

Question 1

What data type is each of the following (evaluate where necessary)?

5 5.0 5 > 1 '5' 5 * 2 '5' * 2 '5' + '2' 5 / 2 5 % 2 {5, 2, 1} 5 == 3 Pi (the number)

Ans for Question 1

5 = Integer 5.0 = Float 5 > 1 = Boolean '5' = String 5 * 2 = Integer '5' * 2 = String '5' + '2' = String 5 / 2 = Float 5 % 2 = Integer {5, 2, 1} = Set 5 == 3 = Boolean Pi (the number) = Float

Question 2

Write (and evaluate) python expressions that answer these questions: a. How many letters are there in 'Supercalifragilisticexpialidocious'? b. Does 'Supercalifragilisticexpialidocious' contain 'ice' as a substring? c. Which of the following words is the longest: Supercalifragilisticexpialidocious, Honorificabilitudinitatibus, or Bababadalgharaghtakamminarronkn? d. Which composer comes first in the dictionary: 'Berlioz', 'Borodin', 'Brian', 'Bartok', 'Bellini', 'Buxtehude', 'Bernstein'. Which one comes last?

Ans to question 2

```
In [8]: s = 'Supercalifragilisticexpialidocious'
t = len(s)
print(f"Number of letters: {t}")
```

Number of letters: 34

```
In [9]: a = 'Supercalifragilisticexpialidocious'
b = 'ice'
c = a in b
print(f"Contains 'ice': {c}")
```

Contains 'ice': False

```
In [11]: a = ['Supercalifragilisticexpialidocious', 'Honorificabilitudinitatibus', 'Bababada']
b = max(a, key=len)
print(f"Longest word: {b}")
```

Longest word: Supercalifragilisticexpialidocious

```
In [14]: a = ['Berlioz', 'Borodin', 'Brian', 'Bartok', 'Bellini', 'Buxtehude', 'Bernstein']
b = min(a)
c = max(a)
print(f"First : {b}, Last : {c}")
```

First : Bartok, Last : Buxtehude

Question 3

Implement function `triangleArea(a,b,c)` that takes as input the lengths of the 3 sides of a triangle and returns the area of the triangle. By Heron's formula, the area of a triangle with side lengths a , b , and c is $s(s - a)(s - b)(s - c)$, where $s = (a+b+c)/2$.

```
In [16]: def triangleArea():
    #asking user to give any choice of their input as a number
    a = float(input("Enter the length of side a: "))
    b = float(input("Enter the length of side b: "))
    c = float(input("Enter the length of side c: "))

    # variable s is semi perimeter
    s = (a + b + c) / 2

    #applying formula for area
    area = (s * (s - a) * (s - b) * (s - c)) ** 0.5

    return area
result = triangleArea()
print(f"The area of the triangle is: {result}")
```

```
Enter the length of side a: 2
Enter the length of side b: 2
Enter the length of side c: 2
The area of the triangle is: 1.7320508075688772
```

Question 4

Write a program in python to separate odd and even integers in separate arrays. Go to the editor Test Data : Input the number of elements to be stored in the array :5 Input 5 elements in the array : element - 0 : 25 element - 1 : 47 element - 2 : 42 element - 3 : 56 element - 4 : 32

Ans for question 4

```
In [17]: # asking the number of elements from the user of their choice
n = int(input("Enter the number of elements: "))
arr = []
for i in range(n):
    e = int(input(f"Element - {i}: "))
    arr.append(e)

odd_numbers = []
even_numbers = []

for num in arr:
    if num % 2 != 0:
        odd_numbers.append(num)
```

```

else:
    even_numbers.append(num)

print("Odd numbers:", odd_numbers)
print("Even numbers:", even_numbers)

```

```

Enter the number of elements: 8
Element - 0: 1
Element - 1: 3
Element - 2: 5
Element - 3: 4
Element - 4: 49
Element - 5: 84
Element - 6: 48
Element - 7: 51
Odd numbers: [1, 3, 5, 49, 51]
Even numbers: [4, 84, 48]

```

Question 5

- Write a program in python to separate odd and even integers in separate arrays. Go to the editor. Write a function `inside(x,y,x1,y1,x2,y2)` that returns True or False depending on whether the point (x,y) lies in the rectangle with lower left corner $(x1,y1)$ and upper right corner $(x2,y2)$.
- Use function `inside()` from part a. to write an expression that tests whether the point $(1,1)$ lies in both of the following rectangles: one with lower left corner $(0.3, 0.5)$ and upper right corner $(1.1, 0.7)$ and the other with lower left corner $(0.5, 0.2)$ and upper right corner $(1.1, 2)$.

