Q1. Write a python program to count the frequency of any specific word (in your domain) in the paragraph.

```
In [ ]:  def main():

             name ="Suryansh Bachchan Verma"
             reg = 2347260
             para = f'My name is {name}, I have choose "Event Management" as my domain my

             word = "my"
             frequency = count_word_frequency(para, word)
             print(f"The word '{word}' appears {frequency} times in the paragraph.")


         def count_word_frequency(paragraph, word):
             words_list = paragraph.split()
             count = 0
             for w in words_list:
                 if w.lower().strip('.,') == word.lower().strip('.,'):
                     count += 1
             return count

         main()
```

```
The word 'my' appears 3 times in the paragraph.
```

Q2. Write a python program to display all the datatypes of selected specific elements in the paragraph. (For example:– name - string, reg.no - int, marks - float, etc.)

```
In [ ]:  from re import split

         def get_datatype(element):
             try:
                 int(element)
                 return "int"
             except ValueError:
                 pass

             try:
                 float(element)
                 return "float"
             except ValueError:
                 pass

             return "string"


         def main():
             name ="Suryansh Bachchan Verma"
             reg = 2347260
             year = 2023
             para = f'My name is {name}, I have choose "Event Management" as my domain my

             words = split(" |, ", para)
             print("Data Types of Selected Specific Elements:")
```

```python
        for word in words:
            datatype = get_datatype(word)
            print(f"{word} - {datatype}")

main()
```

```
Data Types of Selected Specific Elements:
My - string
name - string
is - string
Suryansh - string
Bachchan - string
Verma - string
I - string
have - string
choose - string
"Event - string
Management" - string
as - string
my - string
domain - string
my - string
registration - string
number - string
is - string
2347260 - int
from - string
MCA - string
section - string
B - string
2023 - int
```

Q3. Write a python program to count the number of alphabets, numeric and other special symbols in the paragraph.

```python
In [ ]: def main():

            alpha, numeric, special = 0, 0, 0
            name ="Suryansh Bachchan Verma"
            reg = 2347260
            year = 2023
            para = f'My name is {name}, I have choose "Event Management" as my domain my


            for i in para:

                if i.isdigit():
                    numeric += 1
                elif i.isalpha():
                    alpha += 1
                elif not i == " ":
                    special += 1

            print(f"Number of int characters in the paragraph: {numeric}")
            print(f"Number of String characters in the paragraph: {alpha}")
            print(f"Number of special characters in the paragraph: {special}")

main()
```

```
Number of int characters in the paragraph: 11
Number of String characters in the paragraph: 102
Number of special characters in the paragraph: 4
```

Q4. Create a Set with elements that consists of various data types (int, float, string, Boolean, etc. from your domain) and perform the functions pop(), clear(), discard() and del. Write the insights as docstring.

```python
In [ ]:  # event set = {Sl, Name_of_event, budget, invitation_status(T/F)}
         event = {1, "Pitch Deck", 10000.00, True}

         '''
             pop(): Removes random item from the set

         '''
         event.pop()
         print("pop() : ",event)

         # event set = {Sl, Name_of_event, budget, invitation_status(T/F)}
         event = {1, "Pitch Deck", 10000.00, True}
         '''
             clear(): Removes all elements in a set.
         '''
         event.clear()
         print("clear() : ", event)

         # event set = {Sl, Name_of_event, budget, invitation_status(T/F)}
         event = {1, "Pitch Deck", 10000.00, True}
         '''
             discard(): Similar to remove() just that discard() does not raises
             an error if the value entered does not exsist.
         '''
         event.discard("p")
         print("discard(<value>) : ", event)

         # event set = {Sl, Name_of_event, budget, invitation_status(T/F)}
         event = {1, "Pitch Deck", 10000.00, True}
         '''
             del : Deletes the set event, so print(event) will raise an error
         '''
         del event # del will remove the event element, hence an error will be produced
         print("del event : ", end="")
         print(event)
```

```
pop() :  {1, 10000.0}
clear() :  set()
discard(<value>) :  {'Pitch Deck', 1, 10000.0}
del event :
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[4], line 35
     33 del event
     34 print("del event : ", end="")
---> 35 print(event)

NameError: name 'event' is not defined
```

Q5. Update the Set with minimum 5 string attributes of your domain and arrange the Set in descending order.

```
In [ ]: event = set()
        print(type(event))

        event_details = {"5 Pm", "Near Greenwich Park", "Cultural Event", "Students", "F
        event.update(event_details)

        print(event)

        # Arranging in decending order

        sorted_list = sorted(event, reverse=True)

        print(set(sorted_list))
```

```
<class 'set'>
{'Food:Available', '5 Pm', 'Cultural Event', 'Near Greenwich Park', 'Students'}
{'Food:Available', 'Students', '5 Pm', 'Cultural Event', 'Near Greenwich Park'}
```

Q6. Create a Tuple and Execute the packing and unpacking operations of tuples using the attributes of your domain.

```
In [ ]: event = ("5 Pm", "Near Greenwich Park", "Cultural Event", "Students", "Available

        time, location, event_type, group, food = event

        print("Time :", time)
        print("location :", location)
        print("Event Type :", event_type)
        print("Group :", group)
        print("Food :", food)
```

```
Time : 5 Pm
location : Near Greenwich Park
Event Type : Cultural Event
Group : Students
Food : Available
```

Q7. Enter your domain name as characters and count any number of characters and print the count (for example – ('p','r','o','g','r','a','m') count of 'r' = 2)

```
In [ ]: domain = "Event Management" # input("Enter your domain name : ") -- Since input
        domain = domain.replace(" ","")

        letter = "e" # input("Eneter the letter to search : ") -- Since input values are
        count = 0

        for i in domain.lower():
            if letter == i:
                count += 1

        print(f"Domain : {domain}")
```

```
print(f"letter : {letter}")
print("Count = ", count)
```

```
Domain : EventManagement
letter : e
Count =  4
```

Q8. Enter your domain name, execute all the slicing possibilities and also negative indexing.

```python
In [ ]: domain = "Event Management"  # input("Enter your domain name : ") -- Since input
        l = domain.replace(" ","")

        print(l, end="\n\n")
        # Positive indexing and slicing
        print("Positive indexing and slicing:")
        print("Character at index 0:", domain[0])
        print("Characters from index 1 to 3:", domain[1:4])
        print("Characters from index 2 onwards:", domain[2:])

        # Negative indexing and slicing
        print("\nNegative indexing and slicing:")
        print("Character at index -1:", domain[-1])
        print("Characters from index -4 to -2:", domain[-4:-1])
        print("Characters except the last one:", domain[:-1])

        # Using negative step in slicing
        print("\nUsing negative step:")
        print("Reverse the string:", domain[::-1])
        print("Every second character in reverse:", domain[::-2])
```

```
EventManagement

Positive indexing and slicing:
Character at index 0: E
Characters from index 1 to 3: ven
Characters from index 2 onwards: ent Management

Negative indexing and slicing:
Character at index -1: t
Characters from index -4 to -2: men
Characters except the last one: Event Managemen

Using negative step:
Reverse the string: tnemeganaM tnevE
Every second character in reverse: teeaa nv
```