Wine

Introduction:

This exercise is a adaptation from the UCI Wine dataset. The only pupose is to practice deleting data with pandas.

Step 1. Import the necessary libraries

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

Step 2. Import the dataset from this address.

Step 3. Assign it to a variable called wine

```python
wine = pd.read_csv('https://archive.ics.uci.edu/ml/machine-learning-databases/wine/wine.data')
wine.head()
```

	1	14.23	1.71	2.43	15.6	127	2.8	3.06	.28	2.29	5.64	1.04	3.92	1065
0	1	13.20	1.78	2.14	11.2	100	2.65	2.76	0.26	1.28	4.38	1.05	3.40	1050
1	1	13.16	2.36	2.67	18.6	101	2.80	3.24	0.30	2.81	5.68	1.03	3.17	1185
2	1	14.37	1.95	2.50	16.8	113	3.85	3.49	0.24	2.18	7.80	0.86	3.45	1480
3	1	13.24	2.59	2.87	21.0	118	2.80	2.69	0.39	1.82	4.32	1.04	2.93	735
4	1	14.20	1.76	2.45	15.2	112	3.27	3.39	0.34	1.97	6.75	1.05	2.85	1450

Next steps: Generate code with `wine`   ◉ View recommended plots   New interactive sheet

Step 4. Delete the first, fourth, seventh, nineth, eleventh, thirteenth and fourteenth columns

```python
wine_c = wine.copy()
wine_c.drop(wine_c.columns[[0,3,6,8,10,12,13]], axis=1, inplace=True)
wine_c.head()
```

	14.23	1.71	15.6	127	3.06	2.29	1.04
0	13.20	1.78	11.2	100	2.76	1.28	1.05
1	13.16	2.36	18.6	101	3.24	2.81	1.03
2	14.37	1.95	16.8	113	3.49	2.18	0.86
3	13.24	2.59	21.0	118	2.69	1.82	1.04
4	14.20	1.76	15.2	112	3.39	1.97	1.05

Next steps: Generate code with `wine_c`   ◉ View recommended plots   New interactive sheet

Step 5. Assign the columns as below:

The attributes are (donated by Riccardo Leardi, riclea '@' anchem.unige.it):
1) alcohol
2) malic_acid
3) alcalinity_of_ash
4) magnesium
5) flavanoids
6) proanthocyanins
7) hue

```python
wine_c.columns = ['alcohol', 'malic_acid', 'alcalinity_of_ash', 'magnesium', 'flavanoids', 'proanthocyanins', 'hue']
wine_c.head()
```

	alcohol	malic_acid	alcalinity_of_ash	magnesium	flavanoids	proanthocyanins	hue
0	13.20	1.78	11.2	100	2.76	1.28	1.05
1	13.16	2.36	18.6	101	3.24	2.81	1.03
2	14.37	1.95	16.8	113	3.49	2.18	0.86
3	13.24	2.59	21.0	118	2.69	1.82	1.04
4	14.20	1.76	15.2	112	3.39	1.97	1.05

Next steps: [Generate code with `wine_c`] [⊙ View recommended plots] [New interactive sheet]

⌄ Step 6. Set the values of the first 3 rows from alcohol as NaN

```
wine_c.loc[0:2, 'alcohol'] = np.nan
wine_c.head()
```

	alcohol	malic_acid	alcalinity_of_ash	magnesium	flavanoids	proanthocyanins	hue
0	NaN	1.78	11.2	100	2.76	1.28	1.05
1	NaN	2.36	18.6	101	3.24	2.81	1.03
2	NaN	1.95	16.8	113	3.49	2.18	0.86
3	13.24	2.59	21.0	118	2.69	1.82	1.04
4	14.20	1.76	15.2	112	3.39	1.97	1.05

Next steps: [Generate code with `wine_c`] [⊙ View recommended plots] [New interactive sheet]

⌄ Step 7. Now set the value of the rows 3 and 4 of magnesium as NaN

```
wine_c.loc[2:3, 'magnesium'] = np.nan
wine_c.head()
```

	alcohol	malic_acid	alcalinity_of_ash	magnesium	flavanoids	proanthocyanins	hue
0	NaN	1.78	11.2	100.0	2.76	1.28	1.05
1	NaN	2.36	18.6	101.0	3.24	2.81	1.03
2	NaN	1.95	16.8	NaN	3.49	2.18	0.86
3	13.24	2.59	21.0	NaN	2.69	1.82	1.04
4	14.20	1.76	15.2	112.0	3.39	1.97	1.05

Next steps: [Generate code with `wine_c`] [⊙ View recommended plots] [New interactive sheet]

⌄ Step 8. Fill the value of NaN with the number 10 in alcohol and 100 in magnesium

```
wine_c['alcohol'] = wine_c['alcohol'].fillna(10)
wine_c['magnesium'] = wine_c['magnesium'].fillna(100)
wine_c.head()
```

	alcohol	malic_acid	alcalinity_of_ash	magnesium	flavanoids	proanthocyanins	hue
0	10.00	1.78	11.2	100.0	2.76	1.28	1.05
1	10.00	2.36	18.6	101.0	3.24	2.81	1.03
2	10.00	1.95	16.8	100.0	3.49	2.18	0.86
3	13.24	2.59	21.0	100.0	2.69	1.82	1.04
4	14.20	1.76	15.2	112.0	3.39	1.97	1.05

Next steps: [Generate code with `wine_c`] [⊙ View recommended plots] [New interactive sheet]

⌄ Step 9. Count the number of missing values

```
wine_c.isnull().sum()
```

	0
alcohol	0
malic_acid	0
alcalinity_of_ash	0
magnesium	0
flavanoids	0
proanthocyanins	0
hue	0

dtype: int64

Step 10. Create an array of 10 random numbers up until 10

```
random_indices = np.random.randint(0, len(wine_c), size=10)
print(random_indices)
```

```
[ 48 159 163  23  22  59  78  51 163 126]
```

Step 11. Use random numbers you generated as an index and assign NaN value to each of cell.

```
wine_c.loc[random_indices, :] = np.nan
wine_c.head()
```

	alcohol	malic_acid	alcalinity_of_ash	magnesium	flavanoids	proanthocyanins	hue
0	10.00	1.78	11.2	100.0	2.76	1.28	1.05
1	10.00	2.36	18.6	101.0	3.24	2.81	1.03
2	10.00	1.95	16.8	100.0	3.49	2.18	0.86
3	13.24	2.59	21.0	100.0	2.69	1.82	1.04
4	14.20	1.76	15.2	112.0	3.39	1.97	1.05

Next steps: [Generate code with `wine_c`] [◉ View recommended plots] [New interactive sheet]

Step 12. How many missing values do we have?

```
wine_c.isnull().sum()
```

	0
alcohol	9
malic_acid	9
alcalinity_of_ash	9
magnesium	9
flavanoids	9
proanthocyanins	9
hue	9

dtype: int64

Step 13. Delete the rows that contain missing values

```
wine_c.dropna(inplace=True)
wine_c.head()
```

	alcohol	malic_acid	alcalinity_of_ash	magnesium	flavanoids	proanthocyanins	hue
0	10.00	1.78	11.2	100.0	2.76	1.28	1.05
1	10.00	2.36	18.6	101.0	3.24	2.81	1.03
2	10.00	1.95	16.8	100.0	3.49	2.18	0.86
3	13.24	2.59	21.0	100.0	2.69	1.82	1.04
4	14.20	1.76	15.2	112.0	3.39	1.97	1.05

Next steps: [Generate code with `wine_c`] [◉ View recommended plots] [New interactive sheet]

Step 14. Print only the non-null values in alcohol

```
wine_c['alcohol']
```

	alcohol
0	10.00
1	10.00
2	10.00
3	13.24
4	14.20
...	...
172	13.71
173	13.40
174	13.27
175	13.17
176	14.13

168 rows × 1 columns

dtype: float64

Step 15. Reset the index, so it starts with 0 again

```
wine_c.reset_index(drop=True, inplace=True)
wine_c.head()
```

	alcohol	malic_acid	alcalinity_of_ash	magnesium	flavanoids	proanthocyanins	hue
0	10.00	1.78	11.2	100.0	2.76	1.28	1.05
1	10.00	2.36	18.6	101.0	3.24	2.81	1.03
2	10.00	1.95	16.8	100.0	3.49	2.18	0.86
3	13.24	2.59	21.0	100.0	2.69	1.82	1.04
4	14.20	1.76	15.2	112.0	3.39	1.97	1.05

Next steps: [Generate code with `wine_c`] [◉ View recommended plots] [New interactive sheet]

BONUS: Create your own question and answer it.

```
# q: how many unique values in alcohol column *after filling NaN values*?

# a:
wine_c['alcohol'].nunique()
```

119