

BCA – 502: Python Programming

Rahul Kumar Singh

In today's Class we have discussed on Mobile number, Password, email, url validation using regular expression in Python.

Mobile number validation in Python using regular expression.

Conditions to validate Mobile number:-

- The mobile number must be '10' digits long.
- The mobile number can have '11' digits if including '0' at the starting.
- The mobile number can be of '12' digits if including '91' at the starting.
- First digit should contain number between [6-9].
- The rest 9 digit can contain any number between [0-9].

Example:-

```
import re
number=input("Enter your mobile number: ")
pattern="(0|91)?[-\s]?[6-9][0-9]{9}"
result=re.findall(pattern, number)
if (result):
```

```
print(number,"is valid mobile number.")
```

else:

```
print(number,"is not valid mobile number.")
```

Output:-

Enter your mobile number: 9560283230

9560283230 is valid mobile number.

Password Validation in Python Using Regular Expression:-

Conditions for password validation:-

- At least 8 characters and Maximum 20 Characters
- At least one uppercase letters: A-Z
- At least one lowercase letters: a-z
- At least one numbers: 0-9
- At least one special characters: @?#\$%^&+=

Example:-

```
import re
```

```
pattern=("^.*(?={8,20})(?=.*\d)(?=.*[a-z])(?=.*[A-Z])(?=.*[@?#$%^&+=]).*$")
```

```
password= input("Enter Password: ")
```

```
result = re.findall(pattern, password)
```

```
if (result):  
    print(password,"is valid Password")  
else:  
    print(password,"is not valid Password")
```

Output:-

```
Enter Password: Rohan@123  
Rohan@123 is valid password
```

Url validation in Python using regular expression:-

Conditions for URL validation:-

The URL must start with either http or https and then followed by :// and then it must contain www. and then followed by subdomain of length (2, 256) and last part contains top level domain like .com, .org etc.

Example:-

```
import re  
  
url="https://www.Intcollege.ac.in/"  
  
regex="((http|https)://)(www.)?"+"[a-zA-Z0-9@:~#?&//=]" + "{2,256}\\.[a-z]" + "{2,6}\\b([-a-zA-Z0-9@:~#?&//=]*")  
  
def isValidURL(str):
```

```
if(re.search(regex,str)):
    print ("Url is valid")
else:
    print("Url is not valid")
isValidURL(url)
```

Output:-

Url is valid

Email validation using regular expressions in python:-

Conditions for Email validation:-

An email is a string (a subset of ASCII characters) separated into two parts by @ symbol, a “personal_info” and a domain, that is personal_info@domain.

Example:-

```
import re
email="Intcollege@gmail.com"
pattern="^[a-z0-9]+[\.\_]?[a-z0-9]+[@]\w+[.]\w{2,3}$"
def check(email):
    if(re.search(pattern,email)):
        print(email,"is valid Email")
```

else:

print(email," is invalid Email")

check(email)

Output:-

Intcollege@gmail.com is a valid Email