Handling Missing Values

② Finding Missing Values

```
import pandas as pd.
dataFrame = pd.read_csv('data.csv')
print(dataFrame)
```

① Finding Missing Values

```
dataFrame.isnull()
```

	Name	Surname	salary(th)	Locality
0	False	False	False	False
1	"	"	"	True
2	"	"	"	False
3	"	"	"	"
4	"	True	"	"
5	"	"	"	"

b) Example of finding missing values:

```
dataFrame.isnull().any()
```

Output

```
Name False
Surname True
salary(th) False
Locality True
dtype: bool
```

⑤ Removing Missing Values

dataFrame.dropna()

	Name	Surname	Salary(th)	Locality
0	ABC	Th	15.0	Loc1
2	PQR	R	30.5	Loc2
3	RST	S	45.0	Loc2

③ Replacing with a value

dataFrame.fillna('')

Output

	Name	Surname	salary(th)	Locality
0	ABC	Th	15.0	Loc1
1	XYZ	C	27.0	
2	PQR	R	30.5	Loc2
3	RST	S	45.0	Loc2
4	ORT		23.0	Loc3
5	SDF		35.5	Loc2

④ Reshaping Data

dataFrame['Gender'] = ['M', 'F', 'F', 'M', 'M', 'F']

dataFrame

Output

	Name	Surname	salary(th)	Locality	Gender
0	ABC	Th	15.0	Loc1	M
1	XYZ	C	27.0	NaN	F
2	PQR	R	30.5	Loc2	F
3	RST	S	45.0	Loc2	M
4	ORT	NaN	23.0	Loc3	M
5	SDF	NaN	35.5	Loc2	F

dataFrame['Gender'] = dataFrame['Gender'].map({'M':0, 'F':1,}).astype(float)

dataFrame

Output

	Name	Surname	salary(th)	Locality	Gender
0	ABC	Th	15.0	Loc1	0.0
1	XYZ	C	27.0	NaN	1.0
2	PQR	R	30.5	Loc2	1.0
3	RST	S	45.0	Loc2	0.0
4	ORT	NaN	23.0	Loc3	0.0
5	SDF	NaN	35.5	Loc2	1.0

Grouping Data

```
dataFrame.groupby('Locality').Gender.value_counts()
```

Output

```
Locality  Gender
Loc3      0.0    1
Loc1      0.0    1
Loc2      1.0    2
          0.0    1
Name: Gender, dtype: int64
```

Unique Data

```
dataFrame.Locality.unique()
```

output

```
array(['Loc1', nan, 'Loc2', 'Loc3'], dtype=object)
```

Filtering Data

```
dataFrame[dataFrame['salary(th)']>25]
```

Output

	Name	Surname	salary(th)	Locality	Gender
1	XYZ	C	27.0	NaN	1.0
2	PQR	R	30.5	Loc2	1.0
3	RST	S	45.0	Loc2	0.0
5	SDF	NaN	35.5	Loc2	1.0

```
dataFrame[dataFrame['Locality']=='Loc2']
```

Output

All @Loc2 data will be show.

Merging Data.

```
pd.merge(left, right, how='inner', on=None,
         left_on=None, right_on=None, left_index=
         False, right_index=False, sort=True)
```

```
left = pd.DataFrame({
    'id': [1,2,3,4,5],
    'Name': ['Priya', 'Riya', 'Amit', 'Neha'],
    'subject_id': ['sub1', 'sub2', 'sub3', 'sub4']})
right = pd.DataFrame({
    id: [1,2,3,4,5],
    'Name': ['Ram', 'Raj', 'Shivansh', 'Kajal', Komal]
    'subject_id': ['sub5', 'sub2', 'sub3', 'sub6', 'sub8']})
```

print(left)

Output

	Name	id	subject_id
0	Priya	1	sub1
1	Riya	2	sub2
2	Amit	3	sub3
3	Neha	4	sub4

Print(right)

Output

	Name	id	subject_id
0	Ram	1	sub5
1	Raj	2	sub 2
2	Shivansh	3	sub 3
3	Kajal	4	sub6
4	Komal	5	sub8

Grouping Data

```
ipl_data = {'Team': ['Riders', 'Riders', 'Devils', 'Devils',
            'Kings', 'Kings', 'Riders', ~~'Kings'~~ 'Royals',
            'Royals', 'Riders']
    'Rank': [1,2,2,3,3,4,1,1,2,4,1,2],
    'Year': [2014,2015,2014,2015,2014,2015,2016,2
             2016,2014,2015,2017],
    'Points': [3,5,7,8,7,8,6,3,9,10,11,7]}
df = pd.DataFrame(ipl_data)
grouped = df.groupby('Year')
print(grouped.get_group(2014))
```

Output

	Points	Rank	Team	Year
0	3	1	Riders	2014
2	7	2	Devils	4
4	7	3	Kings	4
9	10	4	Royals	4