Shubhangi_Dhikale_35

1.Write a python program to check that a string contains only a certain set of characters (in this case a-z, A-Z, and 0-9).

In [5]:
```
import re
text=input("enter the string:")
pattern="[A-Z a-z 0-9]"
check=re.findall(pattern,text)
if check:
    print("string contains the alphanumeric characters")
else:
    print("string does not contain the alphanumeric characters")
```

```
enter the string:shubhANGI2345
string contains the alphanumeric characters
```

In [14]:
```
def checkalnum(string):
    import re
    pattern="^[A-Z a-z 0-9_]*$"
    check=re.search(pattern,text)
    if check:
        print("string contains the alphanumeric characters")
    else:
        print("string does not contain the alphanumeric characters")
string="Shubh Dhikale123"
checkalnum(string)
```

```
string contains the alphanumeric characters
```

2.Create a program to split a string by only the first occurrence of any substring

In [35]:
```
import re
string="data science and machine learning python"
result=re.split("[\W]",string)
result
```

Out[35]:
```
['data', 'science', 'and', 'machine', 'learning', 'python']
```

3.Code that would match a string that has an a followed by zero or more b's.

In [40]:
```
import re
string="aabbb bbbaa bbbbbbb baa a"
result=re.findall("b*b",string)
result
```

Out[40]: ['bbb', 'bbb', 'bbbbbbb', 'b']

In [44]:
```
import re
string="a abb aabbbabb bbbb baab"
result=re.findall("ab*",string)
result
```

Out[44]: ['a', 'abb', 'a', 'abbb', 'abb', 'a', 'ab']

4.Wap to find Three-digit numbers followed by space followed by two-digit numbers in a string.

In [54]:
```
import re
numbers="12 234 12 3456 123 56 123 67 123"
result=re.findall(r"\b\d{2}[ ]\d{3}\b",numbers)
result
```

Out[54]: ['12 234', '56 123', '67 123']

In [60]:
```
import re
numbers="12 234 12 3456 123 56 123 67 123 543 67"
result=re.findall("\w{2}[ ]\w{3}",numbers)
result
```

Out[60]: ['12 234', '12 345', '56 123', '67 123']

In [62]:
```
import re
numbers="12 234 12 3456 123 56 123 67 123 543 67"
result=re.findall("[0-9]{2}[ ][0-9]{3}",numbers)
result
```

Out[62]: ['12 234', '12 345', '56 123', '67 123']

5.Write a Python program that matches a string that has an a followed by one or more b's

In [76]:
```
import re
string="a bb aabbbabb bbbb baab bbb ab  b"
pattern="bb+"
result=re.findall(pattern,string)
result
```

Out[76]: ['bb', 'bbb', 'bb', 'bbbb', 'bbb']

In [77]:
```
import re
string="a bb aabbbabb bbbb baab bbb ab  b"
pattern="b+"
result=re.findall(pattern,string)
result
```

Out[77]:
```
['bb', 'bbb', 'bb', 'bbbb', 'b', 'b', 'bbb', 'b', 'b']
```

In [78]:
```
import re
string="a bb aabbbabb bbbb baab bbb ab  b"
pattern="ab+"
result=re.findall(pattern,string)
result
```

Out[78]:
```
['abbb', 'abb', 'ab', 'ab']
```

In [82]:
```
string=input("enter the string:")
pattern="b+"
result=re.search(pattern,string)
if result:
    print("string contain one and more b's")
else:
    print("string does not contain one or more b's")
```

```
enter the string:bbb
string contain one and more b's
```

In [90]:
```
def check(string):
    string=input("enter the string:")
    pattern="b+"
    result=re.search(pattern,string)
    if result:
        print("string contain one and more b's")
    else:
        print("string does not contain one or more b's")

check(string)
```

```
enter the string:abbbbbb
string contain one and more b's
```

6.Write a program to search inform in a string both in uppercase & in lowercase.

