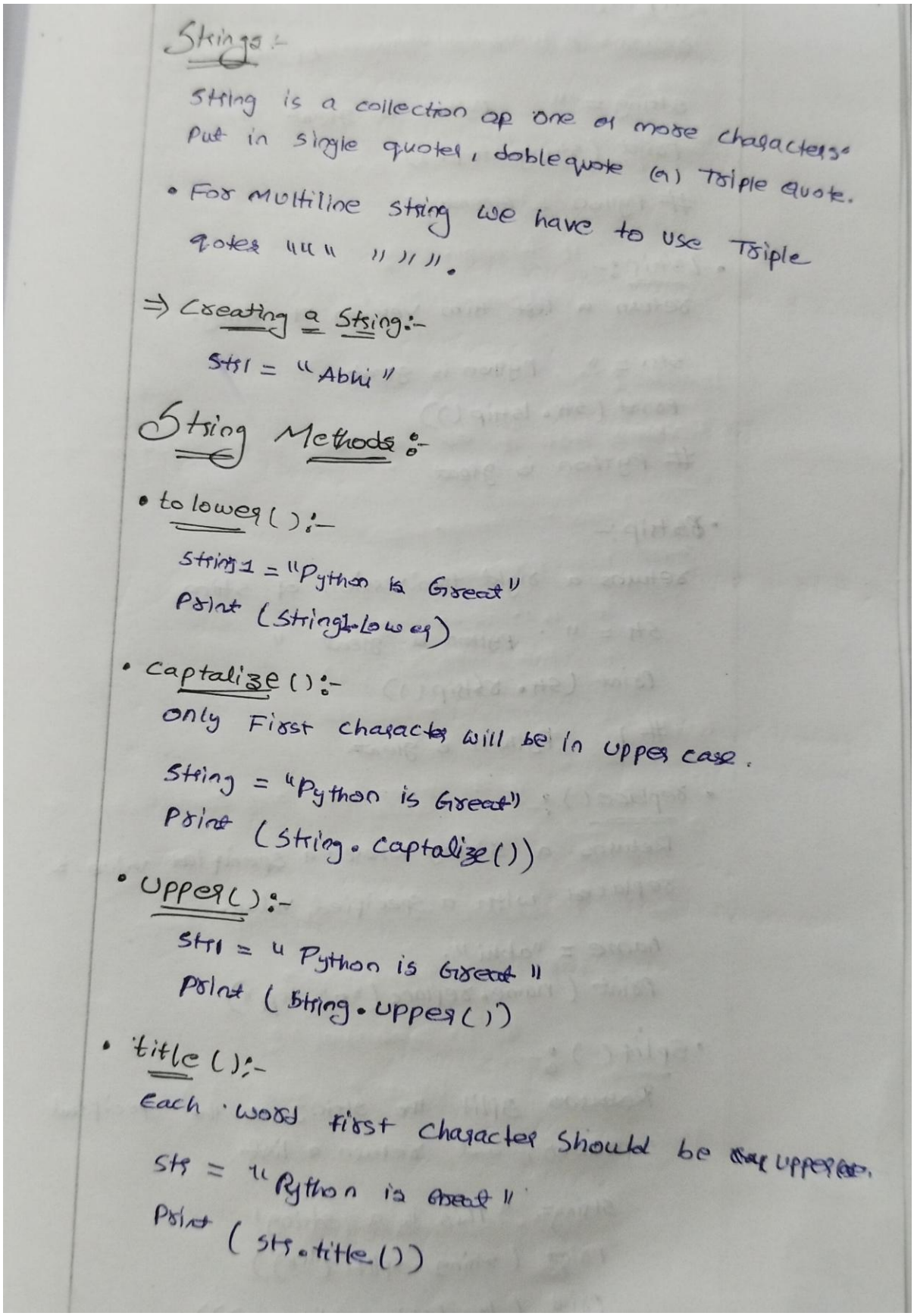
String and string methods:



Strings :-

String is a collection of one or more characters put in single quotes, double quote (or) Triple quote.

