Introduction to Programming with Python

Weekly Programming Assignment – Week 4

*All solution files [**Exercises 1 – 5**] must be submitted at CodeGrade enabled Link for grading. The solution code for exercise 6 must be uploaded in Moodle as **"ex6_week4.txt"** submission page for manual grading. More details at page no 4.*

All solutions must be uploaded on or before 6th October 2021 at 11:59 PM

Exercise 1

Write a program that prints the following table which lists miles (1 to 10) and equivalent kilometers. The sample run is here. (Use for loop only)

```
Miles    Kilometers
1         1.609
2         3.218
3         4.827
4         6.436
5         8.045
6         9.654
7         11.263
8         12.872
9         14.481
10        16.09
```

Exercise 2

Write code that uses **loops to list all the leap years** from year 1900 to 2021. In the end, **print out the number of leap years** that are listed. Every year that is exactly divisible by four is a leap year, except for years that are exactly divisible by 100; the centurial years that are exactly divisible by 400 are still leap years. For example, the year 1900 is not a leap year; the year 2000 is a leap year. Sample run is here.

```
1904
1908
1912
1916
1920
1924
1928
1932
1936
1940
1944
1948
1952
1956
```

1960
1964
1968
1972
1976
1980
1984
1988
1992
1996
2000
2004
2008
2012
2016
2020
Total number of leap years from 1900 to 2021: 30

Exercise 3

Write a program that accepts any integer as input until n times. Then print the total number of positive, negative and zeros entered as input. [should use while loop only]

```
>>> %Run Ex1_week4.py
  Number of time the input to be asked:5
  Enter any integer as input:0
  Enter any integer as input:2
  Enter any integer as input:3
  Enter any integer as input:-4
  Enter any integer as input:-1
  Positive: 2
  Negative: 2
  Zeros: 1
```

Exercise 4

Use a nested loop to print the following pattern:

6 5 4 3 2 1
5 4 3 2 1
4 3 2 1
3 2 1
2 1
1

Exercise 5

Write a program that prompts the user to enter the amount, the annual interest rate, and the number of months and displays the amount in the savings account after the given month.
[Hint: you may be required to use loop, decision statements and round()]

For example, suppose you save `100€` each month into a savings account with the annual interest rate `5%`. So, the monthly interest rate is `0.05/12 = 0.00417`.

After the first month, the amount in the account becomes
`100 * (1 + 0.00417) = 100.417`

After the second month, the amount in the account becomes
`(100 + 100.417) * (1 + 0.00417) = 201.252`

After the third month, the amount in the account becomes
`(100 + 201.252) * (1 + 0.00417) = 302.507`

and so on. The sample run is here.

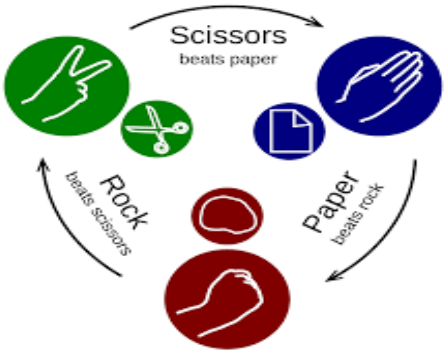
```
Enter the amount you save per month:100
Enter the interest rate per year:5
Enter the number of months:6

Month     Amount in the account
1         100.42
2         201.25
3         302.51
4         404.18
5         506.28
6         608.81
```

It should be noted, the values given above are demonstration purposes only. Your code will be checked with different set of inputs for grading.

Exercise 6

Write a program that lets the user plays the popular scissor-rock-paper game until the player enters “no” to end the game.



As shown in the figures, A scissor can cut a paper, a rock can knock a scissor and a paper can wrap a rock. The program randomly generates a number **0, 1, or 2** representing scissor, rock, and paper. The program prompts the player to enter a number **0, 1, or 2** and displays a message including whether the player or the computer wins, loses, or draws. The play must be continued until the player enters **"no"**. The program ends with closing message "**Thanks for playing with me".** Sample run is here:

```
enter 0-Scissor; 1-rock or 2-paper: 2
The computer is Scissor You are Paper #COMPUTER WON and you lost#

Do you want to play again (type no to exit)?:y

enter 0-Scissor; 1-rock or 2-paper: 0
The computer is Paper You are Scissor #YOU WON#

Do you want to play again (type no to exit)?:y

enter 0-Scissor; 1-rock or 2-paper: 0
The computer is Paper You are Scissor #YOU WON#

Do you want to play again (type no to exit)?:y

enter 0-Scissor; 1-rock or 2-paper: 1
The computer is Scissor You are Rock #YOU WON#

Do you want to play again (type no to exit)?:y

enter 0-Scissor; 1-rock or 2-paper: 2
The computer is Paper You are Paper it is a draw

Do you want to play again (type no to exit)?:no

Thanks for playing with me
```

*The solution code for exercise 6 must be uploaded in Moodle as "**ex6_week4.txt**". **You are required to run your submitted code in the computer during the tutorial session of Week5 (After Quiz 2) for direct and instant manual grading. You will be asked a few questions about this assignment to assure that it was done by you.***

Exercise / task Number	Codegrade link_Moodle for file solution files upload	Points / Marks
1	Exercise1_Week 4	10
2	Exercise2_Week 4	10
3	Exercise3_Week 4	10
4	Exercise4_Week 4	10
5	Exercise5_Week 4	10
6*	Exercise6_Week 4	20
* Exercise 6 will be marking during tutorial session -Week 5 for manual grading.		