WORKSHEET 3 PYTHON

Q1 to Q8 have only one correct answer. Choose the correct option to answer your question.

1. Which of the following will raise a value error in python?
   A) int(32) B) int(3.2)
   C) int(-3.2) D) int('32')
2. What will be the output of round(3.567)?
   A) 3.5 B) 3.0
   C) 4 D) 3
3. How is the function pow(a,b,c) evaluated in python?
   A) a**b**c **B) (a**b)%c**
   C) (a**b)*c D) (a**b)**c
4. What will be the output of **print(type(type(int)))** in python 3?
   A) <class 'type'> **B) <type 'type'>**
   C) <class 'int'> D) <type 'int'>
5. What will be the output of **ord(chr(65))**?
   A) **'A'** B) 'a'
   C) 65 D) TypeError
6. What is called when a function is defined inside a class?
   A) Module B) Function
   C) _init_ function **D) Method**
7. What will be the output of **all([1, 0, 5 ,7])**?
   A) 0 B) False
   C) True **D) error**
8. Is the output of the function abs() the same as that of the function math.fabs()?
   A) Always **B) Sometimes**
   C) Never D) None of these

Q9 and Q10 have multiple correct answers. Choose all the correct options to answer your question.

9. Select all correct float numbers in python?
   A) -68.7e100 B) 42e3
   C) 4.2038 **D) 3.0**
10. Which of the following is(are) correct statement(s) in python?
    A) You can pass positional arguments in any order.
    B) You can pass keyword arguments in any order.
    C) You can call a function with positional and keyword arguments.
    D) Positional arguments must be before keyword arguments in a function call

Q11 to Q15 are programming questions. Answer them in Jupyter Notebook.

11. Write a python function print pyramid of stars. Level of the pyramid should be taken as an input from the user. E.g.
    Input = 5
    Output:

Ans:-

n [2]:

```
# Python 3.x code to demonstrate star pattern

# Function to demonstrate printing pattern triangle
def triangle(n):
    # number of spaces
    k = n - 1

    # outer loop to handle number of rows
    for i in range(0, n):

        # inner loop to handle number spaces
        # values changing acc. to requirement
        for j in range(0, k):
            print(end=" ")

        # decrementing k after each loop
        k = k - 1

        # inner loop to handle number of columns
        # values changing acc. to outer loop
        for j in range(0, i+1):

            # printing stars
            print("* ", end="")

        # ending line after each row
        print("\r")

# Driver Code
n = 5
triangle(n)
```

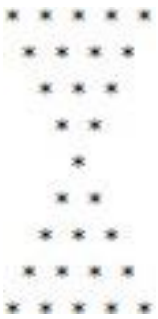
```
    *
   * *
  * * *
 * * * *
* * * * *
```

12. Write a python function print Hourglass pattern.

E.g.

Input = 5

Output:



Ans:-

```
# Python3 code for hour glass pattern

# Function definition
def pattern(rows_no):

    # for loop for printing upper half
    for i in range(1, rows_no + 1):

        # printing i spaces at the
        # beginning of each row
        for k in range(1, i):
            print(" ", end = "")

        # printing i to rows value
        # at the end of each row
        for j in range(i, rows_no + 1):
            print(j, end = " ")

        print()

    # for loop for printing lower half
    for i in range(rows_no - 1, 0, -1):

        # printing i spaces at the
        # beginning of each row
        for k in range(1, i):
            print(" ", end = "")

        # printing i to rows value
        # at the end of each row
        for j in range(i, rows_no + 1):
            print(j, end = " ")

        print()

# Driver code

# taking rows value from the user
rows_no = 5

pattern(rows_no)

# This code is contributed
# by ihritik
```

```
1 2 3 4 5
 2 3 4 5
  3 4 5
   4 5
    5
   4 5
  3 4 5
 2 3 4 5
1 2 3 4 5
```

13. Write a python function to print Pascal's Triangle. The number of levels in the triangle must be taken as input by the user. E.g.

Input = 5

Output:

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

Ans:-

[6]:
```
# Print Pascal's Triangle in Python

# input n
n = 5

for i in range(1, n+1):
    for j in range(0, n-i+1):
        print(' ', end='')

    # first element is always 1
    C = 1
    for j in range(1, i+1):

        # first value in a line is always 1
        print(' ', C, sep='', end='')

        # using Binomial Coefficient
        C = C * (i - j) // j
    print()
```

```
     1
    1 1
   1 2 1
  1 3 3 1
 1 4 6 4 1
```

14. Write a python function to print Diamond Shaped Pattern shown below. Function must take integer input which represents the number of stars in the middle most line. E.g.:
Input = 5
Output:

Ans:-

In [7]:

```
# Python program to
# print Diamond shape

# Function to print
# Diamond shape
def Diamond(rows):
    n = 0
    for i in range(1, rows + 1):
        # loop to print spaces
        for j in range (1, (rows - i) + 1):
            print(end = " ")

        # loop to print star
        while n != (2 * i - 1):
            print("*", end = "")
            n = n + 1
        n = 0

        # line break
        print()

    k = 1
    n = 1
    for i in range(1, rows):
        # loop to print spaces
        for j in range (1, k + 1):
            print(end = " ")
        k = k + 1

        # loop to print star
        while n <= (2 * (rows - i) - 1):
            print("*", end = "")
            n = n + 1
        n = 1
        print()

# Driver Code
# number of rows input
rows = 5
Diamond(rows)
```

```
    *
   ***
  *****
 *******
*********
 *******
  *****
   ***
    *
```

15. Write a python function to print Diamond Shaped Character Pattern shown below. Function must take integer input within range 1 to 26, which represents the rank of the alphabet. E.g.:
    Input = 5
    Output:

```
    A
   A B
  A B C
 A B C D
A B C D E
 A B C D
  A B C
   A B
    A
```

Ans:-

In [8]:

```
# Number of rows
row = 5

# Upper part of hollow diamond
for i in range(1, row+1):
    for j in range(1,row-i+1):
        print(" ", end="")
    for j in range(1, 2*i):
        ch = chr(64+i)
        if j==1 or j==2*i-1:
            print(ch, end="")
        else:
            print(" ", end="")
    print()

# Lower part of hollow diamond
for i in range(row-1,0, -1):
    for j in range(1,row-i+1):
        print(" ", end="")
    for j in range(1, 2*i):
        if j==1 or j==2*i-1:
            ch = chr(64+i)
            print(ch, end="")
        else:
            print(" ", end="")
    print()
```

```
    A
   B B
  C   C
 D     D
E       E
 D     D
  C   C
   B B
    A
```