Ans for question 5

```

In [22]: def inside(x, y, x1, y1, x2, y2):
          return x1 <= x <= x2 and y1 <= y <= y2

          #taking input from the user
          x = float(input(""))
          y = float(input(""))
          x1 = float(input(""))
          y1 = float(input(""))
          x2 = float(input(""))
          y2 = float(input(""))

          result = inside(x, y, x1, y1, x2, y2)

          print("are the point inside the rectangle?", result)

```

```
9
8
2
5
6
7
are the point inside the rectangle? False
```

```
In [23]: r1 = (0.3, 0.5, 1.1, 0.7)
r2 = (0.5, 0.2, 1.1, 2)

point_inside_r1 = inside(1, 1, *r1)
point_inside_r2 = inside(1, 1, *r2)

a = point_inside_r1 and point_inside_r2

print(f"Is the point (1, 1) inside both rectangles? {a}")
```

Is the point (1, 1) inside both rectangles? False

Question 6

You can turn a word into pig-Latin using the following two rules (simplified):

- If the word starts with a consonant, move that letter to the end and append 'ay'. For example, 'happy' becomes 'appyhay' and 'pencil' becomes 'encilpay'.
- If the word starts with a vowel, simply append 'way' to the end of the word. For example, 'enter' becomes 'enterway' and 'other' becomes 'otherway'.

For our purposes, there are 5 vowels: a, e, i, o, u (so we count y as a consonant). Write a function `pig()` that takes a word (i.e., a string) as input and returns its pigLatin form. Your function should still work if the input word contains upper case characters. Your output should always be lower case however.

Ans for question 6

```
In [24]: def pig(word):
word = word.lower()

if word[0] in 'aeiou':
    return word + 'way'
else:
    return word[1:] + word[0] + 'ay'

a = input("Enter a word: ")

b = pig(a)
print(f"The Pig Latin form of '{a}' is: {b}")
```

Enter a word: Happy
The Pig Latin form of 'Happy' is: appyhay

Question 7

File bloodtype1.txt records blood-types of patients (A, B, AB, O or OO) at a clinic. Write a function bldcount() that reads the file with name name and reports (i.e., prints) how many patients there are in each bloodtype

Answer for question 7

```
In [45]: file = open("bloodtype1.txt", "r")
#initializing counts for each blood type
A = 0
B = 0
O = 0
OO = 0
AB = 0

for line in file:
    #iterate through each word in the line
    for word in line.split():
        if word == "A":
            A = 1 + A
        elif word == "B":
            B = 1 + B
        elif word == "O":
            O = 1 + O
        elif word == "OO":
            OO = 1 + OO
        elif word == "AB":
            AB = 1 + AB

#printing the results
print('There are ' + str(A) + ' patients of blood type A.')
print('There are ' + str(B) + ' patients of blood type B.')
print('There are ' + str(O) + ' patients of blood type O.')
print('There are ' + str(OO) + ' patients of blood type OO.')
print('There are ' + str(AB) + ' patients of blood type AB.')
```

```
There are 15 patients of blood type A.
There are 1 patients of blood type B.
There are 15 patients of blood type O.
There are 0 patients of blood type OO.
There are 13 patients of blood type AB.
```

Question 8

Write a function curconv() that takes as input:

1. a currency represented using a string (e.g., 'JPY' for the Japanese Yen or 'EUR' for the Euro)
2. an amount and then converts and returns the amount in US dollars.