- For Multiline string we have to use Triple quotes """ """ """.

⇒ Creating a String :-

```
Str1 = "Abhi"
```

String Methods :-

- to lower() :-

```
string1 = "Python is Great"
Print (string1.lower)
```

- captalize() :-

only First character will be in upper case.

```
String = "python is Great"
Print (String.captalize())
```

- Upper() :-

```
Str1 = " Python is Great "
Print (String.upper())
```

- title() :-

Each word first character should be upper.

```
Str = " Python is Great "
Print (Str.title())
```

- strip():-
returns a trimmed version of the string.

```
string = "    Python is great   "
print (string.strip())
# Python is Great.
```

- lstrip:-
return a left trim version of string.

```
str = "    python is great   "
print (str.lstrip())
# Python is great   "
```

- rstrip:-
returns a right trim version of string

```
str = "   Python is great   "
print (str.rstrip())
#    Python is great.
```

- replace():-
Returns a string where a specified value is replaced with a specified value.

```
name = "abhi";
print ( name.replace ("a", "A"))
```

- Split():-
~~Returns~~ split the string at the specified seperator, and return a list.

```
string= " This is a string"
print ( string .split (" "))
print ( len (string .split(" ")))
```

join():-

Joins the elements of an iterable to the end of the string.

```
String = "This is a string"
Print ("-".join(string.split(" ")))
```

Find():-

Searches the string for a specified value and returns the position of where it was found.

returns the first occurence of the index.

Ex:-

```
String = "this is string"
Print (string.Find("t"))   #0
```

Index():-

Searches the string for a specified value and returns the position of where it was found.

Ex:-

```
string = "String"
Print (string.index("s"))
```

startswith():- Returns true if the string starts with the specified value.

Ex:-

```
txt = "Hello, welcome to my world"
x = txt.startswith("Hello")
print (x)
```

endswith():- returns true if the string ends with the specified value.

```
name = Abhi
Print (name.endswith("i"))  #true.
```

isdigit():-

Returns true if all characters in the string are digit.

Ex:-

```
Password = "secret_123"
for i in Password:
    if i.isdigit():
        Print ("valid", i)
```

The isalpha() method returns True if all the characters are alphabet letters (a-z).

isalpha():-

Returns true if all characters in the string are alpha

Ex:-

```
Password = "secret-123"
for i in password:
    if i.isalpha():
        Print ("valid", i)
```

isalnum():-

Returns true if all characters are combination of alpha and numeric.

Ex:-

```
Password = "12345 alphabets"
for i in password:
    if i.isalnum():
        Print ("valid", i)
```

isnumeric():-

Ex:-

```
txt.isnumeric()
```

isspace():-

return true if all characters in the string are whitespace.

Ex:-

```
txt = "   "
x = txt.isspace()
print(x).
```

swapcase:-

swapcase, lowercase become uppercase and vice versa.

Ex:-

```
name = "Abhi"
print(name.swapcase())
# aBHI
```

Zfill():-

Fills the string with a specified number of 0 values at the beginning.

Ex:-

```
txt = "50"
x = txt.zfill(10)
print(x)
# 0000000050
```

rjust:-

Returns a right justified version of a string.

Ex:-

```
txt = "banana"
x = txt.rjust(20)
print(x)  # banana.
```

ljust:- Returns a left justified version of a string.

Ex:-

```
txt = "banana"
x = txt.ljust(10)
print(x)  # banana
```

center():-

Returns a centered string.

Ex:-

```
txt = "banana"
x = txt.center(10)
print(x)
```

O/P:- banana.

Example:-

```
String = "Python"
res = ""
for i in String:
    if i == i.upper():
        res = res + i.lower()
    else:
        res += i.upper()
Print(res).
```