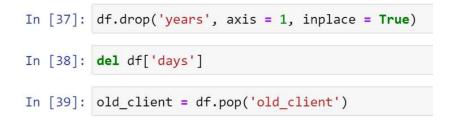


# Pandas – gym example – add and remove columns

Create a new column with the number of years at the gym and insert it at position 3



## Remove the columns years, days and old\_client

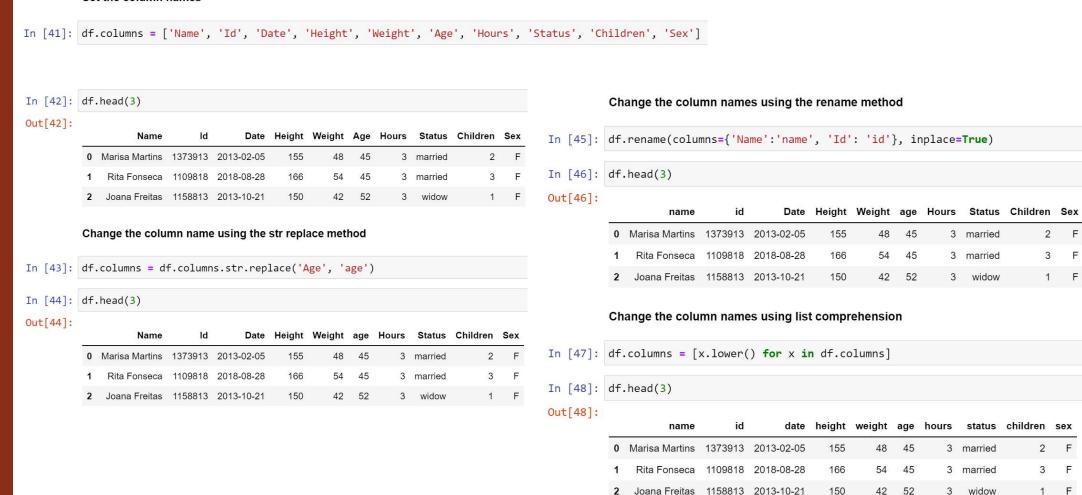


In [40]:	df.head(3)													
Out[40]:		name	id	date	height	weight	age	hours	status	children	sex			
	0	Marisa Martins	1373913	2013-02-05	155	48	45	3	married	2	F			
	1	Rita Fonseca	1109818	2018-08-28	166	54	45	3	married	3	F			
	2	Joana Freitas	1158813	2013-10-21	150	42	52	3	widow	1	F			



# Pandas – gym example – column names

#### Set the column names





## Pandas – gym example – row values

### Assign a new set of values to a row





# Pandas – gym example – filters

## Select the customers over 55 years old

In [53]: df.loc[df['age'] > 55]

Out[53]:

	name	id	date	height	weight	age	hours	status	children	sex
3	Joana Goncalves	1566515	2015-11-16	161	49.0	59.0	2.0	married	2.0	F
7	Rita Cruz	1930916	2016-10-11	168	52.0	56.0	3.0	married	2.0	F
15	Susana Madeira	1436908	2008-09-05	160	49.0	56.0	2.0	divorced	2.0	F
19	Catarina Goncalves	1055806	2006-09-21	168	53.0	59.0	2.0	widow	3.0	F
32	Joana Marinho	1466415	2015-11-26	164	51.0	57.0	1.0	married	3.0	F
43	Catarina Carvalho	1058304	2004-09-25	150	45.0	59.0	2.0	married	4.0	F
58	Manuel Freitas	1658815	2015-11-06	170	51.0	57.0	1.0	widow	2.0	М

## Select the customers over 55 years old and weight over 50kg

In [54]: df.loc[(df['age'] > 55) & (df['weight'] > 50)]

Out[54]:

	name	id	date	height	weight	age	hours	status	children	sex
7	Rita Cruz	1930916	2016-10-11	168	52.0	56.0	3.0	married	2.0	F
19	Catarina Goncalves	1055806	2006-09-21	168	53.0	59.0	2.0	widow	3.0	F
32	Joana Marinho	1466415	2015-11-26	164	51.0	57.0	1.0	married	3.0	F
58	Manuel Freitas	1658815	2015-11-06	170	51.0	57.0	1.0	widow	2.0	М

#### Select the customers over 55 or under 24 years old. View name and age

In [55]: df.loc[(df['age'] > 57) | (df['age'] < 24), ['name', 'age']]</pre>

Out[55]:

	name	age
3	Joana Goncalves	59.0
16	Francisco Pinho	23.0
19	Catarina Goncalves	59.0
38	Francisco Madeira	23.0
43	Catarina Carvalho	59.0
44	Francisco Carvalho	23.0

#### Select the customers with status: divorced or widow

In [56]: df.loc[df['status'].isin(['divorced', 'widow'])]

Out[56]:

	name	id	date	height	weight	age	hours	status	children	sex
2	Joana Freitas	1158813	2013-10-21	150	42.0	52.0	3.0	widow	1.0	F
10	Manuel Marinho	1417018	2018-06-20	173	57.0	40.0	3.0	divorced	2.0	М
15	Susana Madeira	1436908	2008-09-05	160	49.0	56.0	2.0	divorced	2.0	F
19	Catarina Goncalves	1055806	2006-09-21	168	53.0	59.0	2.0	widow	3.0	F
58	Manuel Freitas	1658815	2015-11-06	170	51.0	57.0	1.0	widow	2.0	М

#### Select the customers with name Freitas

In [57]: df.loc[df['name'].str.contains('Freitas',na=False)]

Out[57]:

	name	id	date	height	weight	age	hours	status	children	sex
2	Joana Freitas	1158813	2013-10-21	150	42.0	52.0	3.0	widow	1.0	F
6	Florbela Freitas	1071208	2008-09-26	166	53.0	28.0	5.0	single	0.0	F
11	Joao Freitas	1642316	2016-10-23	165	50.0	54.0	3.0	married	2.0	М
13	Francisco Freitas	1560118	2018-07-15	152	53.0	41.0	3.0	married	2.0	M
24	Catarina Freitas	1105321	2021-10-07	166	70.0	40.0	4.0	single	0.0	F
58	Manuel Freitas	1658815	2015-11-06	170	51.0	57.0	1.0	widow	2.0	M



# Pandas – gym example – apply, value\_counts()

## Apply a function to all the elements of a Series



## Groupby can be used to count values by group

```
In [62]: df.groupby('sex')['status'].value counts()
Out[62]: sex status
              married
                           19
                            7
              single
              widow
              divorced
                            1
              married
                           16
              single
                           13
              divorced
              widow
         Name: status, dtype: int64
```

### Write a DataFrame to a .csv and Excel files

```
In [63]: df.to_csv('gym_new.csv', sep = ';', index=False)
In [64]: df.to_excel('gym_new.xlsx', index=False)
```