      92      36      60
 1      102  SURESH      MBA     87      96      43      42      69      68
```

	103	GANESH		BBA	46	86	68	71	72	79
3	104	CHANDRU	ANALYTICS	MBA	58	74	91	38	73	88
4	105	SOMU		MBA	72	61	81	47	80	97
5	106	SHIVU		BBA	46	41	41	94	53	69
6	107	VINOD	ANALYTICS	MBA	43	45	74	62	62	93
7	108	ROOPA		MBA	62	58	70	64	63	38
8	109	VIGNESH		BBA	63	72	93	85	67	98
9	110	VYLESH	ANALYTICS	MBA	64	47	95	71	50	58
10	111	KUMAR		MBA	46	65	74	94	92	43
11	112	SAKSHI		BBA	96	58	78	66	51	72
12	113	ANIL	ANALYTICS	MBA	68	66	61	96	38	67
13	114	SUJITH		MBA	75	84	39	71	69	39
14	115	KRISHNA		BBA	51	84	74	61	63	73
15	116	VASANTH	ANALYTICS	MBA	96	58	78	66	51	72
16	117	QUEEN	ANALYTICS	MBA	68	66	61	96	38	67

```
[59]: #select * from student order by roll_no asc limit 3  
student.head(3)
```

[59]:	ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI
0	101	RAMESH	ANALYTICS	70	58	67	92	36	60
1	102	SURESH	MBA	87	96	43	42	69	68
2	103	GANESH	BBA	46	86	68	71	72	79

```
[61]: #select * from student order by roll_no desc limit 3  
student.tail(3)
```

[61]:	ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI
14	115	KRISHNA	BBA	51	84	74	61	63	73
15	116	VASANTH	ANALYTICS	96	58	78	66	51	72
16	117	QUEEN	ANALYTICS	68	66	61	96	38	67

```
[63]: student.columns
```

```
[63]: Index(['ROLL_NO', 'NAME', 'DEPT', 'MATHS', 'SCIENCE', 'SOCIAL', 'KANNADA',
           'ENGLISH', 'HINDI'],
          dtype='object')
```

```
[65]: student.shape
```

[65]: (17, 9)

```
• [67]: #describe student
          student.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 17 entries, 0 to 16
Data columns (total 9 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 0   ROLL_NO    17 non-null    int64  
 1   NAME        17 non-null    object  
 2   DEPT       17 non-null    object  
 3   MATHS      17 non-null    int64  
 4   SCIENCE    17 non-null    int64  
 5   SOCIAL     17 non-null    int64  
 6   KANIADA    17 non-null    int64  
 7   ENGLISH    17 non-null    int64  
 8   HINDI      17 non-null    int64  
dtypes: int64(7), object(2)
memory usage: 1.3+ KB
```

```
[69]: student.describe()
```

[69]:	ROLL_NO	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI
count	17.000000	17.000000	17.000000	17.000000	17.000000	17.000000	17.000000
mean	109.000000	65.352941	65.823529	69.882353	71.529412	60.411765	69.470588
std	5.049752	16.624309	15.322513	16.951705	18.917973	15.455819	18.439488
min	101.000000	43.000000	41.000000	39.000000	38.000000	36.000000	38.000000
25%	105.000000	51.000000	58.000000	61.000000	62.000000	51.000000	60.000000
50%	109.000000	64.000000	65.000000	74.000000	71.000000	63.000000	69.000000
75%	113.000000	72.000000	74.000000	78.000000	92.000000	69.000000	79.000000
max	117.000000	96.000000	96.000000	95.000000	96.000000	92.000000	98.000000

```
[85]: # select NAME, DEPT, MATHS from students
student[["NAME", "DEPT", "MATHS"]][0:10:3]
```

[85]:	NAME	DEPT	MATHS
0	RAMESH	ANALYTICS	70
3	CHANDRU	ANALYTICS	58
6	VINOD	ANALYTICS	43
9	WVLESH	ANALYTICS	64

```
[115]: #lock functions  
student.loc[0:10,['NAME','ENGLISH','MATHS']]
```

	NAME	ENGLISH	MATHS
0	RAMESH	36	70
1	SURESH	69	87
2	GANESH	72	46
3	CHANDRU	73	58
4	SOMU	80	72
5	SHIVU	53	46
6	VINOD	62	43
7	ROOPA	63	62
8	VIGNESH	67	63
9	VYLESH	50	64
10	KUMAR	92	46

```
[125]: #index location
student.iloc[5:10,1:5]
```

	NAME	DEPT	MATHS	SCIENCE
5	SHIVU	BBA	46	41
6	VINOD	ANALYTICS	43	45
7	ROOPA	MBA	62	58
8	VIGNESH	BBA	63	72
9	VYLESH	ANALYTICS	64	47

```
[133]: student["TOTAL"] = student["MATHS"]+student["SCIENCE"]+student["SOCIAL"]+student["KANNADA"]+student["ENGLISH"]+student["HINDI"]
```

```
[145]: student["PERCENT"] = round(student["TOTAL"]/600*100,2)
```

```
[147]: student
```

	ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	TOTAL	PERCENT
0	101	RAMESH	ANALYTICS	70	58	67	92	36	60	383	63.83
1	102	SURESH	MBA	87	96	43	42	69	68	405	67.50
2	103	GANESH	BBA	46	86	68	71	72	79	422	70.33
3	104	CHANDRU	ANALYTICS	58	74	91	38	73	88	422	70.33
4	105	SOMU	MBA	72	61	81	47	80	97	438	73.00
5	106	SHIVU	BBA	46	41	41	94	53	69	344	57.33
6	107	VINOD	ANALYTICS	43	45	74	62	62	93	379	63.17
7	108	ROOPA	MBA	62	58	70	64	63	38	355	59.17
8	109	VIGNESH	BBA	63	72	93	85	67	98	478	79.67
9	110	VYLESH	ANALYTICS	64	47	95	71	50	58	385	64.17
10	111	KUMAR	MBA	46	65	74	94	92	43	414	69.00
11	112	SAKSHI	BBA	96	58	78	66	51	72	421	70.17
12	113	ANIL	ANALYTICS	68	66	61	96	38	67	396	66.00
13	114	SUJITH	MBA	75	84	39	71	69	39	377	62.83
14	115	KRISHNA	BBA	51	84	74	61	63	73	406	67.67
15	116	VASANTH	ANALYTICS	96	58	78	66	51	72	421	70.17
16	117	QUEEN	ANALYTICS	68	66	61	96	38	67	396	66.00

```
[151]: #sort the data
student.sort_values(by="DEPT", ascending=False)
```

	ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	TOTAL	PERCENT
4	105	SOMU	MBA	72	61	81	47	80	97	438	73.00
7	108	ROOPA	MBA	62	58	70	64	63	38	355	59.17
1	102	SURESH	MBA	87	96	43	42	69	68	405	67.50
13	114	SUJITH	MBA	75	84	39	71	69	39	377	62.83
10	111	KUMAR	MBA	46	65	74	94	92	43	414	69.00
8	109	VIGNESH	BBA	63	72	93	85	67	98	478	79.67
2	103	GANESH	BBA	46	86	68	71	72	79	422	70.33
5	106	SHIVU	BBA	46	41	41	94	53	69	344	57.33
14	115	KRISHNA	BBA	51	84	74	61	63	73	406	67.67
11	112	SAKSHI	BBA	96	58	78	66	51	72	421	70.17
12	113	ANIL	ANALYTICS	68	66	61	96	38	67	396	66.00
15	116	VASANTH	ANALYTICS	96	58	78	66	51	72	421	70.17
0	101	RAMESH	ANALYTICS	70	58	67	92	36	60	383	63.83
9	110	VYLESH	ANALYTICS	64	47	95	71	50	58	385	64.17
6	107	VINOD	ANALYTICS	43	45	74	62	62	93	379	63.17
3	104	CHANDRU	ANALYTICS	58	74	91	38	73	88	422	70.33
16	117	QUEEN	ANALYTICS	68	66	61	96	38	67	396	66.00

```
*[155]: #
student.sort_values(by="TOTAL", ascending=False).head(3)
```

```
[155]: ROLL_NO NAME DEPT MATHS SCIENCE SOCIAL KANNADA ENGLISH HINDI TOTAL PERCENT
```

8	109	VIGNESH	BBA	63	72	93	85	67	98	478	79.67
4	105	SOMU	MBA	72	61	81	47	80	97	438	73.00
2	103	GANESH	BBA	46	86	68	71	72	79	422	70.33

```
[163]: student.sort_values(by="TOTAL", ascending=True).head(3)
```

ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	TOTAL	PERCENT	
5	106	SHIVU	BBA	46	41	41	94	53	69	344	57.33
7	108	ROOPA	MBA	62	58	70	64	63	38	355	59.17
13	114	SUJITH	MBA	75	84	39	71	69	39	377	62.83

```
[171]: student.sort_values(by=["DEPT", "TOTAL"], ascending=[True, False])
```

ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	TOTAL	PERCENT	
3	104	CHANDRU	ANALYTICS	58	74	91	38	73	88	422	70.33
15	116	VASANTH	ANALYTICS	96	58	78	66	51	72	421	70.17
12	113	ANIL	ANALYTICS	68	66	61	96	38	67	396	66.00
16	117	QUEEN	ANALYTICS	68	66	61	96	38	67	396	66.00
9	110	VYLESH	ANALYTICS	64	47	95	71	50	58	385	64.17
0	101	RAMESH	ANALYTICS	70	58	67	92	36	60	383	63.83
6	107	VINOD	ANALYTICS	43	45	74	62	62	93	379	63.17
8	109	VIGNESH	BBA	63	72	93	85	67	98	478	79.67
2	103	GANESH	BBA	46	86	68	71	72	79	422	70.33
11	112	SAKSHI	BBA	96	58	78	66	51	72	421	70.17
14	115	KRISHNA	BBA	51	84	74	61	63	73	406	67.67
5	106	SHIVU	BBA	46	41	41	94	53	69	344	57.33
4	105	SOMU	MBA	72	61	81	47	80	97	438	73.00
10	111	KUMAR	MBA	46	65	74	94	92	43	414	69.00
1	102	SURESH	MBA	87	96	43	42	69	68	405	67.50
13	114	SUJITH	MBA	75	84	39	71	69	39	377	62.83
7	108	ROOPA	MBA	62	58	70	64	63	38	355	59.17

```
[177]: #Filter the data - where
student[student["DEPT"]=="MBA"]
```

ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	TOTAL	PERCENT	
1	102	SURESH	MBA	87	96	43	42	69	68	405	67.50
4	105	SOMU	MBA	72	61	81	47	80	97	438	73.00
7	108	ROOPA	MBA	62	58	70	64	63	38	355	59.17
10	111	KUMAR	MBA	46	65	74	94	92	43	414	69.00
13	114	SUJITH	MBA	75	84	39	71	69	39	377	62.83

```
[185]: student[(student["DEPT"]=="MBA") & (student["TOTAL"]<=400)]
```

ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	TOTAL	PERCENT	
7	108	ROOPA	MBA	62	58	70	64	63	38	355	59.17
13	114	SUJITH	MBA	75	84	39	71	69	39	377	62.83

```
*[187]: student[(student["MATHS"]>=80) & (student["SCIENCE"]>=80)] #AND
```

ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	TOTAL	PERCENT	
1	102	SURESH	MBA	87	96	43	42	69	68	405	67.5

```
*[189]: student[(student["MATHS"]>=80) | (student["SCIENCE"]>=80)] #OR
```

ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	TOTAL	PERCENT	
1	102	SURESH	MBA	87	96	43	42	69	68	405	67.50
2	103	GANESH	BBA	46	86	68	71	72	79	422	70.33
11	112	SAKSHI	BBA	96	58	78	66	51	72	421	70.17
13	114	SUJITH	MBA	75	84	39	71	69	39	377	62.83
14	115	KRISHNA	BBA	51	84	74	61	63	73	406	67.67
15	116	VASANTH	ANALYTICS	96	58	78	66	51	72	421	70.17

```
*[193]: student[student["NAME"].str.startswith("S")] # where NAME Like "S%"
```

ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	TOTAL	PERCENT	
1	102	SURESH	MBA	87	96	43	42	69	68	405	67.50
4	105	SOMU	MBA	72	61	81	47	80	97	438	73.00
5	106	SHIVU	BBA	46	41	41	94	53	69	344	57.33
11	112	SAKSHI	BBA	96	58	78	66	51	72	421	70.17
13	114	SUJITH	MBA	75	84	39	71	69	39	377	62.83

```
*[197]: student[student["NAME"].str.endswith("H")] # where NAME Like "%H"
```

