**Regular Expression Practice Questions**

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

import re

def contains\_only\_allowed\_characters(input\_string):

# Define the regular expression pattern

pattern = r'^[a-zA-Z0-9]+$'

# Use re.match() to check if the input\_string matches the pattern

if re.match(pattern, input\_string):

return True

else:

return False

# Test the function with some examples

test\_strings = ["Hello123", "GoodMorning!", "12345", "abcXYZ", "123\_abc"]

for test\_string in test\_strings:

if contains\_only\_allowed\_characters(test\_string):

print(f"'{test\_string}' contains only allowed characters.")

else:

print(f"'{test\_string}' contains characters not allowed.")

**'Hello123' contains only allowed characters.**

**'GoodMorning!' contains characters not allowed.**

**'12345' contains only allowed characters.**

**'abcXYZ' contains only allowed characters.**

**'123\_abc' contains characters not allowed.**

Question 2- Write a RegEx pattern that matches a string that has an a followed by zero or more b's

import re

# Sample text

text = "ab abb abbb ac abc"

# Define the regular expression pattern

pattern = r'ab\*'

# Use re.findall() to find all matching substrings

matches = re.findall(pattern, text)

# Print the list of matched substrings

print(matches)

**['ab', 'abb', 'abbb', 'a']**

Question 3- Write a RegEx pattern that matches a string that has an a followed by one or more b's

import re

pattern = r'ab+'

# Test the pattern with some example strings

test\_strings = ["a", "ab", "abb", "abbb", "ac", "abc", "bb", "bbb"]

for test\_string in test\_strings:

if re.match(pattern, test\_string):

print(f"'{test\_string}' matches the pattern.")

else:

print(f"'{test\_string}' does not match the pattern.")

**'a' does not match the pattern.**

**'ab' matches the pattern.**

**'abb' matches the pattern.**

**'abbb' matches the pattern.**

**'ac' does not match the pattern.**

**'abc' matches the pattern.**

**'bb' does not match the pattern.**

**'bbb' does not match the pattern.**

Question 4- Write a RegEx pattern that matches a string that has an a followed by zero or one 'b'.

import re

pattern = r'ab?'

# Test the pattern with some example strings

test\_strings = ["a", "ab", "abb", "abbb", "ac", "abc", "bb", "bbb"]

for test\_string in test\_strings:

if re.match(pattern, test\_string):

print(f"'{test\_string}' matches the pattern.")

else:

print(f"'{test\_string}' does not match the pattern.")

**'a' matches the pattern.**

**'ab' matches the pattern.**

**'abb' matches the pattern.**

**'abbb' matches the pattern.**

**'ac' matches the pattern.**

**'abc' matches the pattern.**

**'bb' does not match the pattern.**

**'bbb' does not match the pattern.**

Question 5- Write a RegEx pattern in python program that matches a string that has an a followed by three 'b'.

import re

pattern = r'abbb'

# Test the pattern with some example strings

test\_strings = ["abbb", "a", "abb", "abbbb", "abc"]

for test\_string in test\_strings:

if re.match(pattern, test\_string):

print(f"'{test\_string}' matches the pattern.")

else:

print(f"'{test\_string}' does not match the pattern.")

**'abbb' matches the pattern.**

**'a' does not match the pattern.**

**'abb' does not match the pattern.**

**'abbbb' matches the pattern.**

**'abc' does not match the pattern**

Question 6- Write a RegEx pattern in python program that matches a string that has an a followed by two to three 'b'.

import re

pattern = r'ab{2,3}'

# Test the pattern with some example strings

test\_strings = ["ab", "abb", "abbb", "a", "abbbb", "abc"]

for test\_string in test\_strings:

if re.match(pattern, test\_string):

print(f"'{test\_string}' matches the pattern.")

else:

print(f"'{test\_string}' does not match the pattern.")

**'ab' does not match the pattern.**

**'abb' matches the pattern.**

**'abbb' matches the pattern.**

**'a' does not match the pattern.**

**'abbbb' matches the pattern.**

**'abc' does not match the pattern**

Question 7- Write a Python program that matches a string that has an 'a' followed by anything, ending in 'b'.

import re

def match\_pattern(input\_string):

pattern = r'a.\*b$'

return re.match(pattern, input\_string)

# Test the program with some example strings

test\_strings = ["acb", "abcdb", "a12345b", "ab", "a", "abbb", "xyzabcw"]

for test\_string in test\_strings:

if match\_pattern(test\_string):

print(f"'{test\_string}' matches the pattern.")

else:

print(f"'{test\_string}' does not match the pattern.")

**'acb' matches the pattern.**

**'abcdb' matches the pattern.**

**'a12345b' matches the pattern.**

**'ab' matches the pattern.**

**'a' does not match the pattern.**

**'abbb' matches the pattern.**

**'xyzabcw' does not match the pattern.**

Question 8- Write a RegEx pattern in python program that matches a word at the beginning of a string.

import re

def match\_pattern(input\_string):

pattern = r'^\w+'

return re.match(pattern, input\_string)

# Test the program with some example strings

test\_strings = ["Hello, world!", "123abc", " leading spaces", "word at the start", "123 456"]

for test\_string in test\_strings:

match = match\_pattern(test\_string)

if match:

matched\_word = match.group()

print(f"Matched word: '{matched\_word}' in '{test\_string}'")

else:

print(f"No match found in '{test\_string}'")

**Matched word: 'Hello' in 'Hello, world!'**

**Matched word: '123abc' in '123abc'**

**No match found in ' leading spaces'**

**Matched word: 'word' in 'word at the start'**

**Matched word: '123' in '123 456'**

Question 9- Write a RegEx pattern in python program that matches a word at the end of a string.

import re

def match\_pattern(input\_string):

pattern = r'\w+$'

return re.search(pattern, input\_string)

# Test the program with some example strings

test\_strings = ["Hello, world!", "123abc", "trailing spaces ", "end word", "123 456"]

for test\_string in test\_strings:

match = match\_pattern(test\_string)

if match:

matched\_word = match.group()

print(f"Matched word: '{matched\_word}' in '{test\_string}'")

else:

print(f"No match found in '{test\_string}'")

**Matched word: 'world' in 'Hello, world!'**

**Matched word: '123abc' in '123abc'**

**No match found in 'trailing spaces '**

**Matched word: 'word' in 'end word'**

**Matched word: '456' in '123 456'**

Question 10- Write a RegEx pattern in python program to find all words that are 4 digits long in a string.

Sample text- '01 0132 231875 1458 301 2725.'

Expected output- ['0132', '1458', '2725']

import re

text = '01 0132 231875 1458 301 2725.'

# Define the regular expression pattern to match 4-digit words

pattern = r'\b\d{4}\b'

# Use re.findall() to find all matching words

matches = re.findall(pattern, text)

# Print the list of matched words

print(matches)

**['0132', '1458', '2725']**