(INFORM or inform)

In [92]:
```
import re
string="inform"
pattern="[a-z]{6}"
result=re.search(pattern,string)
result.group()
```

Out[92]: 'inform'

In [96]:
```
import re
string="INFORM"
pattern="\w+"
result=re.search(pattern,string)
result.group()
```

Out[96]: 'INFORM'

In [101]:
```
import re
string="INFORM"
pattern="[A-Z]+"
result=re.search(pattern,string)
result.group()
```

Out[101]: 'INFORM'

7.WAP that matches a string that has an a followed by zero or one 'b'

In [126]:
```
import re
string="abc bbb abbc bbbb b ccab ab ac baba"
result=re.findall("ab?",string)
result
```

Out[126]: ['ab', 'ab', 'ab', 'ab', 'a', 'ab', 'a']

In [145]:
```
def check(string):
    import re
    pattern="ab?"
    result=re.search(pattern,string)

    if result:
        print("found")
    else:
        print("not found" )
string="abc baba bb abab aabb ccabab a"
check(string)
```

found

In [147]:
```
def check(string):
#     import re
    pattern="ab?"
    result=re.match(pattern,string)

    if result:
        print("found")
    else:
        print("not found" )
string="c baba bb abab aabb ccabab a"
check(string)
```

```
not found
```

8.WAP that matches a string that has an a followed by three 'b'. .

In [167]:
```
import re
text="abbbb bbbb abbba abbbabbb cabbb cabab aa b"
result=re.findall("[a]{1}[b]{3}",text)
result
```

Out[167]:
```
['abbb', 'abbb', 'abbb', 'abbb', 'abbb']
```

In [178]:
```
import re
text="abbb abbb bbbb abbba abbbabbbb cabbb cabab aa b"
pattern=r"\b[a]{1}[b]{3}\b"
result=re.findall(pattern,text)
result
```

Out[178]:
```
['abbb', 'abbb']
```

In [179]:
```
import re
text="abbb abbb bbbb abbba abbbabbbb cabbb cabab aa b"
pattern=r"\b[a]{1}[b]{3}"
result=re.findall(pattern,text)
result
```

Out[179]:
```
['abbb', 'abbb', 'abbb', 'abbb']
```

In [197]:
```
def check(string):
    pattern="abbb"
    result=re.search(pattern,string)
    if result:
        return "found"
    else:
        return "not found"
print(check("abbb"))
print(check("baab"))
print(check("abbbb"))
```

```
found
not found
found
```

9.Write a Python program that matches a string that has an a followed by two to three 'b'.

In [198]:
```
import re
text="abba abbba bbba aabb abba bbba babbbba"
result=re.findall('[b]{2,3}',text)
result
```

Out[198]:
```
['bb', 'bbb', 'bbb', 'bb', 'bb', 'bbb', 'bbb']
```

In [199]:
```
import re
text="abba abbba bbba aabb abba bbba babbbba"
result=re.findall('[a]{1}[b]{2,3}',text)
result
```

Out[199]:
```
['abb', 'abbb', 'abb', 'abb', 'abbb']
```

10.Find all the words starting in range of k-n using a for loop & re.

In [216]:
```
string="mansi kajal dhikale nilisha aditya akash vijay shubhangi"
pattern=r"\b[k-n]{1}\w+\b"
result=re.findall(pattern,string)
for i in result:
    print(i,end=" ")
```

```
mansi kajal nilisha
```

In [219]:
```
string="mansi kajal dhikale nilisha aditya akash vijay shubhangi"
pattern=r"\b[k-n][a-z]+\b"
result=re.findall(pattern,string)
for i in result:
    print(i,end=" ")
```

```
mansi kajal nilisha
```

11.Write a Python program to find sequences of lowercase letters joined with an underscore.

In [225]:
```
string="python_class and data_science classes,Machine Learning"
pattern="[\w]+[_][\w]+"
result=re.findall(pattern,string)
(result)
```

Out[225]:
```
['python_class', 'data_science']
```

In [226]:
```
string="python_class and data_science classes,Machine Learning"
pattern="[a-z]+[_][a-z]+"
result=re.findall(pattern,string)
(result)
```

Out[226]:
```
['python_class', 'data_science']
```

12.Write a program to find the sequences of one upper case letter followed by lower case letters.

In [230]:
```
string="To search Inform in a String BOTH in Uppercase  in lowercase"
pattern="[A-Z][a-z]+"
result=re.findall(pattern,string)
result
```

Out[230]:
```
['To', 'Inform', 'String', 'Uppercase']
```

In [234]:
```
string="To search Inform in a String BOTH in Uppercase  in lowercase"
pattern=r"\b[A-Z][a-z]+\b"
result=re.findall(pattern,string)
result
```

Out[234]:
```
['To', 'Inform', 'String', 'Uppercase']
```

13.Write a program that matches a string that has an 'a' followed by anything,ending in 'b'.