```
[197]: ROLL_NO NAME DEPT MATHS SCIENCE SOCIAL KANNADA ENGLISH HINDI TOTAL PERCENT
```

	ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	TOTAL	PERCENT
0	101	RAMESH	ANALYTICS	70	58	67	92	36	60	383	63.83
1	102	SURESH	MBA	87	96	43	42	69	68	405	67.50
2	103	GANESH	BBA	46	86	68	71	72	79	422	70.33
8	109	VIGNESH	BBA	63	72	93	85	67	98	478	79.67
9	110	VYLESH	ANALYTICS	64	47	95	71	50	58	385	64.17
13	114	SUJITH	MBA	75	84	39	71	69	39	377	62.83
15	116	VASANTH	ANALYTICS	96	58	78	66	51	72	421	70.17

```
*[199]: student[student["NAME"].str.contains("A")] # where NAME Like "%A%"
```

	ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	TOTAL	PERCENT
0	101	RAMESH	ANALYTICS	70	58	67	92	36	60	383	63.83
2	103	GANESH	BBA	46	86	68	71	72	79	422	70.33
3	104	CHANDRU	ANALYTICS	58	74	91	38	73	88	422	70.33
7	108	ROOPA	MBA	62	58	70	64	63	38	355	59.17
10	111	KUMAR	MBA	46	65	74	94	92	43	414	69.00
11	112	SAKSHI	BBA	96	58	78	66	51	72	421	70.17
12	113	ANIL	ANALYTICS	68	66	61	96	38	67	396	66.00
14	115	KRISHNA	BBA	51	84	74	61	63	73	406	67.67
15	116	VASANTH	ANALYTICS	96	58	78	66	51	72	421	70.17

```
*[203]: student[(student["MATHS"]>=60) & (student["MATHS"]<=80)].sort_values(by="MATHS", ascending=True) # BETWEEN MATHS >=60 AND MATHS<=80
```

	ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	TOTAL	PERCENT
7	108	ROOPA	MBA	62	58	70	64	63	38	355	59.17
8	109	VIGNESH	BBA	63	72	93	85	67	98	478	79.67
9	110	VYLESH	ANALYTICS	64	47	95	71	50	58	385	64.17
12	113	ANIL	ANALYTICS	68	66	61	96	38	67	396	66.00
16	117	QUEEN	ANALYTICS	68	66	61	96	38	67	396	66.00
0	101	RAMESH	ANALYTICS	70	58	67	92	36	60	383	63.83
4	105	SOMU	MBA	72	61	81	47	80	97	438	73.00
13	114	SUJITH	MBA	75	84	39	71	69	39	377	62.83

```
[ ]: #Aggregate functions(count, sum, mean, median, std, variance)
```

```
[205]: student["MATHS"].sum()
```

```
[205]: 1111
```

```
[207]: student["SCIENCE"].sum()
```

```
[207]: 1119
```

```
[209]: student["TOTAL"].sum()
```

```
[209]: 6842
```

```
[213]: student["TOTAL"].mean()
```

```
[213]: 402.47058823529414
```

```
[215]: student["TOTAL"].max()
```

```
[215]: 478
```

```
[219]: student.sort_values(by="DEPT", ascending=True)
```

	ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	TOTAL	PERCENT
0	101	RAMESH	ANALYTICS	70	58	67	92	36	60	383	63.83
12	113	ANIL	ANALYTICS	68	66	61	96	38	67	396	66.00
9	110	VYLESH	ANALYTICS	64	47	95	71	50	58	385	64.17
15	116	VASANTH	ANALYTICS	96	58	78	66	51	72	421	70.17
6	107	VINOD	ANALYTICS	43	45	74	62	62	93	379	63.17
16	117	QUEEN	ANALYTICS	68	66	61	96	38	67	396	66.00
3	104	CHANDRU	ANALYTICS	58	74	91	38	73	88	422	70.33
2	103	GANESH	BBA	46	86	68	71	72	79	422	70.33
11	112	SAKSHI	BBA	96	58	78	66	51	72	421	70.17
14	115	KRISHNA	BBA	51	84	74	61	63	73	406	67.67
5	106	SHIVU	BBA	46	41	41	94	53	69	344	57.33
8	109	VIGNESH	BBA	63	72	93	85	67	98	478	79.67
4	105	SOMU	MBA	72	61	81	47	80	97	438	73.00
7	108	ROOPA	MBA	62	58	70	64	63	38	355	59.17
10	111	KUMAR	MBA	46	65	74	94	92	43	414	69.00
1	102	SURESH	MBA	87	96	43	42	69	68	405	67.50
13	114	SUJITH	MBA	75	84	39	71	69	39	377	62.83

