Introduction to Computer Programming

Week 7.2: Modules for reading & writing files

Writing csv files

Example: Use the `csv` module to write the header and first row of the high score table shown to a .txt file.

As a delimiter other than the default `','` is being used, we must specify this when creating the writer object.

Place	Name	Score
1	Elena	550
2	Sajid	480
3	Tom	380
4	Farhad	305
5	Manesha	150

In [11]:

```python
import csv

with open('sample_data/scores_.txt', 'w') as f:      # open file

    w = csv.writer(f, delimiter=' ')                 # writer obje

    w.writerow(['place', 'name', 'score'])           # list to row

    w.writerow([1,'Elena', 550])
```

__Example:__ Write the high score table data to a csv file, scores_.csv

`writerows` can be used to write a data struct

In [12]:
```python
header = ['place', 'name', 'score']

# data is list of lists
data = [[1, 'Elena', 550],
        [2, 'Sajid', 480],
        [3, 'Tom', 380],
        [4, 'Farhad', 305],
        [5, 'Manesha', 150]
       ]
```

In [13]:
```python
import csv

with open('sample_data/scores_.csv', 'w') as f:

    w = csv.writer(f)

    w.writerow(header)

    w.writerows(data)
```

Reading csv files

Example: Read the file scores_.csv and print the third line.

In [14]:
```python
import csv

with open ('sample_data/scores_.csv') as f:

    r = csv.reader(f, delimiter= ' ')

    r = list(r)

#     for l in r:
#         print(l)

    print(r[1])
```

```
['1,Elena,550']
```

Reading and writing csv files

The same mode specifiers are used as for .txt files.

A `reader` *and* `writer` object are created.

Example: Print the data in the file sample_data/scores.csv and add a new entry.

In [15]:
```python
# choose append mode as we will write then read

# what if we wanted to read then append?

with open('sample_data/scores_.csv', 'a+') as f:

    r = csv.reader(f)

    w = csv.writer(f)

    w.writerow([10, 'Lois', 70])

    f.seek(0)

    for line in r:
        print(line)
```

```
['place', 'name', 'score']
['1', 'Elena', '550']
['2', 'Sajid', '480']
['3', 'Tom', '380']
['4', 'Farhad', '305']
['5', 'Manesha', '150']
['6', 'Carl', '100']
['7', 'Theo', '105']
['8', 'Mark', '75']
['9', 'Grace', '50']
['10', 'Lois', '70']
['10', 'Lois', '70']
['10', 'Lois', '70']
['10', 'Lois', '70']
```

In [ ]: