

```
# Pavithra S
# 21BIT0389
# Assignment 1
import pandas as pd
import numpy as np

# Task 1 (Create a pandas dataframe (DataFrame name as 'df') (10 observation and 5 features))
df = pd.DataFrame([[ 'Ram',20, 'M',75000, 'NPS' ],[ 'Radha',25, 'F',85000, 'LGMS' ],[ 'Yashoda',21, 'F',80000, 'NPS' ],
[ 'Sid',23, 'M',np.nan, 'DPS' ],[ 'Viblav',np.nan, 'M',55000, 'EVPS' ],[ 'Savi',20, 'M',40000, 'NPS' ],
[ 'Sudha',19, 'F',60000, 'St.Judes' ],[ 'Santhosh',20, 'M',65000, 'LGS' ],[ 'Sam',19, 'F',80000,np.nan],[ 'Arav',20, 'M',35000, 'EVPS' ]])
df.columns=['Name', 'Age', 'Gender', 'Salary', 'School']
df
```

|   | Name     | Age  | Gender | Salary  | School   |
|---|----------|------|--------|---------|----------|
| 0 | Ram      | 20.0 | M      | 75000.0 | NPS      |
| 1 | Radha    | 25.0 | F      | 85000.0 | LGMS     |
| 2 | Yashoda  | 21.0 | F      | 80000.0 | NPS      |
| 3 | Sid      | 23.0 | M      | NaN     | DPS      |
| 4 | Viblav   | NaN  | M      | 55000.0 | EVPS     |
| 5 | Savi     | 20.0 | M      | 40000.0 | NPS      |
| 6 | Sudha    | 19.0 | F      | 60000.0 | St.Judes |
| 7 | Santhosh | 20.0 | M      | 65000.0 | LGS      |
| 8 | Sam      | 19.0 | F      | 80000.0 | NaN      |
| 9 | Arav     | 20.0 | M      | 35000.0 | EVPS     |

```
# Task 2 (Check the info of 'df')
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 5 columns):
#   Column  Non-Null Count  Dtype
---  -
0   Name    10 non-null        object
1   Age     9 non-null         float64
2   Gender  10 non-null        object
3   Salary  9 non-null         float64
4   School  9 non-null         object
dtypes: float64(2), object(3)
memory usage: 528.0+ bytes
```

```
# Task 3 (Check the descriptive statistics of 'df')
df.describe()
```

|       | Age       | Salary       |
|-------|-----------|--------------|
| count | 9.000000  | 9.000000     |
| mean  | 20.777778 | 63888.888889 |
| std   | 1.986063  | 17989.194287 |
| min   | 19.000000 | 35000.000000 |
| 25%   | 20.000000 | 55000.000000 |
| 50%   | 20.000000 | 65000.000000 |
| 75%   | 21.000000 | 80000.000000 |
| max   | 25.000000 | 85000.000000 |

```
# Task 4 (check the 4th index observation with 'loc' slicing operator.)
# index = 4 implies that the serial number = 5, since index starts from 0
df.loc[5]
```

|      |      |
|------|------|
| Name | Savi |
| Age  | 20.0 |

```
Gender      M
Salary    40000.0
School     NPS
Name: 5, dtype: object

# or we can use iloc
df.iloc[4]

Name      Viblav
Age       NaN
Gender     M
Salary    55000.0
School     EVPS
Name: 4, dtype: object

# Task 5 (Check the null values in your 'df')
df.isnull().any()

Name      False
Age       True
Gender     False
Salary    True
School    True
dtype: bool

# Remove nan
df.Age = df.Age.fillna(df.Age.median())
df.Salary = df.Salary.fillna(df.Salary.median())
df.School = df.School.fillna(df.School.mode().iloc[0])
df
```

|   | Name     | Age  | Gender | Salary  | School   |
|---|----------|------|--------|---------|----------|
| 0 | Ram      | 20.0 | M      | 75000.0 | NPS      |
| 1 | Radha    | 25.0 | F      | 85000.0 | LGMS     |
| 2 | Yashoda  | 21.0 | F      | 80000.0 | NPS      |
| 3 | Sid      | 23.0 | M      | 65000.0 | DPS      |
| 4 | Viblav   | 20.0 | M      | 55000.0 | EVPS     |
| 5 | Savi     | 20.0 | M      | 40000.0 | NPS      |
| 6 | Sudha    | 19.0 | F      | 60000.0 | St.Judes |
| 7 | Santhosh | 20.0 | M      | 65000.0 | LGS      |
| 8 | Sam      | 19.0 | F      | 80000.0 | NPS      |
| 9 | Arav     | 20.0 | M      | 35000.0 | EVPS     |