```
[221]: #groupby is used for summarizing data
```

```
[233]: student.groupby("DEPT")[["MATHS", "SCIENCE", "TOTAL"]].sum()
```

[233]:	MATHS	SCIENCE	TOTAL
DEPT			
ANALYTICS	467	414	2782
BBA	302	341	2071
MBA	342	364	1989

[249]:	#select DEPT,sum(Maths),count(Maths),avg(math) from student group by DEPT
	st_report = student.groupby("DEPT")["MATHS"].agg({"count","sum","max","min","mean","std"})
[263]:	#saving data in excel
	st_report.to_excel(r"C:\Users\CHANDRASHEKAR\OneDrive\Desktop\imax_training_docs\st_report.xlsx")
[265]:	#saving data in csv
	st_report.to_csv(r"C:\Users\CHANDRASHEKAR\OneDrive\Desktop\imax_training_docs\st_report_xxx.csv")
[267]:	student

[267]:	ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	TOTAL	PERCENT
0	101	RAMESH	ANALYTICS	70	58	67	92	36	60	383	63.83
1	102	SURESH	MBA	87	96	43	42	69	68	405	67.50
2	103	GANESH	BBA	46	86	68	71	72	79	422	70.33
3	104	CHANDRU	ANALYTICS	58	74	91	38	73	88	422	70.33
4	105	SOMU	MBA	72	61	81	47	80	97	438	73.00
5	106	SHIVU	BBA	46	41	41	94	53	69	344	57.33
6	107	VINOD	ANALYTICS	43	45	74	62	62	93	379	63.17
7	108	ROOPA	MBA	62	58	70	64	63	38	355	59.17
8	109	VIGNESH	BBA	63	72	93	85	67	98	478	79.67
9	110	VYLESH	ANALYTICS	64	47	95	71	50	58	385	64.17
10	111	KUMAR	MBA	46	65	74	94	92	43	414	69.00
11	112	SAKSHI	BBA	96	58	78	66	51	72	421	70.17
12	113	ANIL	ANALYTICS	68	66	61	96	38	67	396	66.00
13	114	SUJITH	MBA	75	84	39	71	69	39	377	62.83
14	115	KRISHNA	BBA	51	84	74	61	63	73	406	67.67
15	116	VASANTH	ANALYTICS	96	58	78	66	51	72	421	70.17
16	117	QUEEN	ANALYTICS	68	66	61	96	38	67	396	66.00

[327]:	city_data = pd.read_excel(r'C:\Users\CHANDRASHEKAR\Downloads\student_pbi.xlsx',sheet_name="city")
[329]:	city_data

[329]:	ROLL_NO	NAME	DEPT	CITY
0	101	RAMESH	ANALYTICS	BELGAVI
1	102	SURESH	MBA	TUMKUR
2	103	GANESH	BBA	HUBLI
3	104	CHANDRU	ANALYTICS	DAVANGERE
4	105	SOMU	MBA	NAGAMANGALA
5	106	SHIVU	BBA	BELGAVI
6	107	VINOD	ANALYTICS	MANDYA
7	108	ROOPA	MBA	SHIMOGA
8	109	VIGNESH	BBA	HUBLI
9	110	VYLESH	ANALYTICS	DAVANGERE
10	111	KUMAR	MBA	BIJAPUR
11	112	SAKSHI	BBA	HASSAN
12	113	ANIL	ANALYTICS	mysore
13	114	SUJITH	MBA	BHADRAVATHI
14	115	KRISHNA	BBA	mysore
15	116	VASANTH	ANALYTICS	BANGALORE
16	117	QUEEN	ANALYTICS	BANGALORE
17	118	KRISHNA	BBA	mysore
18	119	VASANTH	ANALYTICS	BANGALORE
19	120	QUEEN	ANALYTICS	BANGALORE

[277]:	catg_data = pd.read_excel(r'C:\Users\CHANDRASHEKAR\Downloads\student_pbi.xlsx',sheet_name="Catg")
[279]:	catg_data

[279]:	ROLL_NO	NAME	DEPT	FEES	CATG
0	101	RAMESH	ANALYTICS	33854	OBC
1	102	SURESH	MBA	49065	GM
2	103	GANESH	BBA	40410	OBC
3	104	CHANDRU	ANALYTICS	50000	ST
4	105	SOMU	MBA	48823	SC
5	106	SHIVU	BBA	48974	GM

#	ROLL_NO	NAME	DEPT	FEES	CATG
7	108	ROOPA	MBA	29914	SC
8	109	VIGNESH	BBA	29005	ST
9	110	VYLESH	ANALYTICS	28253	GM
10	111	KUMAR	MBA	32007	SC
11	112	SAKSHI	BBA	44790	ST
12	113	ANIL	ANALYTICS	42913	OBC
13	114	SUJITH	MBA	32724	OBC
14	115	KRISHNA	BBA	48896	GM
15	116	VASANTH	ANALYTICS	42000	OBC
16	117	QUEEN	ANALYTICS	41000	OBC

```
[331]: city = pd.DataFrame(city_data)
catg = pd.DataFrame(catg_data)
```

```
[333]: catg
```

#	ROLL_NO	NAME	DEPT	FEES	CATG
0	101	RAMESH	ANALYTICS	33854	OBC
1	102	SURESH	MBA	49065	GM
2	103	GANESH	BBA	40410	OBC
3	104	CHANDRU	ANALYTICS	50000	ST
4	105	SOMU	MBA	48823	SC
5	106	SHIVU	BBA	48974	GM
6	107	VINOD	ANALYTICS	46870	SC
7	108	ROOPA	MBA	29914	SC
8	109	VIGNESH	BBA	29005	ST
9	110	VYLESH	ANALYTICS	28253	GM
10	111	KUMAR	MBA	32007	SC
11	112	SAKSHI	BBA	44790	ST
12	113	ANIL	ANALYTICS	42913	OBC
13	114	SUJITH	MBA	32724	OBC
14	115	KRISHNA	BBA	48896	GM
15	116	VASANTH	ANALYTICS	42000	OBC
16	117	QUEEN	ANALYTICS	41000	OBC

```
[305]: #INNER JOIN
student.merge(city, left_on="ROLL_NO", right_on="ROLL_NO", how="inner")
```

#	ROLL_NO	NAME_x	DEPT_x	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	TOTAL	PERCENT	NAME_y	DEPT_y	CITY
0	101	RAMESH	ANALYTICS	70	58	67	92	36	60	383	63.83	RAMESH	ANALYTICS	BELGAVI
1	102	SURESH	MBA	87	96	43	42	69	68	405	67.50	SURESH	MBA	TUMKUR
2	103	GANESH	BBA	46	86	68	71	72	79	422	70.33	GANESH	BBA	HUBLI
3	104	CHANDRU	ANALYTICS	58	74	91	38	73	88	422	70.33	CHANDRU	ANALYTICS	DAVANGERE
4	105	SOMU	MBA	72	61	81	47	80	97	438	73.00	SOMU	MBA	NAGAMANGALA
5	106	SHIVU	BBA	46	41	41	94	53	69	344	57.33	SHIVU	BBA	BELGAVI
6	107	VINOD	ANALYTICS	43	45	74	62	62	93	379	63.17	VINOD	ANALYTICS	MANDYA
7	108	ROOPA	MBA	62	58	70	64	63	38	355	59.17	ROOPA	MBA	SHIMOGA
8	109	VIGNESH	BBA	63	72	93	85	67	98	478	79.67	VIGNESH	BBA	HUBLI
9	110	VYLESH	ANALYTICS	64	47	95	71	50	58	385	64.17	VYLESH	ANALYTICS	DAVANGERE
10	111	KUMAR	MBA	46	65	74	94	92	43	414	69.00	KUMAR	MBA	BIJAPUR
11	112	SAKSHI	BBA	96	58	78	66	51	72	421	70.17	SAKSHI	BBA	HASSAN
12	113	ANIL	ANALYTICS	68	66	61	96	38	67	396	66.00	ANIL	ANALYTICS	MYSORE
13	114	SUJITH	MBA	75	84	39	71	69	39	377	62.83	SUJITH	MBA	BHADRAVATHI
14	115	KRISHNA	BBA	51	84	74	61	63	73	406	67.67	KRISHNA	BBA	MYSORE
15	116	VASANTH	ANALYTICS	96	58	78	66	51	72	421	70.17	VASANTH	ANALYTICS	BANGALORE
16	117	QUEEN	ANALYTICS	68	66	61	96	38	67	396	66.00	QUEEN	ANALYTICS	BANGALORE