Ans for question 8

```
In [61]: file = open("currencies.txt", "r")
def curconv(currency, amount):
    USD = 0
    for word in file:
        if word.split()[0] == currency:
            USD = amount * float(word.split()[1])
    return USD
```

```
In [62]: curconv('AUD',125)
```

```
Out[62]: 129.3144625
```

```
In [63]: curconv('INR',155)
```

```
Out[63]: 2.7534665
```

```
In [64]: curconv('NOK',245)
```

```
Out[64]: 41.0880435
```

Question 9

Each of the following will cause an exception (an error). Identify what type of exception each will cause. Trying to add incompatible variables, as in adding 6 + 'a' Referring to the 12th item of a list that has only 10 items Using a value that is out of range for a function's input, such as calling math.sqrt(-1.0) Using an undeclared variable, such as print(x) when x has not been defined Trying to open a file that does not exist, such as mistyping the file name or looking in the wrong directory.

Ans to question 9

Trying to add incompatible variables, as in adding 6 + 'a' = TypeError

```
In [71]: 6 + "a"
```

```
-----
TypeError                                Traceback (most recent call last)
Cell In[71], line 1
----> 1 6 + "a"

TypeError: unsupported operand type(s) for +: 'int' and 'str'
```

Referring to the 12th item of a list that has only 10 items = IndexError

```
In [73]: list = [0,1,2,3,4,5,6,7,8,9]
print(list(12))
```

```

-----
TypeError                                Traceback (most recent call last)
Cell In[73], line 2
      1 list = [0,1,2,3,4,5,6,7,8,9]
----> 2 print(list(12))

TypeError: 'list' object is not callable

```

Using a value that is out of range for a function's input, such as calling `math.sqrt(-1.0)` = `NameError`

In [75]: `math.sqrt(-1,0)`

```

-----
NameError                                Traceback (most recent call last)
Cell In[75], line 1
----> 1 math.sqrt(-1,0)

NameError: name 'math' is not defined

```

Using an undeclared variable, such as `print(x)` when `x` has not been defined = `NameError`

In [77]: `print(z)`

```

-----
NameError                                Traceback (most recent call last)
Cell In[77], line 1
----> 1 print(z)

NameError: name 'z' is not defined

```

Trying to open a file that does not exist, such as mistyping the file name or looking in the wrong directory. = `FileNotFoundError`

In [78]: `file= open('currency.txt', 'r')`

```

-----
FileNotFoundError                        Traceback (most recent call last)
Cell In[78], line 1
----> 1 file= open('currency.txt', 'r')

File ~\AppData\Local\Programs\Python\Python311\Lib\site-packages\IPython\core\interactiveshell.py:282, in _modified_open(file, *args, **kwargs)
    275 if file in {0, 1, 2}:
    276     raise ValueError(
    277         f"IPython won't let you open fd={file} by default "
    278         "as it is likely to crash IPython. If you know what you are doing,
    "
    279         "you can use builtins' open."
    280     )
--> 282 return io_open(file, *args, **kwargs)

FileNotFoundError: [Errno 2] No such file or directory: 'currency.txt'

```

Question 10

Encryption is the process of hiding the meaning of a text by substituting letters in the message with other letters, according to some system. If the process is successful, no one but the intended recipient can understand the encrypted message. Cryptanalysis refers to attempts to undo the encryption, even if some details of the encryption are unknown (for example, if an encrypted message has been intercepted). The first step of cryptanalysis is often to build up a table of letter frequencies in the encrypted text. Assume that the string letters is already defined as 'abcdefghijklmnopqrstuvwxyz'. Write a function called frequencies() that takes a string as its only parameter, and returns a list of integers, showing the number of times each character appears in the text. Your function may ignore any characters that are not in letters.

Ans to question 10

```
In [82]: def frequencies(text):
        letters = 'abcdefghijklmnopqrstuvwxyz'
        f = [0] * 26
        # converting the text to lower case to avoid further errors while comparing
        text = text.lower()
        for char in text:
            if char in letters:
                index = letters.index(char)
                f[index] += 1
        return f

        a = "Kashish"
        b = "Data analysts"
        result = frequencies(a)
        result1 = frequencies(b)

        print("Character frequencies:", result)
        print("Character frequencies:", result1)
```

```
Character frequencies: [1, 0, 0, 0, 0, 0, 0, 2, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 2,
0, 0, 0, 0, 0, 0]
Character frequencies: [4, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 2,
2, 0, 0, 0, 0, 1, 0]
```

In []: