Regular expressions (regex or regexp) in Python are a powerful tool for pattern matching within strings. They allow you to search, manipulate, and validate strings based on defined patterns. The re module in Python provides support for regular expressions.

Basic Functions of re Module:

re.search(pattern, string):

Searches for the first occurrence of a pattern within a string.

python

Copy code

import re

text = "Hello, this is a sample text"

result = re.search(r"sample", text)

print(result.group()) # Output: 'sample'

re.match(pattern, string):

Matches the pattern only at the beginning of the string.

python

Copy code

text = "Hello, this is a sample text"

result = re.match(r"Hello", text)

print(result.group()) # Output: 'Hello'

re.findall(pattern, string):

Finds all occurrences of a pattern within a string and returns them as a list.

python

Copy code

text = "apple, banana, cherry, apple"

result = re.findall(r"apple", text)

print(result) # Output: ['apple', 'apple']

re.sub(pattern, replacement, string):

Substitutes occurrences of a pattern in a string with another specified string.

python

Copy code

text = "Hello, World!"

result = re.sub(r"World", "Universe", text)

print(result) # Output: 'Hello, Universe!'

Common Patterns:

\d: Matches any digit (equivalent to [0-9]).

\w: Matches any alphanumeric character (equivalent to [a-zA-Z0-9\_]).

\s: Matches any whitespace character.

.: Matches any character except a newline.

^: Matches the start of a string.

$: Matches the end of a string.

Regular Expression Flags:

re.IGNORECASE or re.I: Case-insensitive matching.

re.MULTILINE or re.M: Multiline matching, affecting ^ and $.

Compiled Patterns:

re.compile(pattern):

Compiles a regular expression pattern into a regex object for reusing.

python

Copy code

pattern = re.compile(r"apple")

result = pattern.findall("apple, banana, cherry, apple")

print(result) # Output: ['apple', 'apple']