Prelab 12

In this prelab we shall describe how a class is built and you will then be required to write the main function to use the methods defined within the class.

Remember that the Dice object was described in class in the following way:

```python
#
# dice.py
# Class definition for a multi-sided dice.
#

from random import randrange

class Dice:

    def __init__(self, sides):
        self.sides = sides
        self.value = 1

    def roll(self):
        self.value = randrange(1, self.sides+1)

    def getValue(self):
        return self.value

    def setValue(self, value):
        self.value = value

def Main():
    d = Dice(6)
    d.roll()
    print("Value: ", d.getValue())

if __name__ == '__main__':
    Main()
```

Then write a separate dice game program that rolls the dice with you.

Here's how you will need to do it:

1. Roll the dice on his part after the user presses <Enter>
2. Roll the dice on your part after the user presses <Enter>
3. If the sum of dice on his part are greater than sum of dice on your part then print “You Win!”
4. If the sum of dice on your part are greater than sum of dice on his part then print “I Win!”
5. Ask the user if they want to play again - “Do you want to play again (yes/no)?”

6. If yes then repeat the process, if no then close the program

**Here is an example of the input/output:**

```
Press <Enter> to roll the dice ...
You got  6  and  6
Press <Enter> for my turn to roll the dice ...
I got  2  and  5
You win!!!
Do you want to play again (yes/no)? yes
Press <Enter> to roll the dice ...
You got  2  and  1
Press <Enter> for my turn to roll the dice ...
I got  1  and  6
I win!!!
Do you want to play again (yes/no)? yes
Press <Enter> to roll the dice ...
You got  5  and  4
Press <Enter> for my turn to roll the dice ...
I got  6  and  3
We are even!!
Do you want to play again (yes/no)? no
```

**Stack**

A stack or LIFO (last in, first out) is an abstract data type that serves as a collection of elements, with two principal operations: push, which adds an element to the collection, and pop, which removes the last element that was added.