**TRABAJO PRÁCTICO N° 1:**

**IMPLEMENTACIÓN CON PYTHON PARA VALIDACIÓN DE DATOS MEDIANTE EXPRESIONES REGULARES**

**Alumno:** Lucas Damián Soria Gava

**Legajo:** 58156

**PARTE A**

**Ejercicio 1:**

alfabeto = {a, b, c, d, 0, 1, 2, 3, 4}

Cadenas:

x = a1

y = d04

Longitud:

|x| = 2

|y| = 3

Concatenacion:

xy = a1d04

Potencia:

**Ejercicio 2:**

A = {lucas}

B = {20}

A∪B = {lucas, 20}

A∩B = { }

A.B = {lucas20}

A3 = {lucaslucaslucas}

B2 = {2020}

B0 = {ε}

A\* = {ε, lucas, lucaslucas, lucaslucaslucas, lucaslucaslucaslucas, ...}

A(A∪B)\* = {lucas, lucaslucas, lucas20, lucaslucaslucas, lucaslucas20, lucas2020, lucas20lucas, ...}

**Ejercicio 3:**

N = {1, 2, 3}

5 cadenas más cortas de N\* = {ε, 1, 2, 3, 11}

**PARTE B**

**Ejercicio 1:**

1. (ba|b)\*(a|bb)\*

Todas las cadenas que empiezan con cero o más “ba” ó cero o más “b” y terminan con cero o más “a” ó cero o más “bb”

1. (0|1|.. .|9|A|B|C|D|E|F)(w|W)

Todas las cadenas que empiezan con un dígito hexadecimal, seguido de una “w” o una “W”

1. 3. (A|B|...|Z)(a|b|...z)\*

Todas las cadenas de caracteres que empiecen por una letra mayúscula seguidas de 0 o más letras minúsculas

**Ejercicio 2:**

1. Seguridad de una contraseña

^(?=.\*[a-z])(?=.\*[A-Z])((?=.\*\d)|(?=.\*\W))[A-Za-z\d\W]{8,}$

1. Cuenta de twitter

^@[a-zA-Z0-9\_]{1,15}$

1. URL

^https?:\/\/[\w\-\.]+(\.[\w\-]+)+[/#?]?.\*$

1. Email

^[^@]{1,64}@([^@\_])+(\.[^@\_])\*$

1. ID de un vídeo de Youtube

(^https?:\/\/www\.youtube\.com\/watch\?v=)(?P<id>[a-zA-Z0-9\_-]{11,})(&?.\*$)

1. Fecha con formato dd/mm/yyyy o dd-mm-yyyy

^((3[01]|[12][0-9]|0[1-9])/(0[1-9]|1[12])/\d{4})|(3[01]|[12][0-9]|0[1-9])-(0[1-9]|1[12])-\d{4}$

1. Número de teléfono móvil de Argentina, que incluya código de país, de provincia, y el 15

^\+549(11|((2|3)(\d){2,3}))15(\d{7})$

1. Código postal

^[A-Z][\d]{4}[A-Z]{3}$

1. Número de teléfono fijo, que incluya código de país, de provincia

^\+54(11|((2|3)(\d){2,3}))4(\d{6})$

**Ejercicio 3:**

1. Seguridad de una contraseña

import re

def match():

regex = re.compile(r"^(?=.\*[a-z])(?=.\*[A-Z])((?=.\*\d)|(?=.\*\W))[A-Za-z\d\W]{8,}$")

exp = input("Ingrese una contraseña a validar: ")

if regex.match(exp):

print(f'"{exp}": es una contraseña válida')

else:

print(f'"{exp}": NO es una contraseña válida')

if \_\_name\_\_ == "\_\_main\_\_":

match()

1. Cuenta de twitter

import re

def match():

regex = re.compile(r"^@[a-zA-Z0-9\_]{1,15}$")

exp = input("Ingrese una cuenta de Twitter a validar: ")

if regex.match(exp):

print(f'"{exp}": es una cuenta válida')

else:

print(f'"{exp}": NO es una cuenta válida')

if \_\_name\_\_ == "\_\_main\_\_":

match()

1. URL

import re

def match():

regex = re.compile(r"^https?:\/\/[\w\-\.]+(\.[\w\-]+)+[/#?]?.\*$")

exp = input("Ingrese una url a validar: ")

if regex.match(exp):

print(f'"{exp}": es una url válida')

else:

print(f'"{exp}": NO es una url válida')

if \_\_name\_\_ == "\_\_main\_\_":

match()

1. Email

import re

def match():

regex = re.compile(r"^[^@]{1,64}@([^@\_])+(\.[^@\_])\*$")

exp = input("Ingrese un email a validar: ")

if regex.match(exp):

print(f'"{exp}": es un email válido')

else:

print(f'"{exp}": NO es un email válido')

if \_\_name\_\_ == "\_\_main\_\_":

match()

1. ID de un vídeo de Youtube

import re

def search():

regex = re.compile(r"(^https?:\/\/www\.youtube\.com\/watch\?v=)(?P<id>[a-zA-Z0-9\_-]{11,})(&?.\*$)")

exp = input("Ingrese la url de un video a validar: ")

s = regex.search(exp)

if s:

print(f'"{s.group("id")}": es la id del video')

else:

print(f'"{exp}": NO es una url de video válida')

if \_\_name\_\_ == "\_\_main\_\_":

search()

1. Fecha con formato dd/mm/yyyy o dd-mm-yyyy

import re

def match():

regex = re.compile(r"^((3[01]|[12][0-9]|0[1-9])/(0[1-9]|1[12])/\d{4})|(3[01]|[12][0-9]|0[1-9])-(0[1-9]|1[12])-\d{4}$")

exp = input("Ingrese una fecha a validar: ")

if regex.match(exp):

print(f'"{exp}": es una fecha válida')

else:

print(f'"{exp}": NO es una fecha válida')

if \_\_name\_\_ == "\_\_main\_\_":

match()

1. Número de teléfono móvil de Argentina, que incluya código de país, de provincia, y el 15

import re

def match():

regex = re.compile(r"^\+549(11|((2|3)(\d){2,3}))15(\d{7})$")

exp = input("Ingrese un teléfono móvil de Argentina a validar: ")

if regex.match(exp):

print(f'"{exp}": es un teléfono móvil de Argentina válido')

else:

print(f'"{exp}": NO es un teléfono móvil de Argentina válido')

if \_\_name\_\_ == "\_\_main\_\_":

match()

1. Código postal

import re

def match():

regex = re.compile(r"^[A-Z][\d]{4}[A-Z]{3}$")

exp = input("Ingrese un codigo postal a validar: ")

if regex.match(exp):

print(f'"{exp}": es un código postal válido')

else:

print(f'"{exp}": NO es un código postal válido')

if \_\_name\_\_ == "\_\_main\_\_":

match()

1. Número de teléfono fijo, que incluya código de país, de provincia

import re

def match():

regex = re.compile(r"^\+54(11|((2|3)(\d){2,3}))4(\d{6})$")

exp = input("Ingrese un teléfono fijo de Argentina a validar: ")

if regex.match(exp):

print(f'"{exp}": es un teléfono fijo de Argentina válido')

else:

print(f'"{exp}": NO es un teléfono fijo de Argentina válido')

if \_\_name\_\_ == "\_\_main\_\_":

match()