

```
In [ ]: #strings ae immutable
STR1="HELLO"
STR1[1]="J"
```

```
-----
-----
TypeError                                Traceback (most recent call last)
<ipython-input-3-015f153cd62a> in <module>()
      1 #strings ae immutable
      2 STR1="HELLO"
----> 3 STR1[1]="J"

TypeError: 'str' object does not support item assignment
```

```
In [ ]: name = "Alan Turing"
name[0]
```

```
Out[ ]: 'A'
```

```
In [ ]: name = "Alan Turing"
print(name[0])
print(len(name))
print (name[len(name)-1])
```

```
A
11
9
```

```
In [ ]: print(name[-1])
print(name[-3])
print(name[-5])
```

```
g
i
u
```

```
In [ ]: name="hello"
print(name[0:-2:1])
```

```
hel
```

```
In [ ]: name="hello"
print(name[::1])
```

```
hello
```

```
In [ ]: name="hello"
print(name[::-1])
```

```
olleh
```

```
In [ ]: name="hello"
        print(name[-2:0:-1])
```

lle

```
In [ ]: name="hello"
        print(name[-3:])
```

llo

```
In [ ]: name="hello"
        print(name[-3::-1])
```

leh

```
In [ ]: fileList = ["myfile.txt", "myprogram.exe", "yourfile.txt"]
        for fileName in fileList:
            if ".txt" in fileName:
                print(fileName)
```

myfile.txt  
yourfile.txt

```
In [ ]: data="myprogram.exe"
        print(data[2])
        print(data[-1])
        print(len(data))
        print(data[0:8])
```

p  
e  
13  
myprogra

```
In [1]: #string methods
        data="hello"
        print(data.center(20), "*")
        print(data)
        print(data.endswith("lo"))
        print(data.find("llo"))
```

hello \*

hello  
True  
2

```
In [ ]: data="hello"
        print(data.center (20))
        print(data.center (20, "*"))
        print(data)
```

hello  
\*\*\*\*\*hello\*\*\*\*\*  
hello

```
In [ ]: print(data.count("l"))
        print(data.count("l",0,2))
        print(data.count("l",0,3))
```

```
2
0
1
```

```
In [ ]: #find returns index
        print(data.find("o"))
        print(data.find("o",0,2))
```

```
4
-1
```

```
In [ ]: print(data.isalpha())
        print(data.isdigit())
```

```
True
False
```

```
In [2]: l1=["apple","orange","grapes"]
        print("".join(l1))
        print(l1)
        print("and".join(["a","b","c"]))
```

```
appleorangegrapes
['apple', 'orange', 'grapes']
aandbandc
```

```
In [ ]: #for join the list should be strings
        l1=[1,2,3]
        print("-".join(l1))
```

```
-----
----
TypeError                                Traceback (most recent call 1
ast)
<ipython-input-21-61ec243e10f1> in <module>()
      1 l1=[1,2,3]
----> 2 print("-".join(l1))

TypeError: sequence item 0: expected str instance, int found
```

```
In [ ]: str1="helloHOW"
        print(str1.lower())
```

```
hellohow
```

```
In [ ]: str1="helloHOWel"  
print(str1.replace("el", "me"))  
print(str1)
```

```
hmeloHOWme  
helloHOWel
```

```
In [ ]: str1="helloHOWelel"  
print(str1.replace("el", "me", 2))
```

```
hmeloHOWmeel
```

```
In [ ]: str1="    hello    "  
print(str1.split())
```

```
['hello']
```

```
In [ ]: str1="*****hello*** "  
print(str1.split("l"))
```

```
['*****he', '', 'o*** ']
```

```
In [ ]: str1="helloHOW"  
print(str1.startswith("hello"))
```

```
True
```

```
In [ ]: str1="    hello    "  
print(str1.lstrip())
```

```
hello
```

```
In [3]: str1="    thello    "  
print(str1.rstrip())
```

```
thello
```

```
In [ ]: str1="    thello    "  
print(str1.strip())
```

```
thello
```

```
In [ ]: str1="thello    "  
print(str1.strip("th"))
```

```
ello
```

```
In [ ]: str1="thello    "  
print(str1.strip("el"))
```

```
thello
```

```
In [ ]: str1="hello how are you "  
print(str1.split(" "))  
  
['hello', '', 'how', 'are', 'you', '', '']
```

```
In [ ]: str1="hello ,how ,are, you "  
print(str1.split())  
  
['hello', ',how', ',are,', 'you']
```

```
In [ ]: #string is pallindrome  
str1=input("enter the string")  
rev=str1[-1::-1]  
  
if (str1==rev):  
    print("pallindrome")  
else:  
    print("not")  
  
enter the stringhello  
not
```

```
In [ ]: str1="hello how are you "  
print(str1.split(" ",2))  
  
['hello', 'how', 'are you ']
```

```
In [ ]: str1="hello.txt"  
print(str1.split("."))  
  
['hello', 'txt']
```

```
In [ ]: str1=".hello.txt."  
print(str1.split("."))  
  
['', 'hello', 'txt', '']
```

```
In [ ]: str1=".hello.txt."  
print(str1.strip("."))  
  
hello.txt
```

```
In [ ]: str1="mon tues wednes"  
str2=str1.split()  
print (str2)  
print (" ".join(str2))  
  
['mon', 'tues', 'wednes']  
mon tues wednes
```

```
In [ ]: #Remove all vowel characters from a string( university question)
vowels="AEIOUaeiou"
s=input("Enter the string...")
ns=""
for char in s:
    if char not in vowels:
        ns=ns+char
print("new string after removing vowels=",ns)
```