In [244]:
```
import re
string="absorb adhira ashab atlab adverb sahir ab"
pattern=r"\b[a][a-z]+[b]\b"
result=re.findall(pattern,string)
result
```

Out[244]:
```
['absorb', 'ashab', 'atlab', 'adverb']
```

14..WAP that matches a word at the beginning of a string.

In [266]:
```
def word(string,substring):
    pattern="^[a-z]+"                       # or  ^\w+
    result=re.findall(pattern,string)
    if result:
        print(f"string starts with {substring}")
    else:
        print(f"string not starts with {substring}")
string="we are learning data science and python"
substring="we"
word(string,substring)
```

```
string starts with we
```

15.Write a program that matches a word at the end of a string, with optional punctuation.

In [284]:
```
def check(string):
    pattern="[a-zA-Z]+.?$"
    result=re.search(pattern,string)
    if result:
        print("string with punctuation mark")
    else:
        print("string with no punctuation mark")
string="my name is shubhangi dhikale."
check(string)
```

```
string with punctuation mark
```

16.Write a Python program that matches a word containing 'z'.

In [323]:
```
string="zudio amazon program Buzz"
pattern="\w*[z]\w+"
result=re.findall(pattern,string)
result
```

Out[323]:
```
['zudio', 'amazon', 'Buzz']
```

17.Write a program that matches a word containing 'z', not at the start or end of the word.

In [327]:
```
string="zudio amazon freez,breezz enzymes zip quartz bronze"
pattern="[a-y]+[z][a-y]+"
result=re.findall(pattern,string)
result
```

Out[327]:
```
['amazon', 'enzymes', 'bronze']
```

18.Write a program to match a string that contains only upper and lowercase letters, numbers, and underscores.

In [330]:
```
def check(string):
    pattern="[A-Z a-z 0-9]"
    result=re.search(pattern,string)
    if result:
        return "found"
    else:
        return "not found"
print(check("Shubhangi_Dhikale1999"))
print(check("*$#"))
```

```
found
not found
```

In [331]:
```
def check(string):
    pattern="\w"
    result=re.search(pattern,string)
    if result:
        return "found"
    else:
        return "not found"
print(check("Shubhangi_Dhikale1999"))
print(check("*$#"))
```

```
found
not found
```

19.WAP where a string will start with a specific number.

In [332]:
```
def check(string):
    pattern="[95]{2}\d{8}"
    result=re.search(pattern,string)
    if result:
        return "Number is start with specific number"
    else:
        return "Number is not start with specific number"
print(check("9567419351"))
print(check("9145673490"))
```

```
Number is start with specific number
Number is not start with specific number
```

20. Write a Python program to remove leading zeros from an IP address.

In [335]:
```
import re
string="001.011.002.012"
pattern="0"
result=re.sub(pattern,"",string)
result
```

Out[335]:
```
'1.11.2.12'
```

In [341]:
```
import re
string="001.011.002.012"
pattern="^[0]."
result=re.sub(pattern,"",string)
result
```

Out[341]:
```
'1.011.002.012'
```

21.Write a program to check for a number at the end of a string. .

In [5]:
```
import re
def check(string):
    pattern="\d$"
    number=re.search(pattern,string)
    if number:
        print("number is at the end of the string")
    else:
        print("number is not end of the string")
string="shubhangi1990"
check(string)
```

```
number is at the end of the string
```

22.Code a program to search the numbers (0-9) of length between 1 to 3 in given string. "Exercises number 1, 12, 13, and 345 are important"

In [6]:
```
import re
string="Exercises number 1, 12, 13, and 345 are important"
pattern="[0-9]{1,3}"
result=re.findall(pattern,string)
result
```

Out[6]:
```
['1', '12', '13', '345']
```

In [7]:
```
import re
string="Exercises number 1, 12, 13, and 345 are important"
pattern="\d{1,3}"
result=re.findall(pattern,string)
result
```

Out[7]:
```
['1', '12', '13', '345']
```

23.Write a program to search some literals strings in a string. .

Sample text : 'The quick brown fox jumps over the lazy dog.' Searched words : 'fox', 'dog', 'horse'

In [10]:
```
import re
text ="The quick brown fox jumps over the lazy dog"
pattern="fox|dog|horse"
match=re.search(pattern,text)
if match:
    print("found")
else:
    print("Not found")
```

```
found
```

24.Write a Python program to search a literals string in a string and also find

the location within the original string where the pattern occurs. . Sample text : 'The quick brown fox jumps over the lazy dog.' Searched words : 'fox'

In [15]:
```
import re
string= 'The quick brown fox jumps over the lazy dog.'
pattern="fox"
result=re.search(pattern,string)
print(result.group())
print(result.span())
```

```
fox
(16, 19)
```

25.Write a code to find a common substrings within a string.

Sample text : 'Python exercises, PHP exercises, C# exercises' Pattern : 'exercises' Note: There are two instances of exercises in the input string.

In [18]:
```
import re
string="Python exercises, PHP exercises, C# exercises"
pattern="exercises"
result=re.findall(pattern,string)
result
```

Out[18]:
```
['exercises', 'exercises', 'exercises']
```

In [19]:
```
import re
string="Python exercises, PHP exercises, C# exercises"
pattern="e........"
result=re.findall(pattern,string)
result
```

Out[19]:
```
['exercises', 'exercises', 'exercises']
```

26.Write a program to find the occurrence and position of the substrings within a string.

In [21]:
```
string="Python exercises, PHP exercises, C# exercises"
pattern="exercises"
result=re.findall(pattern,string)
print(result)
count=0
for i in result:
    count+=1
print(count)
print(string.index("exercises"))
```

```
['exercises', 'exercises', 'exercises']
3
7
```

27.Write a code to replace whitespaces with an underscore and vice versa.

In [23]:
```
string="Write a_program_to search_some literals strings_in a string"
string1=re.sub("_","&",string)
string2=re.sub("\s","_",string1)
string3=re.sub("&"," ",string2)
print(string3)
```

```
Write_a program to_search some_literals_strings in_a_string
```

28.How would you remove all whitespaces from a string?

In [25]:
```
string="my name is shubhangi dhikale"
result=re.sub("\s","",string)
result
```

Out[25]:
```
'mynameisshubhangidhikale'
```

29.Write a Python program to match if two words from a list of words starting with the letter 'P'

In [28]:
```
def check(string):
    pattern="P.+P.+"
    result=re.search(pattern,string)
    if result:
        print("match is found")
    else:
        print("match is not found")
string="Python Program"
check(string)
```

```
match is found
```

30.Write a code to find all words starting with 'a' or 'e' in a given string.

.

In [30]:
```
string="all evening end other given string around"
pattern=r'\b[ae]{1}\w+\b'
result=re.findall(pattern,string)
result
```

Out[30]:
```
['all', 'evening', 'end', 'around']
```

31.Write a Python program to separate and print the numbers and their position of a given string.

In [36]:
```
string="""my mail id is shubhangik.dhikale1999@gmail.com
        mobile number is:9130419351
        account no:904607690"""
pattern="\d+"
result=re.finditer(pattern,string)
for number in result:
    print("number:",number.group(),"start:",number.start(),"end:",number.end())
```

```
number: 1999 start: 32 end: 36
number: 9130419351 start: 72 end: 82
number: 904607690 start: 102 end: 111
```

32.Write a Python program to replace all occurrences of space, comma, ordot with a colon.

In [44]:
```
string="shubhangi, dhikale,90."
pattern="[ ,.]"
result=re.sub(pattern,":",string)
result
```

Out[44]:
```
'shubhangi::dhikale:90:'
```

33.Write a program to replace maximum 2 occurrences of space, comma, or dot with a colon.

In [45]:
```
string="shubhangi, dhikale,90."
pattern="[ ,.]"
result=re.sub(pattern,":",string,count=2)
result
```

Out[45]:
```
'shubhangi::dhikale,90.'
```

34.Write a Python program to find all three, four, five characters long words in a string.

In [50]:
```
string="Write a Python program to find all three, four, five characters long word
pattern=r"\b\w{3,5}\b"
result=re.findall(pattern,string)
result
```

Out[50]:
```
['Write', 'find', 'all', 'three', 'four', 'five', 'long', 'words']
```

35.Write a program to extract values between quotation marks of a string.

In [51]:
```
string="my name is shubhangi dhikale"
pattern="^.+"
result=re.findall(pattern,string)
result
```

Out[51]:
```
['my name is shubhangi dhikale']
```

36.How would you remove multiple spaces in a string?

In [55]:
```
string="How  would you   remove multiple spaces    in a string?"
pattern="\s+"
result=re.sub(pattern, " ",string)
result
```

Out[55]:
```
'How would you remove multiple spaces in a string?'
```