```
[313]: #LEFT JOIN
student.merge(city, left_on="ROLL_NO", right_on="ROLL_NO", how="left")
```

#	ROLL_NO	NAME_x	DEPT_x	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	NAME_y	DEPT_y	CITY
0	101	RAMESH	ANALYTICS	70	58	67	92	36	60	RAMESH	ANALYTICS	BELGAVI
1	102	SURESH	MBA	87	96	43	42	69	68	SURESH	MBA	TUMKUR
2	103	GANESH	BBA	46	86	68	71	72	79	GANESH	BBA	HUBLI
3	104	CHANDRU	ANALYTICS	58	74	91	38	73	88	CHANDRU	ANALYTICS	DAVANGERE
4	105	SOMU	MBA	72	61	81	47	80	97	SOMU	MBA	NAGAMANGALA
5	106	SHIVU	BBA	46	41	41	94	53	69	SHIVU	BBA	BELGAVI
6	107	VINOD	ANALYTICS	43	45	74	62	62	93	VINOD	ANALYTICS	MANDYA
7	108	ROOPA	MBA	62	58	70	64	63	38	ROOPA	MBA	SHIMOGA
8	109	VIGNESH	BBA	63	72	93	85	67	98	VIGNESH	BBA	HUBLI
9	110	VYLESH	ANALYTICS	64	47	95	71	50	58	VYLESH	ANALYTICS	DAVANGERE

10	111	KUMAR	MBA	46	65	74	94	92	43	KUMAR	MBA	BIJAPUR
11	112	SAKSHI	BBA	96	58	78	66	51	72	SAKSHI	BBA	HASSAN
12	113	ANIL	ANALYTICS	68	66	61	96	38	67	ANIL	ANALYTICS	mysore
13	114	SUJITH	MBA	75	84	39	71	69	39	SUJITH	MBA	BHADRAVATHI
14	115	KRISHNA	BBA	51	84	74	61	63	73	KRISHNA	BBA	mysore
15	116	VASANTH	ANALYTICS	96	58	78	66	51	72	VASANTH	ANALYTICS	BANGALORE
16	117	QUEEN	ANALYTICS	68	66	61	96	38	67	QUEEN	ANALYTICS	BANGALORE
17	118	VASANTH	ANALYTICS	96	58	78	66	51	72	Nan	Nan	Nan
18	119	QUEEN	ANALYTICS	68	66	61	96	38	67	Nan	Nan	Nan
19	120	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	QUEEN	ANALYTICS	BANGALORE

```
*[337]: #RIGHT JOIN
student.merge(city, left_on="ROLL_NO", right_on="ROLL_NO", how="right")
```

	ROLL_NO	NAME_x	DEPT_x	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	NAME_y	DEPT_y	CITY
0	101	RAMESH	ANALYTICS	70.0	58.0	67.0	92.0	36.0	60.0	RAMESH	ANALYTICS	BELGAVI
1	102	SURESH	MBA	87.0	96.0	43.0	42.0	69.0	68.0	SURESH	MBA	TUMKUR
2	103	GANESH	BBA	46.0	86.0	68.0	71.0	72.0	79.0	GANESH	BBA	HUBLI
3	104	CHANDRU	ANALYTICS	58.0	74.0	91.0	38.0	73.0	88.0	CHANDRU	ANALYTICS	DAVANGERE
4	105	SOMU	MBA	72.0	61.0	81.0	47.0	80.0	97.0	SOMU	MBA	NAGAMANGALA
5	106	SHIVU	BBA	46.0	41.0	41.0	94.0	53.0	69.0	SHIVU	BBA	BELGAVI
6	107	VINOD	ANALYTICS	43.0	45.0	74.0	62.0	62.0	93.0	VINOD	ANALYTICS	MANDYA
7	108	ROOPA	MBA	62.0	58.0	70.0	64.0	63.0	38.0	ROOPA	MBA	SHIMOGA
8	109	VIGNESH	BBA	63.0	72.0	93.0	85.0	67.0	98.0	VIGNESH	BBA	HUBLI
9	110	VYLESH	ANALYTICS	64.0	47.0	95.0	71.0	50.0	58.0	VYLESH	ANALYTICS	DAVANGERE
10	111	KUMAR	MBA	46.0	65.0	74.0	94.0	92.0	43.0	KUMAR	MBA	BIJAPUR
11	112	SAKSHI	BBA	96.0	58.0	78.0	66.0	51.0	72.0	SAKSHI	BBA	HASSAN
12	113	ANIL	ANALYTICS	68.0	66.0	61.0	96.0	38.0	67.0	ANIL	ANALYTICS	mysore
13	114	SUJITH	MBA	75.0	84.0	39.0	71.0	69.0	39.0	SUJITH	MBA	BHADRAVATHI
14	115	KRISHNA	BBA	51.0	84.0	74.0	61.0	63.0	73.0	KRISHNA	BBA	mysore
15	116	VASANTH	ANALYTICS	96.0	58.0	78.0	66.0	51.0	72.0	VASANTH	ANALYTICS	BANGALORE
16	117	QUEEN	ANALYTICS	68.0	66.0	61.0	96.0	38.0	67.0	QUEEN	ANALYTICS	BANGALORE
17	118	VASANTH	ANALYTICS	96.0	58.0	78.0	66.0	51.0	72.0	KRISHNA	BBA	mysore
18	119	QUEEN	ANALYTICS	68.0	66.0	61.0	96.0	38.0	67.0	VASANTH	ANALYTICS	BANGALORE
19	120	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	QUEEN	ANALYTICS	BANGALORE

[335]: city

	ROLL_NO	NAME	DEPT	CITY
0	101	RAMESH	ANALYTICS	BELGAVI
1	102	SURESH	MBA	TUMKUR
2	103	GANESH	BBA	HUBLI
3	104	CHANDRU	ANALYTICS	DAVANGERE
4	105	SOMU	MBA	NAGAMANGALA
5	106	SHIVU	BBA	BELGAVI
6	107	VINOD	ANALYTICS	MANDYA
7	108	ROOPA	MBA	SHIMOGA
8	109	VIGNESH	BBA	HUBLI
9	110	VYLESH	ANALYTICS	DAVANGERE
10	111	KUMAR	MBA	BIJAPUR
11	112	SAKSHI	BBA	HASSAN
12	113	ANIL	ANALYTICS	mysore
13	114	SUJITH	MBA	BHADRAVATHI
14	115	KRISHNA	BBA	mysore
15	116	VASANTH	ANALYTICS	BANGALORE
16	117	QUEEN	ANALYTICS	BANGALORE
17	118	KRISHNA	BBA	mysore
18	119	VASANTH	ANALYTICS	BANGALORE
19	120	QUEEN	ANALYTICS	BANGALORE

