OOP inheritance – exercises

# Exercise 1

This exercise deals with the *Person* class we’ve defined earlier. Define a sub-class of *Person* called *Israeli*, and add to it the following properties:

* *served* – Boolean, initialized with *False*
* *serve(self, keva=0)* – method that does the following:
  + Updates *served*
  + Calculates the serving time of the person – *2+keva* for females and *3+keva* for males
  + Calls *grow()* for each year during the service time
* *\_\_str\_\_(self,…)* (modified) – add the *served* information

Test your class.

# Exercise 2

This exercise deals with the Queue class we’ve defined earlier.

## Part I

Define a sub-class of Queue called GroupsQueue that supports the insertion of a bunch of elements (one after the other) to the queue (dequeuer is still one element at a time). For that you have to override only the *enqueue(self, element)* method to the form *enqueue(self, \*elements)*.

## Part II

The file “queue3.txt” contains a list of patients by the time of their arrival to a clinic, but now some of them arrive in groups. Assume that the order of the names in the file can be used for the order of enqueuing and discover who is the patient that suffered the longest line.

Tip: You may want to use the string method [strip()](https://docs.python.org/2/library/string.html#string.strip) for parsing the file.