

Tu calificación: 90 %

Tu calificación más reciente: 90 % • Tu calificación más alta: 90 % • Para aprobar necesitas al menos un 80 %. Guardamos tu puntaje más alto.

Próximo artículo 🔿

(i) Estás viendo una versión traducida automáticamente de esta evaluación

Puedes volver a ver el contenido en su idioma original si lo prefieres. No perderás el progreso que hayas conseguido si cambias el idioma. Mostrar la versión en Inglés

Desestimar X

Trabajar con expresiones regulares

Este cuestionario calificado evalúa su comprensión de los conceptos y procedimientos cubiertos en el módulo y el laboratorio que acaba de completar. Responda a las preguntas basándose en lo aprendido en el módulo y en las actividades realizadas en el laboratorio.

Nota:

- Puede consultar el laboratorio que ha completado como ayuda para el cuestionario.
- Para completar este cuestionario, debe haber completado el laboratorio anterior.

```
¿Qué es una expresión regular?
```

Una secuencia de caracteres que forma un patrón de búsqueda

- Un tipo de estructura de los datos en Python
- Método de cifrado de datos
- Una función en Python para el manejo de excepciones

Correcto

2. ¿Qué características tiene un Archivo CSV?

Los Archivos CSV sólo pueden contener datos numéricos.

1/1 punto

1/1 punto

- No se puede leer con un editor de texto.
- Cada línea representa una columna de datos diferente.
- Los datos de cada fila están separados por un carácter especial.

✓ Correcto

para los sitios web seguros.

encriptados que empiezan por http://. La lista incluye los sitios web que terminan en .com y.co. Complete la función secure website domain () para que devuelva la parte del sitio web comprendida entre www. y la última parte de la URL (.com o .co) sólo

3. Tienes una lista de URL de sitios web que incluye tanto los sitios web encriptados de forma segura que empiezan por https://www.como los sitios web no

1/1 punto

def secure_website_domain(website): 1 pattern = r'https://www\.(.*?)\.(com|co)' # enter the regex pattern here

```
result = re.search(pattern, website) # enter the re method here
          if result is None:
           return ""
    5
          return result.group(1) # enter the correct capturing group
    6
    8
         print(secure_website_domain("http://www.text.com")) #Should return nothing
    9
         print(secure_website_domain("https://www.text.com")) #Should return text
   10
         print(secure_website_domain("http://www.text.co")) #Should return nothing
   11
                                                                                                                                     Ejecutar
         print(secure_website_domain("https://www.text.co")) #Should return text
                                                                                                                                  Restablecer
(v) Correcto
```

¡Correcto! ¡Buen trabajo!

def parse_sentences(sentence): pattern = $r' b w + b \lceil ^w \rceil$ #enter the regex pattern here 2

Está explorando los signos de puntuación al final de las frases y desea dividir las frases de modo que cada palabra esté separada y cualquier signo de puntuación se

0 / 1 punto

```
result = re.findall(pattern, sentence) #enter the re method here
    3
          return result
    5
         print(parse_sentences("Hello! How are you doing?")) # should return ['Hello!', 'How', 'are', 'you', 'doing?']
         print(parse_sentences("what a beautiful day it is")) # should return ['what', 'a', 'beautiful', 'day', 'it', 'is']
         print(parse_sentences("2 + 2 is definitely 4!")) # should return ['2', '+', '2', 'is', 'definitely', '4!']
                                                                                                                                       Ejecutar
                                                                                                                                    Restablecer
(X) Incorrecto
   No del todo Revisa el vídeo sobre dividir y sustituir.
```

identificación de los empleados, en el formato: A-1234567 0 A-12345678

5. Una empresa utiliza códigos únicos de 9 caracteres que comienzan con una letra mayúscula, seguida de un guión (-), seguidos de 7 u 8 dígitos como números de

1 / 1 punto

Los informes de proyecto presentados a la empresa incluyen el número de identificación del empleado y un resumen del trabajo que ha realizado en el proyecto. Un

números de ID de empleado de los informes. def find_eid(report): 1 pattern = r'\b[A-C]-\d{7,8}\b' # Expresión regular para encontrar el EID (letra mayúscula seguida de números) 2

analista de datos quiere extraer todos los números de identificación de los empleados de estos proyectos. Complete la función find eid() para extraer estos

```
result = re.findall(pattern, report) # Método para encontrar todas las coincidencias
           return result
    4
    5
         print(find_eid("Employees B-1234567 and C-12345678 worked with products X-123456 and Z-123456789")) # Should return ['B-12345678']
    7
        print(find_eid("Employees B-1234567 and C-12345678, not employees b-1234567 and c-12345678")) #Should return ['B-1234567', 'C-12349
✓ Correcto
   ¡Correcto! ¡Buen trabajo!
```

def replace_domain(address, old_domain, new_domain): old_domain_pattern = r'' + old_domain + '\$'

6. En el fragmento de código proporcionado, ¿cuál es el propósito de la función replace domain ()?

incluya en la palabra contigua. Complete la función parse sentences () para realizar esta tarea.

1/1 punto

```
address = re.sub(old domain pattern, new domain, address)
       return address
Para validar el formato de una dirección de correo electrónico
```

Para crear una nueva dirección de correo electrónico sustituyendo un dominio antiguo por otro nuevo

Para eliminar cualquier dominio de la dirección de correo electrónico Para extraer la parte del nombre de usuario de una dirección de correo electrónico

✓ Correcto

Para almacenar direcciones de correo electrónico con el dominio antiguo que coincidan con el patrón regex

¿Cuál es el propósito de inicializar el old domain email list en el código del laboratorio?

1/1 punto

- Para realizar una operación de sustitución de direcciones de correo electrónico Para almacenar todas las direcciones de correo electrónico de user email list
- Para almacenar direcciones de correo electrónico con el nuevo dominio

✓ Correcto

antiguos por otros nuevos en un Archivo CSV utilizando Python, tal como se describe en el laboratorio? Para definir cabeceras para el archivo de salida que ayuden a organizar la información cuando se abra en una hoja de cálculo

¿Cuál es el objetivo principal de escribir la dirección user data list actualizada en un archivo de salida en el proceso de sustitución de nombres de dominio

1/1 punto

- Para cerrar el Archivo CSV después de leerlo y asegurarse de que no se pueden realizar más cambios en él Para guardar los cambios realizados en las direcciones de correo electrónico y crear un nuevo Archivo CSV con los datos actualizados
- Para almacenar una copia de seguridad de los datos originales del usuario **⊘** Correcto

Desarrollar aplicaciones de realidad virtual. Para automatizar tareas repetitivas y análisis de datos.

¿Cuál es la principal ventaja de utilizar Python para crear informes y emplear expresiones regulares?

1/1 punto

1/1 punto

- Mejorar las medidas de ciberseguridad.
- Para optimizar el rendimiento del sitio web.
- ✓ Correcto

10. En el script de Python para procesar user_emails.csv, ¿cuál es el propósito de la función contains_domain()? Para comprobar si una dirección de correo electrónico pertenece a un dominio concreto, utilice expresiones regulares (regex)

- Para contar el número de direcciones de correo electrónico del Archivo CSV
- Para añadir un nuevo dominio a cada dirección de correo electrónico del Archivo CSV Para encriptar las direcciones de correo electrónico en el Archivo CSV por seguridad
- ✓ Correcto