```
[357]: #LEFT JOIN
st_merge = student.merge(catg, left_on="ROLL_NO", right_on="ROLL_NO", how="left")
```

[365]: type(st_merge)

[365]: pandas.core.frame.DataFrame

[381]: st_merge

	ROLL_NO	NAME_x	DEPT_x	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI	NAME_y	DEPT_y	FEES	CATG
0	101	RAMESH	ANALYTICS	70	58	67	92	36	60	RAMESH	ANALYTICS	33854.0	OB
1	102	SURESH	MBA	87	96	43	42	69	68	SURESH	MBA	49065.0	GM
2	103	GANESH	BBA	46	86	68	71	72	79	GANESH	BBA	40410.0	OB
3	104	CHANDRU	ANALYTICS	58	74	91	38	73	88	CHANDRU	ANALYTICS	60000.0	OT

3	104	CHANDRU	ANALYTICS	70	74	91	50	73	86	CHANDRU	ANALYTICS	50000.0	ST
4	105	SOMU	MBA	72	61	81	47	80	97	SOMU	MBA	48823.0	SC
5	106	SHIVU	BBA	46	41	41	94	53	69	SHIVU	BBA	48974.0	GM
6	107	VINOD	ANALYTICS	43	45	74	62	62	93	VINOD	ANALYTICS	46870.0	SC
7	108	ROOPA	MBA	62	58	70	64	63	38	ROOPA	MBA	29914.0	SC
8	109	VIGNESH	BBA	63	72	93	85	67	98	VIGNESH	BBA	29005.0	ST
9	110	VYLESH	ANALYTICS	64	47	95	71	50	58	VYLESH	ANALYTICS	28253.0	GM
10	111	KUMAR	MBA	46	65	74	94	92	43	KUMAR	MBA	32007.0	SC
11	112	SAKSHI	BBA	96	58	78	66	51	72	SAKSHI	BBA	44790.0	ST
12	113	ANIL	ANALYTICS	68	66	61	96	38	67	ANIL	ANALYTICS	42913.0	OBC
13	114	SUJITH	MBA	75	84	39	71	69	39	SUJITH	MBA	32724.0	OBC
14	115	KRISHNA	BBA	51	84	74	61	63	73	KRISHNA	BBA	48896.0	GM
15	116	VASANTH	ANALYTICS	96	58	78	66	51	72	VASANTH	ANALYTICS	42000.0	OBC
16	117	QUEEN	ANALYTICS	68	66	61	96	38	67	QUEEN	ANALYTICS	41000.0	OBC
17	118	VASANTH	ANALYTICS	96	58	78	66	51	72	NaN	NaN	NaN	NaN
18	119	QUEEN	ANALYTICS	68	66	61	96	38	67	NaN	NaN	NaN	NaN

```
[389]: st_merge[["ROLL_NO","NAME_y","DEPT_y","MATHS","CATG","FEES"]]
```

	ROLL_NO	NAME_y	DEPT_y	MATHS	CATG	FEES
0	101	RAMESH	ANALYTICS	70	OBC	33854.0
1	102	SURESH	MBA	87	GM	49065.0
2	103	GANESH	BBA	46	OBC	40410.0
3	104	CHANDRU	ANALYTICS	58	ST	50000.0
4	105	SOMU	MBA	72	SC	48823.0
5	106	SHIVU	BBA	46	GM	48974.0
6	107	VINOD	ANALYTICS	43	SC	46870.0
7	108	ROOPA	MBA	62	SC	29914.0
8	109	VIGNESH	BBA	63	ST	29005.0
9	110	VYLESH	ANALYTICS	64	GM	28253.0
10	111	KUMAR	MBA	46	SC	32007.0
11	112	SAKSHI	BBA	96	ST	44790.0
12	113	ANIL	ANALYTICS	68	OBC	42913.0
13	114	SUJITH	MBA	75	OBC	32724.0
14	115	KRISHNA	BBA	51	GM	48896.0
15	116	VASANTH	ANALYTICS	96	OBC	42000.0
16	117	QUEEN	ANALYTICS	68	OBC	41000.0
17	118	NaN	NaN	96	NaN	NaN
18	119	NaN	NaN	68	NaN	NaN

```
[431]: missing = pd.read_excel(r'C:\Users\CHANDRASHEKAR\Downloads\student_pbi.xlsx',sheet_name="st_missing")
```

```
[433]: st_miss = pd.DataFrame(missing)
```

```
[435]: st_miss.isna().sum()
```

```
[435]: ROLL_NO      0
NAME        1
DEPT        1
MATHS       2
SCIENCE     2
SOCIAL      1
KANNADA     1
ENGLISH     1
HINDI       1
dtype: int64
```

```
[439]: st_miss.fillna({"NAME":"NOT AVAILABLE"})
```

	ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI
0	101	RAMESH	ANALYTICS	70.0	58.0	67.0	92.0	36.0	60.0
1	102	SURESH	MBA	87.0	96.0	43.0	42.0	69.0	68.0
2	103	GANESH	Nan	NaN	86.0	68.0	71.0	72.0	79.0
3	104	CHANDRU	ANALYTICS	58.0	74.0	91.0	38.0	73.0	88.0
4	105	SOMU	MBA	72.0	61.0	81.0	47.0	NaN	97.0
5	106	SHIVU	BBA	46.0	41.0	41.0	94.0	53.0	69.0
6	107	VINOD	ANALYTICS	43.0	45.0	74.0	62.0	62.0	93.0
7	108	ROOPA	MBA	62.0	NaN	70.0	64.0	63.0	38.0
8	109	VIGNESH	BBA	63.0	72.0	93.0	85.0	67.0	98.0
9	110	VYLESH	ANALYTICS	64.0	47.0	95.0	71.0	50.0	58.0
10	111	NOT AVAILABLE	MBA	46.0	65.0	74.0	94.0	92.0	43.0
11	112	SAKSHI	BBA	NaN	58.0	78.0	NaN	51.0	NaN
12	113	ANIL	ANALYTICS	68.0	66.0	61.0	96.0	38.0	67.0
13	114	SUJITH	MBA	75.0	NaN	39.0	71.0	69.0	39.0
14	115	KRISHNA	BBA	51.0	84.0	74.0	61.0	63.0	73.0
15	116	VASANTH	ANALYTICS	96.0	58.0	NaN	66.0	51.0	72.0

```
[16]    117      QUEEN ANALYTICS   68.0    66.0    61.0    96.0    38.0    67.0
```

```
[429]: st_miss
```

```
[429]:
```

	ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI
0	101	RAMESH	ANALYTICS	70.0	58.0	67.0	92.0	36.0	60.0
1	102	SURESH	MBA	87.0	96.0	43.0	42.0	69.0	68.0
2	103	GANESH	Nan	90.0	86.0	68.0	71.0	72.0	79.0
3	104	CHANDRU	ANALYTICS	58.0	74.0	91.0	38.0	73.0	88.0
4	105	SOMU	MBA	72.0	61.0	81.0	47.0	NaN	97.0
5	106	SHIVU	BBA	46.0	41.0	41.0	94.0	53.0	69.0
6	107	VINOD	ANALYTICS	43.0	45.0	74.0	62.0	62.0	93.0
7	108	ROOPA	MBA	62.0	NaN	70.0	64.0	63.0	38.0
8	109	VIGNESH	BBA	63.0	72.0	93.0	85.0	67.0	98.0
9	110	VYLESH	ANALYTICS	64.0	47.0	95.0	71.0	50.0	58.0
10	111	NaN	MBA	46.0	65.0	74.0	94.0	92.0	43.0
11	112	SAKSHI	BBA	90.0	58.0	78.0	NaN	51.0	NaN
12	113	ANIL	ANALYTICS	68.0	66.0	61.0	96.0	38.0	67.0
13	114	SUJITH	MBA	75.0	NaN	39.0	71.0	69.0	39.0
14	115	KRISHNA	BBA	51.0	84.0	74.0	61.0	63.0	73.0
15	116	VASANTH	ANALYTICS	96.0	58.0	NaN	66.0	51.0	72.0
16	117	QUEEN	ANALYTICS	68.0	66.0	61.0	96.0	38.0	67.0

```
[441]: dup_data = pd.read_excel(r'C:\Users\CHANDRASHEKAR\Downloads\student_pbi.xlsx',sheet_name="st_duplicates")
```

```
[445]: st_dup = pd.DataFrame(dup_data)
```

```
[449]: st_dup.duplicated()
```

```
[449]:
```

0	False
1	False
2	False
3	False
4	False
5	False
6	False
7	False
8	False
9	False
10	False
11	False
12	False
13	False
14	False
15	False
16	False
17	True
18	True
19	True
20	True

dtype: bool

```
[451]: st_dup.drop_duplicates()
```

```
[451]:
```

	ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI
0	101	RAMESH	ANALYTICS	70	58	67	92	36	60
1	102	SURESH	MBA	87	96	43	42	69	68
2	103	GANESH	BBA	46	86	68	71	72	79
3	104	CHANDRU	ANALYTICS	58	74	91	38	73	88
4	105	SOMU	MBA	72	61	81	47	80	97
5	106	SHIVU	BBA	46	41	41	94	53	69
6	107	VINOD	ANALYTICS	43	45	74	62	62	93
7	108	ROOPA	MBA	62	58	70	64	63	38
8	109	VIGNESH	BBA	63	72	93	85	67	98
9	110	VYLESH	ANALYTICS	64	47	95	71	50	58
10	111	KUMAR	MBA	46	65	74	94	92	43
11	112	SAKSHI	BBA	96	58	78	66	51	72
12	113	ANIL	ANALYTICS	68	66	61	96	38	67
13	114	SUJITH	MBA	75	84	39	71	69	39
14	115	KRISHNA	BBA	51	84	74	61	63	73
15	116	VASANTH	ANALYTICS	96	58	78	66	51	72
16	117	QUEEN	ANALYTICS	68	66	61	96	38	67

```
[463]: st_dup.drop(["DEPT"],axis=1)
```

```
[463]:
```

	ROLL_NO	NAME	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI
0	101	RAMESH	70	58	67	92	36	60
1	102	SURESH	87	96	43	42	69	68
2	103	GANESH	46	86	68	71	72	79
3	104	CHANDRU	58	74	91	38	73	88
4	105	SOMU	72	61	81	47	80	97
5	106	SHIVU	46	41	41	94	53	69

6	107	VINOD	43	45	74	62	62	93
7	108	ROOPA	62	58	70	64	63	38
8	109	VIGNESH	63	72	93	85	67	98
9	110	VYLESH	64	47	95	71	50	58
10	111	KUMAR	46	65	74	94	92	43
11	112	SAKSHI	96	58	78	66	51	72
12	113	ANIL	68	66	61	96	38	67
13	114	SUJITH	75	84	39	71	69	39
14	115	KRISHNA	51	84	74	61	63	73
15	116	VASANTH	96	58	78	66	51	72
16	117	QUEEN	68	66	61	96	38	67
17	107	VINOD	43	45	74	62	62	93
18	108	ROOPA	62	58	70	64	63	38
19	109	VIGNESH	63	72	93	85	67	98
20	110	VYLESH	64	47	95	71	50	58

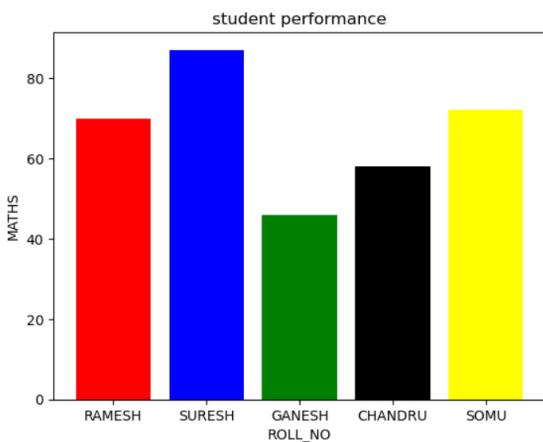
```
[465]: student
```

	ROLL_NO	NAME	DEPT	MATHS	SCIENCE	SOCIAL	KANNADA	ENGLISH	HINDI
0	101	RAMESH	ANALYTICS	70	58	67	92	36	60
1	102	SURESH	MBA	87	96	43	42	69	68
2	103	GANESH	BBA	46	86	68	71	72	79
3	104	CHANDRU	ANALYTICS	58	74	91	38	73	88
4	105	SOMU	MBA	72	61	81	47	80	97
5	106	SHIVU	BBA	46	41	41	94	53	69
6	107	VINOD	ANALYTICS	43	45	74	62	62	93
7	108	ROOPA	MBA	62	58	70	64	63	38
8	109	VIGNESH	BBA	63	72	93	85	67	98
9	110	VYLESH	ANALYTICS	64	47	95	71	50	58
10	111	KUMAR	MBA	46	65	74	94	92	43
11	112	SAKSHI	BBA	96	58	78	66	51	72
12	113	ANIL	ANALYTICS	68	66	61	96	38	67
13	114	SUJITH	MBA	75	84	39	71	69	39
14	115	KRISHNA	BBA	51	84	74	61	63	73
15	116	VASANTH	ANALYTICS	96	58	78	66	51	72
16	117	QUEEN	ANALYTICS	68	66	61	96	38	67
17	118	VASANTH	ANALYTICS	96	58	78	66	51	72
18	119	QUEEN	ANALYTICS	68	66	61	96	38	67

