1 Sorting and ranking with Pandas

1.1 Sorting

Pandas allows easy and flexible sorting.

As usual, let's first build a dataframe:

```
import pandas as pd
df = pd.DataFrame()

names = ['Gandolf','Gimli','Frodo','Legolas','Bilbo']
types = ['Wizard','Dwarf','Hobbit','Elf','Hobbit']
magic = [10, 1, 4, 6, 4]
aggression = [7, 10, 2, 5, 1]
stealth = [8, 2, 5, 10, 5]

df['names'] = names
df['type'] = types
df['magic_power'] = magic
df['aggression'] = aggression
df['stealth'] = stealth
```

And now let's sort first by magic power and then (in reverse order aggression.

```
new_df = df.sort_values(['magic_power','aggression'], ascending=[False,True])
print (new_df)

OUT:

     names    type  magic_power  aggression  stealth
0  Gandolf  Wizard           10           7        8
3  Legolas     Elf            6           5       10
4    Bilbo  Hobbit            4           1        5
2    Frodo  Hobbit            4           2        5
1    Gimli   Dwarf            1          10        2
```

Usually it is fine to use the default sorting method. Sometimes though you may wish to do a series of sequential sorts where you maintain the previous order within the sorted the dataframe. In that case use a mergesort by passing *kind = 'mergesort'* as one of the arguments.

We can use *sort_index* to sort by the index field. Let's sort our new dataframe by reverse index order:

```
print (new_df.sort_index(ascending=False))
i
     names    type  magic_power  aggression  stealth
4    Bilbo  Hobbit            4           1        5
3  Legolas     Elf            6           5       10
2    Frodo  Hobbit            4           2        5
1    Gimli   Dwarf            1          10        2
0  Gandolf  Wizard           10           7        8
```

1.2 Ranking

Pandas allows easy ranking of dataframes by a single column. Where two values are identical the result is the average of the number of ranks they would cover. Notice that a higher number is a higher rank.

```
i
print (df['magic_power'].rank())

OUT:
```

```
0    5.0
1    1.0
2    2.5
3    4.0
4    2.5
Name: magic_power, dtype: float64
```

Pandas does not offer a direct method for ranking using multiple columns. One way would be to sort the dataframe, reset the index with *df.reset_index()* and compare the index values to the original table.