

## MDS-173 PYTHON FOR DATA SCIENCE

### Department of Data Science & Statistics

### CHRIST (Deemed to be University), Bengaluru

**Name: Prateek Kumar**

**Class: 1MDS**

**REG.NO: 2248013**

### Checking the Htpps of Web-Pages on Internet

```
# Importing the In-built Regex library
import re

# Re.compile() method is used to compile a regular expression pattern provided as a string into a regex pattern object
pattern=re.compile("^(\http:\\\\www\\.|https:\\\\www\\.|http:\\\\|https:\\\\)?[a-z0-9]+([\-\.\.]{1}[a-z0-9]+)*\\. [a-z]{2,5}(:[0-9]{1,5})?(\\

# Call the Match object method to return a string of the actual matched text.
print(pattern.match("https://christuniversity.in"))

<re.Match object; span=(0, 27), match='https://christuniversity.in'>
```

### Describing the Regex Function available in Regex Module using Phone Number Validation

```
import re # the built-in regex library
my_phone_number = input("Enter Your Number") # # Create User Defined Input Call function.
numbers = [] # Create List of Numbers

# re.compile() method is used to compile a regular expression pattern provided as a string into a regex pattern object
pattern = re.compile("^(\\+\\d{1,3}[- ]?)?\\d{10}$")

# Flags allow you to control the regular expression engine and flags argument seems to be an integer argument with the default value
# to control the default regex behavior.

# Match() method finds match if it occurs at start of the string.
def match(pattern, string, flags=0):
    return _compile(pattern, flags).match(string)

# Re.fullmatch() returns a match object if and only if the entire string matches the pattern.
def fullmatch(pattern, string, flags=0):
    return _compile(pattern, flags).fullmatch(string)

# Search() returns a Match object if there is a match anywhere in the string
def search(pattern, string, flags=0):
    return _compile(pattern, flags).search(string)

# Sub() replaces one or many matches with a string
def sub(pattern, repl, string, count=0, flags=0):
    return _compile(pattern, flags).sub(repl, string, count)

# Subn() specifies strings or a set of strings or patterns that match it.
def subn(pattern, repl, string, count=0, flags=0):
    return _compile(pattern, flags).subn(repl, string, count)

# Spilt() returns a list where the string has been split at each match
def split(pattern, string, maxsplit=0, flags=0):
    return _compile(pattern, flags).split(string, maxsplit)

# Findall() returns a list containing all matches
def findall(pattern, string, flags=0):
    return _compile(pattern, flags).findall(string)

# finditer() works exactly the same as the re. findall() method except it returns an iterator yielding match objects matching the
# regex pattern in a string instead of a list.
def finditer(pattern, string, flags=0):
    return _compile(pattern, flags).finditer(string)

# Compile() method is used to compile a regular expression pattern provided as a string into a regex pattern object
def compile(pattern, flags=0):
    return _compile(pattern, flags)

# Purge() are in tests (specifically in the test_re unittests for the re module.
def purge():
```

```
_cache.clear()
_compile_repl.cache_clear()

# Template() helps field the details in the wikitext of any page on the wiki. Normally searches ignore non-alphanumeric characters,
# but regular expressions (regex) accept all characters, plus metacharacters.
def template(pattern, flags=0):
    return _compile(pattern, flags|T)

_special_chars_map = {i: '\\' + chr(i) for i in b'()[\]{}?*+-|^$\\.&~# \t\n\r\v\f'}

# Escape() to match a character having special meaning in regex and use a escape sequence prefix with a backslash
def escape(pattern):

    if isinstance(pattern, str):
        return pattern.translate(_special_chars_map)
    else:
        pattern = str(pattern, '')
        return pattern.translate(_special_chars_map).encode('')

# Print all the Functions.
print(findall)
print(escape)
print(split)
print(compile)
print(template)
print(escape)
print(purge)
print(finditer)
print(sub)
print(subn)
print(search)
print(fullmatch)
print(match)
```

```
Enter Your Number55993554596
<function findall at 0x7fb86f516320>
<function escape at 0x7fb86f5165f0>
<function split at 0x7fb86f516200>
<function compile at 0x7fb86f516560>
<function template at 0x7fb86f5167a0>
<function escape at 0x7fb86f5165f0>
<function purge at 0x7fb86f516680>
<function finditer at 0x7fb86f516440>
<function sub at 0x7fb872f6cf80>
<function subn at 0x7fb86f5160e0>
<function search at 0x7fb872f6cd40>
<function fullmatch at 0x7fb872f6cc20>
<function match at 0x7fb872f6cb90>
```