```
[467]: import matplotlib.pyplot as plt
import seaborn as sns
```

```
[485]: x=student[["NAME"]].head()
y=student[["MATHS"]].head()

plt.bar(x,y,color=["red","blue","green","black","yellow"])
plt.xlabel("ROLL_NO")
plt.ylabel("MATHS")
plt.title("student performance")
plt.show()
```

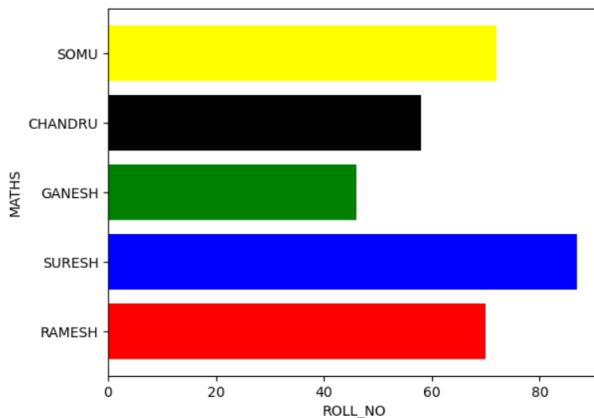


```
[487]: x=student[["NAME"]].head()
y=student[["MATHS"]].head()

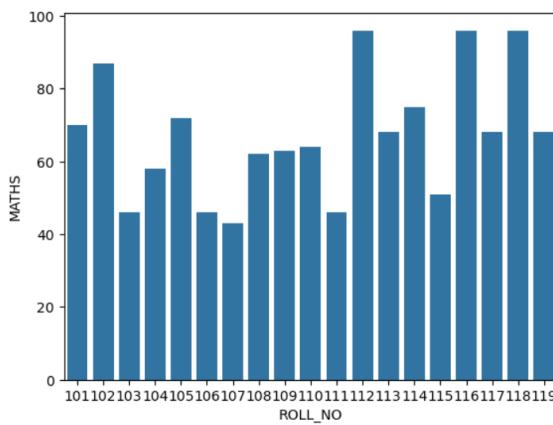
plt.barh(x,y,color=["red","blue","green","black","yellow"])
plt.xlabel("ROLL_NO")
plt.ylabel("MATHS")
plt.title("student performance")
plt.show()
```



student performance



```
[497]: sns.barplot(x="ROLL_NO",y="MATHS",data=student)
plt.show()
```



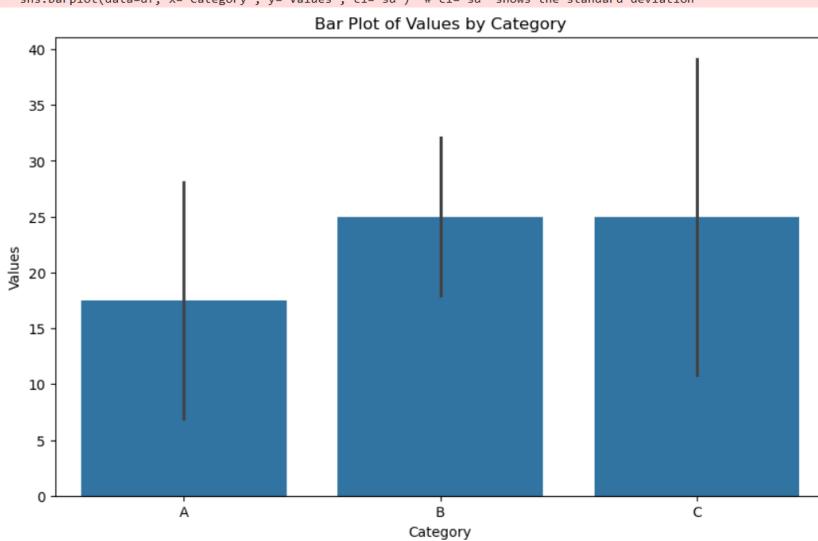
```
[501]: import seaborn as sns
import matplotlib.pyplot as plt
import pandas as pd

# Sample DataFrame creation (you can replace this with your actual DataFrame)
data = {
    'Category': ['A', 'B', 'C', 'A', 'B', 'C'],
    'Values': [10, 20, 15, 25, 30, 35]
}
df = pd.DataFrame(data)

# Create a bar plot
plt.figure(figsize=(10, 6))
sns.barplot(data=df, x='Category', y='Values', ci='sd') # ci='sd' shows the standard deviation
plt.title('Bar Plot of Values by Category')
plt.show()

C:\Users\CHANDRASHEKAR\AppData\Local\Temp\ipykernel_6572\3343906816.py:14: FutureWarning:
The 'ci' parameter is deprecated. Use 'errorbar='sd'' for the same effect.

sns.barplot(data=df, x='Category', y='Values', ci='sd') # ci='sd' shows the standard deviation
```



```
[ ]:
```

