**LABORATORIO 19. MÓDULO RE**

* Define la función **es\_valida\_fecha**, que recibe un string con una fecha y regrese **True** si la fecha es válida, de lo contrario regresa **False**. Utiliza la función **match**. Define la función **main**, pide una fecha y manda llamar a la función **es\_valida\_fecha**.

           Ejemplos:

* fecha = **01-09-2024** regresaría **True**
* fecha = **2024-13-04** regresaría **False**
* fecha = **14-09-24** regresaría **False**

**(0?[1-9]|[12][0-9]|3[01])\-(0?[1-9]|1[012])\-((19|20)\d\d)**

**A screenshot of a computer

Description automatically generated**

# Validar fecha

import re

def es\_valida\_fecha(fecha):

  patron = "(0?[1-9]|[12][0-9]|3[01])\-(0?[1-9]|1[012])\-((19|20)\d\d)"

  resultado = re.match(patron, fecha)

  if resultado != None:

    return True

  else:

    return False

def main():

  fecha = input("Dame una fecha: ")

  res = es\_valida\_fecha(fecha)

  print("La fecha es válida: ", res)

main()

* + Define la función **es\_valido\_url**, que recibe un string con un url y regrese True si el url es válido, de lo contrario regresa False. Utiliza la función **match**. Define la función **main**, pide un url y manda llamar a la función **es\_valido\_url**.

Ejemplos:

* url = **https://www.pedro.lopez.com.mx** regresaría **True**
* url = **http://juansanchez.net** regresaría **True**
* url = **http:/www.pedrop.com** regresaría **False**
* url = **www.pedro.lopez.com.mx** regresaría **False**
* url = **http://www.pedro.lopez.com** regresaría **True**

# Validar url

import re

def es\_valido\_url(url):

  patron = "^(https?:\/\/)(www\.)?[a-zA-Z0-9.-\_]+\.(com|net|com.mx)+$"

  lista = re.findall(patron, url)

  print(lista)

  resultado = re.match(patron, url)

  if resultado != None:

    return True

  else:

    return False

def main():

  url = input("Dame un url: ")

  res = es\_valido\_url(url)

  print("El url es válido: ", res)

main()

**ACTIVIDAD 19. MÓDULO RE**

Utiliza el archivo de texto [**act19\_E.txt**](file:///C:\Users\L00614578\Documents\Github\Analitica\CalendarioAgo2024\actividades\act19_E.txt) y define un programa que encuentre todos los **teléfonos** y los guarde en una lista. Imprime la lista. Utiliza la función **findall**.

# Encontrar todos los telefonoss

import re

texto = """

Laura Sánchez

442-153-32-42

laura.sanch@hotmail-mx.com

http://www.laurasan.com

Pedro LÓPEZ

448 342 33 12

pedro-lopez54@gmail.com

https://www.pedro.lopez.com.mx

Violeta Pérez

123.214.43.72

violeta\_45@outlook.com

www.violeta.net

Juan Sánchez

juan\_sanchez@gmail.com.mx

345 234-34-12

www.juan\_sanchez.net"""

patron = "\d{3}[ -.]\d{3}[ -.]\d{2}[ -.]\d{2}"

print(re.findall(patron, texto, flags = re.M))

* + Define la función **filtra**, que recibe un string con un conjunto de palabras y regresa una lista con las palabras que contengan ñ (eñes). Utiliza la función **findall**. Define la función **main**, pide un string con palabras y mande llamar a la función f**iltra**. Imprime la lista.

           Ejemplo: **filtra** (**"NIÑO, Acción, Perro, Gato, Expresión, Español"**) regresaría **["NIÑO", "Español"]**

# Filtra palabras con ñÑ

import re

def filtra(palabras):

  patron = "[a-zA-Z]+[Ññ][a-zA-Z]+"

  resultado = re.findall(patron, palabras)

  return resultado

def main():

  palabras = input("Dame las palabras: ")

  res = filtra(palabras)

  print(res)

main()

**Valida correo**

* + Define la función **es\_valido\_mail**, que recibe un string con una dirección de correo electrónico y regrese **True** si la dirección IP es válida, de lo contrario regresa **False**. Utiliza la función **match**. Define la función **main**, pide un correo y manda llamar a la función **es\_valido\_mail**.

                      Ejemplos:

* correo = **pedro-lopez54@gmail.com** regresaría **True**
* correo = **juansanchez.gmail**regresaría **False**
* correo = **pedro@com** regresaría **False**
* correo = **juan\_sanchez@gmail.com.mx**regresaría **True**
* correo = **laura.sanch@hotmail-mx.com**regresaría **True**

**A screenshot of a computer

Description automatically generated**

**^[a-zA-Z0-9-.\_]+@[a-zA-Z0-9-.\_]+\.(com|com\.mx)$**

pedro-lopez54@gmail.com

juansanchez.gmail

pedro@com

Laura Sánchez

442-153-32-42

laura.sanch@hotmail-mx.com

http://www.laurasan.com

Pedro LÓPEZ

448 342 33 12

pedro-lopez54@gmail.com

https://www.pedro.lopez.com.mx

Violeta Pérez

123.214.43.72

violeta\_45@outlook.com

www.violeta.net

Juan Sánchez

juan\_sanchez@gmail.com.mx

345 234-34-12

www.juan\_sanchez.net

* + Define la función **reemplaza\_acentos** que recibe un texto y regresa el texto remplazando todas las vocales acentuadas por asteriscos. Utiliza la función **sub**. Define la función **main**, pide un texto y manda llamar a la función **reemplaza\_acentos** e imprime el nuevo texto.

                      Ejemplo: **"Hola Salomón, Andrés e Adán, buen día"** regresaría **"Hola Salom\*n, Andr\*s y Ad\*n, buen d\*a"**.

# Reemplaza acentos

import re

def reemplaza\_acentos(texto):

  resultado = re.sub("[ÁÉÍÓÚáéíóú]", "\*", texto)

  return resultado

def main():

  texto = input("Dame el texto: ")

  res = reemplaza\_acentos(texto)

  print(res)

main()

* + Define la función **reemplaza\_espacios\_slashes** que recibe un teléfono y sustituye los **espacios** y **slashes** por **puntos**. Define la función **main**, pide un texto y manda llamar a la función **reemplaza\_espacios\_slashes** e imprime el nuevo texto. Utiliza la función **sub**.

                    Ejemplos:

* **"442 130/12/80"** regresaría **"442.130.12.80"**
* **"442/130/12/80"** regresaría **"442.130.12.80"**
* Define la