Write a program that takes a sentence as input. Capitalize the first letter of the sentence and print the modified sentence.

In [1]:
```
a=input("Write a text:")
print(a.capitalize())
```

```
Hi hello
```

Write a program that takes a string as input. Convert the entire string to lowercase and print it.

In [6]:
```
string=input("Type something")
string.lower()
```

Out[6]:
```
'hi hello'
```

Write a program that takes a string as input. Convert the entire string to uppercase and print it.

In [8]:
```
string=input("Type something")
string.upper()
```

Out[8]:
```
'HI THERE'
```

Write a program that takes a sentence as input. Convert the sentence to title case (where the first letter of each word is capitalized) and print it.

In [10]:
```
string=input("Type something")
string.title()
```

Out[10]:
```
'A Big Elephant'
```

Write a program that takes a string with leading spaces as input. Remove the leading spaces and print the stripped string.

In [16]:
```
string=input("Type something")
string.lstrip()
```

Out[16]:
```
'hi ther'
```

Write a program that takes a string with trailing spaces as input. Remove the trailing spaces and print the stripped string.

In [18]:
```
string=input("Type something")
string.rstrip()
```

Out[18]:
```
'hi hello'
```

Write a program that takes a string with leading and trailing spaces as input. Remove both leading and trailing spaces and print the stripped string.

In [20]:
```
string=input("Type something")
string.strip()
```

Out[20]:
```
'how are you'
```

Write a program that takes a sentence and two words (old and new) as input. Replace all occurrences of the old word with the new word in the sentence and print the result.

In [22]:
```
string=input("Enter a text:")
old_word=input("Enter a word to replace:")
new_word=input("Enter the new word:")
string.replace(old_word,new_word)
```

Out[22]: 'Biriyani is my favourite.'

Write a program that takes a string and a character as input. Count the number of times the character appears in the string (case-insensitive) using a loop and conditional statements (or directly with count() after converting the string to a single case).

In [28]:
```
string=input("Enter a string")
char=input("Enter a character")
string.count(char)
```

Out[28]: 3

In [32]:
```
string=input("Enter a string")
char=input("Enter a character")
string.lower()
string.lower()
count=0
for ch in string:
    if ch==char:
        count+=1
print(f"{count} number of times {char}")
```

```
3 number of times k
```

Write a program that takes a string and a substring as input. Find the index of the first occurrence of the substring. If the substring is not found, print a message indicating that.

In [42]:
```
string=input("Enter a string:")
string1=input("Enter a substring:")
index=string.index(string1)
if index!= -1:
    print(f"the {string1} is found at index {index}")
else:
    print(f"the{string1} is not found")
```

```
the Arun is found at index 12
```

Write a program that takes a string and a substring as input. Find all occurrences of the substring in the string using a while loop and the find() method. Print the starting index of each occurrence.

In [19]:
```
string=input("Type something")
string1=input("Type the word need to find")
index=string.find(string1)
while index != -1:
    print(f"The substring is found at",index)
    index=string1.find(string1, index + 1)
```

```
The substring is found at 6
```

Write a program that takes a comma-separated string of items as input. Split the string into a list of individual items and print each item.

In [27]:
```
strings=input("Enter a comma separeted string")
text=strings.split(',')
for string in text:
    print(string.strip())
```

```
hi how are you
Arun
```

Write a program that takes a filename as input. Check if the filename ends with ".txt". If it does, print "This is a text file.";

otherwise, print "This is not necessarily a text file."

In [29]:
```
file=input("write the filename")
file.endswith(".txt")
```

Out[29]: True

Write a program that takes a line of text as input. Check if the line starts with "Subject: ". If it does, print "This line is likely an email subject."; otherwise, print "This line does not appear to be an email subject."

In [33]:
```
text=input("write a text")
text1=input("starting word of the sentence")
text.startswith(text1)
```

Out[33]: True

Write a program that takes a string as input and checks if all characters in the string are alphanumeric. Print "Alphanumeric string" or "Not an alphanumeric string."

In [47]:
```
string=input("write something")
if string.isalnum():
    print("Alphanumeric string")
else:
    print("Not an alphanumeric string")
```

```
Alphanumeric string
```

Write a program that takes a string as input and checks if all characters in the string are alphabetic. Print "Alphabetic string" or "Not an alphabetic string."

In [63]:
```
text=input("write something")
if text.isalpha():
    print("Alphabetic string")
else:
    print("Not an alphabetic string.")
```

```
Alphabetic string
```

Write a program that takes a string as input and checks if all characters in the string are decimal digits. Print "Decimal string" or "Not a decimal string."

In [69]:
```
text=input("type decimal numbers")
if text.isdecimal():
    print("Decimal string")
else:
    print("Not a decimal string")
```

```
Decimal string
```

Write a program that takes a string as input and checks if all characters in the string are digits. Print "Digit string" or "Not a digit string."

In [73]:
```
text=input("type digits")
if text.isdigit():
    print("Digit string")
else:
    print("Not a digit string")
```

```
Digit string
```

Write a program that takes a string as input and checks if all characters in the string are numeric characters. Print "Numeric

string" or "Not a numeric string

In [77]:
```
text=input("type numbers")
if text.isnumeric():
    print("Numeric string")
else:
    print("Not a Numeric string")
```

```
Not a Numeric string
```

Write a program that takes a string as input. Check if all cased characters in the string are lowercase. If they are, print "All lowercase"; otherwise, print "Not all lowercase."

In [81]:
```
text=input("type something")
if text.islower():
    print("All lowercase")
else:
    print("Not all lowercase.")
```

```
All lowercase
```

Write a program that takes a string as input. Check if all cased characters in the string are uppercase. If they are, print "All uppercase"; otherwise, print "Not all uppercase."

In [85]:
```
text=input("type something")
if text.isupper():
    print("All uppercase")
else:
    print("Not all uppercase.")
```

```
All uppercase
```

Write a program that takes a string as input. Check if the string is in title case. If it is, print "Title case string"; otherwise, print "Not a title case string."

In [95]:
```
text=input("type something")
if text.istitle():
    print("Title case string")
else:
    print("Not a title case string.")
```

```
Title case string
```

Write a program that takes a string as input. Check if all characters in the string are whitespace characters. If they are, print "Whitespace string"; otherwise, print "Not a whitespace string."

In [101…
```
text=input("type something")
if text.isspace():
    print("Whitespace string")
else:
    print("Not a whitespace string.")
```

```
Whitespace string
```

Write a program that takes a number (as a string) and a desired width as input. Pad the beginning of the number with leading zeros so that it reaches the specified width. Print the zero-filled string.

In [111…
```
text=input("type a number")
text1=int(input("desired width of input"))
padded_text = text.zfill(text1)
```

```
print("Padded text:", padded_text)
print("Length of padded text:", len(padded_text))
```

```
Padded text: 000045
Length of padded text: 6
```

Question: Write a program that takes a string and a width as input. Center the string within the specified width, using '*' as the fill character, and print the result.

In [113…

```
text=input("type a number")
text1=int(input("desired width of input"))
centered_text = text.center(text1, '*')
print("Centered string:", centered_text)
```

```
Centered string: **34*
```

In [ ]: