

## 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

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

### Step 2. Import the dataset from this [address](https://archive.ics.uci.edu/ml/machine-learning-databases/wine/wine.data).

### Step 3. Assign it to a variable called wine

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



	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
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
172	3	13.71	5.65	2.45	20.5	95	1.68	0.61	0.52	1.06	7.70	0.64	1.74	740
173	3	13.40	3.91	2.48	23.0	102	1.80	0.75	0.43	1.41	7.30	0.70	1.56	750
174	3	13.27	4.28	2.26	20.0	120	1.59	0.69	0.43	1.35	10.20	0.59	1.56	835
175	3	13.17	2.59	2.37	20.0	120	1.65	0.68	0.53	1.46	9.30	0.60	1.62	840
176	3	14.13	4.10	2.74	24.5	96	2.05	0.76	0.56	1.35	9.20	0.61	1.60	560

177 rows × 14 columns

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

```
wine = wine.drop(wine.columns[[0,3,6,8,10,12,13]], axis = 1)
wine
```



	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
...	...	...	...	...	...	...	...
172	13.71	5.65	20.5	95	0.61	1.06	0.64
173	13.40	3.91	23.0	102	0.75	1.41	0.70
174	13.27	4.28	20.0	120	0.69	1.35	0.59
175	13.17	2.59	20.0	120	0.68	1.46	0.60
176	14.13	4.10	24.5	96	0.76	1.35	0.61

177 rows × 7 columns

### ✓ 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

```
wine.columns = ['alcohol', 'malic_acid', 'alcalinity_of_ash', 'magnesium', 'flavanoids', 'proanthocyanins', 'hue']  
wine
```



	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
...	...	...	...	...	...	...	...
172	13.71	5.65	20.5	95	0.61	1.06	0.64
173	13.40	3.91	23.0	102	0.75	1.41	0.70
174	13.27	4.28	20.0	120	0.69	1.35	0.59
175	13.17	2.59	20.0	120	0.68	1.46	0.60
176	14.13	4.10	24.5	96	0.76	1.35	0.61

177 rows × 7 columns

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

```
wine.iloc[0:3, 0] = np.nan
wine
```



	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
...	...	...	...	...	...	...	...
172	13.71	5.65	20.5	95	0.61	1.06	0.64
173	13.40	3.91	23.0	102	0.75	1.41	0.70
174	13.27	4.28	20.0	120	0.69	1.35	0.59
175	13.17	2.59	20.0	120	0.68	1.46	0.60
176	14.13	4.10	24.5	96	0.76	1.35	0.61

177 rows × 7 columns

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

```
wine.iloc[2:4, 3] = np.nan
wine
```



	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
...	...	...	...	...	...	...	...
172	13.71	5.65	20.5	95.0	0.61	1.06	0.64
173	13.40	3.91	23.0	102.0	0.75	1.41	0.70
174	13.27	4.28	20.0	120.0	0.69	1.35	0.59
175	13.17	2.59	20.0	120.0	0.68	1.46	0.60
176	14.13	4.10	24.5	96.0	0.76	1.35	0.61

177 rows × 7 columns

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

```
wine.loc[wine['alcohol'] == 10, 'alcohol'] = np.nan
wine.loc[wine['magnesium'] == 100, 'magnesium'] = np.nan
wine
```



	alcohol	malic_acid	alcalinity_of_ash	magnesium	flavanoids	proanthocyanins	hue
0	NaN	1.78	11.2	NaN	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
...	...	...	...	...	...	...	...
172	13.71	5.65	20.5	95.0	0.61	1.06	0.64
173	13.40	3.91	23.0	102.0	0.75	1.41	0.70
174	13.27	4.28	20.0	120.0	0.69	1.35	0.59
175	13.17	2.59	20.0	120.0	0.68	1.46	0.60
176	14.13	4.10	24.5	96.0	0.76	1.35	0.61

177 rows × 7 columns

## ✓ Step 9. Count the number of missing values

```
wine.isnull().sum()
```



	0
alcohol	3
malic_acid	0
alcalinity_of_ash	0
magnesium	5
flavanoids	0
proanthocyanins	0
hue	0

dtype: int64

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

```
arr = np.random.randint(10, size = 10)
arr
```



```
array([0, 4, 0, 2, 6, 0, 7, 3, 3, 2])
```

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

```
wine.iloc[arr] = np.nan
wine
```



	alcohol	malic_acid	alcalinity_of_ash	magnesium	flavanoids	proanthocyanins	hue
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	NaN	2.36	18.6	101.0	3.24	2.81	1.03
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...	...
172	13.71	5.65	20.5	95.0	0.61	1.06	0.64
173	13.40	3.91	23.0	102.0	0.75	1.41	0.70
174	13.27	4.28	20.0	120.0	0.69	1.35	0.59
175	13.17	2.59	20.0	120.0	0.68	1.46	0.60
176	14.13	4.10	24.5	96.0	0.76	1.35	0.61

177 rows × 7 columns

## ✓ Step 12. How many missing values do we have?

```
sum = 0
for i in wine.columns:
    sum += wine[i].isnull().sum()
print(sum)
```



45

## ✓ Step 13. Delete the rows that contain missing values

```
wine = wine.dropna(axis = 0, how = "any")
wine
```




	alcohol	malic_acid	alcalinity_of_ash	magnesium	flavanoids	proanthocyanins	hue
5	14.39	1.87	14.6	96.0	2.52	1.98	1.02
8	13.86	1.35	16.0	98.0	3.15	1.85	1.01
9	14.10	2.16	18.0	105.0	3.32	2.38	1.25
10	14.12	1.48	16.8	95.0	2.43	1.57	1.17
11	13.75	1.73	16.0	89.0	2.76	1.81	1.15
...	...	...	...	...	...	...	...
172	13.71	5.65	20.5	95.0	0.61	1.06	0.64
173	13.40	3.91	23.0	102.0	0.75	1.41	0.70
174	13.27	4.28	20.0	120.0	0.69	1.35	0.59
175	13.17	2.59	20.0	120.0	0.68	1.46	0.60
176	14.13	4.10	24.5	96.0	0.76	1.35	0.61

168 rows × 7 columns

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

```
wine['alcohol'].dropna()
```




	alcohol
5	14.39
8	13.86
9	14.10
10	14.12
11	13.75
...	...
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 = wine.reset_index(drop = True)
wine
```



	alcohol	malic_acid	alcalinity_of_ash	magnesium	flavanoids	proanthocyanins	hue
0	14.39	1.87	14.6	96.0	2.52	1.98	1.02
1	13.86	1.35	16.0	98.0	3.15	1.85	1.01
2	14.10	2.16	18.0	105.0	3.32	2.38	1.25
3	14.12	1.48	16.8	95.0	2.43	1.57	1.17
4	13.75	1.73	16.0	89.0	2.76	1.81	1.15
...	...	...	...	...	...	...	...
163	13.71	5.65	20.5	95.0	0.61	1.06	0.64
164	13.40	3.91	23.0	102.0	0.75	1.41	0.70
165	13.27	4.28	20.0	120.0	0.69	1.35	0.59
166	13.17	2.59	20.0	120.0	0.68	1.46	0.60
167	14.13	4.10	24.5	96.0	0.76	1.35	0.61

168 rows × 7 columns