Iris

Introduction:

This exercise may seem a little bit strange, but keep doing it.

Step 1. Import the necessary libraries

```
import pandas as pd
import numpy as np
```

Step 2. Import the dataset from this address.

Step 3. Assign it to a variable called iris

```
iris = pd.read_csv('iris.csv', header=None)
print("Iris dataset (before column names):\n", iris.head())
```

```
Iris dataset (before column names):
      0    1    2    3            4
0  5.1  3.5  1.4  0.2  Iris-setosa
1  4.9  3.0  1.4  0.2  Iris-setosa
2  4.7  3.2  1.3  0.2  Iris-setosa
3  4.6  3.1  1.5  0.2  Iris-setosa
4  5.0  3.6  1.4  0.2  Iris-setosa
```

Step 4. Create columns for the dataset

```
# 1. sepal_length (in cm)
# 2. sepal_width (in cm)
# 3. petal_length (in cm)
# 4. petal_width (in cm)
# 5. class
iris.columns = ['sepal_length', 'sepal_width', 'petal_length', 'petal_width', 'class']
print("Iris dataset (after column names):\n", iris.head())
```

```
Iris dataset (after column names):
    sepal_length  sepal_width  petal_length  petal_width        class
0           5.1          3.5           1.4          0.2  Iris-setosa
1           4.9          3.0           1.4          0.2  Iris-setosa
2           4.7          3.2           1.3          0.2  Iris-setosa
3           4.6          3.1           1.5          0.2  Iris-setosa
4           5.0          3.6           1.4          0.2  Iris-setosa
```

Step 5. Is there any missing value in the dataframe?

```
missing_values = iris.isna().sum()
print("Missing values:\n", missing_values)
```

```
Missing values:
 sepal_length    0
sepal_width     0
petal_length    0
petal_width     0
class           0
dtype: int64
```

Step 6. Lets set the values of the rows 10 to 29 of the column 'petal_length' to NaN

```
iris.loc[10:29, 'petal_length'] = np.nan
print("Iris after setting petal_length (rows 10-29) to NaN:\n", iris[10:30])
```

```
Iris after setting petal_length (rows 10-29) to NaN:
     sepal_length  sepal_width  petal_length  petal_width        class
```

```
    10           5.4          3.7           NaN          0.2  Iris-setosa
    11           4.8          3.4           NaN          0.2  Iris-setosa
    12           4.8          3.0           NaN          0.1  Iris-setosa
    13           4.3          3.0           NaN          0.1  Iris-setosa
    14           5.8          4.0           NaN          0.2  Iris-setosa
    15           5.7          4.4           NaN          0.4  Iris-setosa
    16           5.4          3.9           NaN          0.4  Iris-setosa
    17           5.1          3.5           NaN          0.3  Iris-setosa
    18           5.7          3.8           NaN          0.3  Iris-setosa
    19           5.1          3.8           NaN          0.3  Iris-setosa
    20           5.4          3.4           NaN          0.2  Iris-setosa
    21           5.1          3.7           NaN          0.4  Iris-setosa
    22           4.6          3.6           NaN          0.2  Iris-setosa
    23           5.1          3.3           NaN          0.5  Iris-setosa
    24           4.8          3.4           NaN          0.2  Iris-setosa
    25           5.0          3.0           NaN          0.2  Iris-setosa
    26           5.0          3.4           NaN          0.4  Iris-setosa
    27           5.2          3.5           NaN          0.2  Iris-setosa
    28           5.2          3.4           NaN          0.2  Iris-setosa
    29           4.7          3.2           NaN          0.2  Iris-setosa
```

Step 7. Good, now lets substitute the NaN values to 1.0

```
iris['petal_length'] = iris['petal_length'].fillna(1.0)
print("Iris after replacing NaN with 1.0:\n", iris[10:30])
```

```
Iris after replacing NaN with 1.0:
     sepal_length  sepal_width  petal_length  petal_width        class
10           5.4          3.7           1.0          0.2  Iris-setosa
11           4.8          3.4           1.0          0.2  Iris-setosa
12           4.8          3.0           1.0          0.1  Iris-setosa
13           4.3          3.0           1.0          0.1  Iris-setosa
14           5.8          4.0           1.0          0.2  Iris-setosa
15           5.7          4.4           1.0          0.4  Iris-setosa
16           5.4          3.9           1.0          0.4  Iris-setosa
17           5.1          3.5           1.0          0.3  Iris-setosa
18           5.7          3.8           1.0          0.3  Iris-setosa
19           5.1          3.8           1.0          0.3  Iris-setosa
20           5.4          3.4           1.0          0.2  Iris-setosa
21           5.1          3.7           1.0          0.4  Iris-setosa
22           4.6          3.6           1.0          0.2  Iris-setosa
23           5.1          3.3           1.0          0.5  Iris-setosa
24           4.8          3.4           1.0          0.2  Iris-setosa
25           5.0          3.0           1.0          0.2  Iris-setosa
26           5.0          3.4           1.0          0.4  Iris-setosa
27           5.2          3.5           1.0          0.2  Iris-setosa
28           5.2          3.4           1.0          0.2  Iris-setosa
29           4.7          3.2           1.0          0.2  Iris-setosa
```

Step 8. Now let's delete the column class

```
iris = iris.drop(columns=['class'])
print("Iris after dropping class column:\n", iris.head())
```

```
Iris after dropping class column:
    sepal_length  sepal_width  petal_length  petal_width
0           5.1          3.5           1.4          0.2
1           4.9          3.0           1.4          0.2
2           4.7          3.2           1.3          0.2
3           4.6          3.1           1.5          0.2
4           5.0          3.6           1.4          0.2
```

Step 9. Set the first 3 rows as NaN

```
iris.iloc[0:3, :] = np.nan
print("Iris after setting first 3 rows to NaN:\n", iris.head())
```

```
Iris after setting first 3 rows to NaN:
    sepal_length  sepal_width  petal_length  petal_width
0           NaN          NaN           NaN          NaN
1           NaN          NaN           NaN          NaN
2           NaN          NaN           NaN          NaN
3           4.6          3.1           1.5          0.2
4           5.0          3.6           1.4          0.2
```

Step 10. Delete the rows that have NaN

```
iris = iris.dropna()
print("Iris after dropping rows with NaN:\n", iris.head())
```

```
Iris after dropping rows with NaN:
    sepal_length  sepal_width  petal_length  petal_width
3           4.6          3.1           1.5          0.2
4           5.0          3.6           1.4          0.2
5           5.4          3.9           1.7          0.4
6           4.6          3.4           1.4          0.3
7           5.0          3.4           1.5          0.2
```

Step 11. Reset the index so it begins with 0 again

```
iris = iris.reset_index(drop=True)
print("Iris after resetting index:\n", iris.head())
```

```
Iris after resetting index:
    sepal_length  sepal_width  petal_length  petal_width
0           4.6          3.1           1.5          0.2
1           5.0          3.6           1.4          0.2
2           5.4          3.9           1.7          0.4
3           4.6          3.4           1.4          0.3
4           5.0          3.4           1.5          0.2
```

BONUS: Create your own question and answer it.

```
# tính trung bình sepal_length và petal_length theo petal_width_bin
iris['petal_width_bin'] = pd.qcut(iris['petal_width'], q=3, labels=['Low', 'Medium', 'High'])
bonus_result = iris.groupby('petal_width_bin')[['sepal_length', 'petal_length']].mean()
print("Average sepal_length and petal_length by petal_width bin:\n", bonus_result)
```

```
Average sepal_length and petal_length by petal_width bin:
                  sepal_length  petal_length
petal_width_bin
Low                   5.064815      1.570370
Medium                6.042222      4.428889
High                  6.591667      5.539583
<ipython-input-11-643361821ba3>:3: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future ve
  bonus_result = iris.groupby('petal_width_bin')[['sepal_length', 'petal_length']].mean()
```