Enter the string...hello how are you  
new string after removing vowels= hll hw r y

```
In [ ]: #Remove characters at odd index positions from a string ( university q
uestion)
s=input("Enter the string...")
i=0
ns=""
while i<len(s):
    if i%2==0:
        ns=ns+s[i]
    i=i+1
print("New string:",ns)
```

Enter the string...good morning  
New string: go onn

```
In [ ]: #Palindrome checking using loop
s=input("Enter the string..")
beg=0
end=len(s)-1
while beg<end:
    if s[beg]!=s[end]:
        print("Not palindrome")
        break
    beg+=1
    end-=1
else:
    print("Palindrome")
```

Enter the string..malayalam  
Palindrome

```
In [ ]: #Replace all the spaces in the input string with * or if no spaces fou
nd,put $ at the start and end of the string.(university question)
s=input("Enter the string:")
s=s.replace(" ", "*")
if "*" not in s:
    s="$"+s+"$ "
    print(s)
else:
    print(s)
```

Enter the string:hello  
\$hello\$

```
In [ ]: #Slice the string to two separate strings; one with all the characters
in the odd indices and one with all characters in even indices.(univer
sity question)
s=input("enter the string:")
eps=s[0:len(s):2]
print("slice with even position characters:",eps)
ops=s[1:len(s):2]
print("slice with odd position chracters:",ops)
```

enter the string:python programming language  
slice with even position characters: pto rgamn agae  
slice with odd position chracters: yhnpormigl nug

```
In [ ]: #Remove all occurrence of a substring from a string
s=input("enter the string..")
ss=input("enter substring to remove..")
ls=len(s) # length of the string
lss=len(ss) # length of the substring
ns="" # new string
i=0
while i<ls:
    css=s[i:lss+i] #css is the substring to be compared extracted from m
ain string
    if css==ss:
        i=i+lss
    else:
        ns=ns+s[i]
        i=i+1
print("new string",ns)
```

enter the string..hello how are you  
enter substring to remove..are  
new string hello how you

```
In [ ]: #Program to replace all occurrence of a substring with a new substring
         (university question)
s=input("enter string..")
ss=input("enter substring to remove..")
nss=input("enter the substring to replace....")
ls=len(s)
lss=len(ss)
ns=""
i=0
while i<ls:
    css=s[i:lss+i]
    if css==ss:
        ns=ns+nss
        i=i+lss
    else:
        ns=ns+s[i]
        i=i+1
print("new string",ns)
```

```
enter string..hello python language
enter substring to remove..hello
enter the substring to replace....hai
new string hai python language
```