

TALLER 1 – Lenguajes de Programación 2

Valor 12.5%

El siguiente taller plantea tres ejercicios para ser resueltos en Python, la idea de estos ejercicios es emplear lo visto hasta ahora en nuestro curso. El taller puede ser solucionado en grupos con un máximo de 3 personas, es decir que pueden haber grupos de 3, de 2 o de 1, ustedes deciden como quieren trabajar. Todo conforme a la planilla que ustedes llenaron:

https://docs.google.com/spreadsheets/d/1-uoVgy7kCAOUe7dUxgbEq6ev4b5zMibD3Zu8_QWX-R0/edit?usp=sharing

Es importante que todos los scripts posean **comentarios y docstrings** para entender la lógica del código y que respeten [la formulación del PEP8](#).

NO se permite la importación de NINGUNA BIBLIOTECA a excepción de la biblioteca random.

Los códigos serán sustentados el **próximo martes 13 de septiembre** de forma aleatoria entre los miembros de cada grupo en un pequeño espacio de tiempo. Y hará parte de la calificación final del taller. La selección aleatoria se hace por medio de un script de Python que selecciona que grupo va sustentando.

El taller debe ser enviado como máximo el martes 13 de septiembre a las 10 am, si lo tienen antes sería mucho mejor. El archivo debe ser cargado en un Formulario de Google Forms que enviaré para ustedes y después de las 10 am ya no se aceptará ningún envío.

Guardar el Script en una Carpeta que se llame igual al grupo al que usted pertenece y comprima la carpeta para que la pueda anexar al Google Form.



Empresa de Comunicaciones TELEPYTHON

La empresa de telefonía Telepython está necesitando un generador de teléfonos aleatorios el cual es asignado a un usuario cuando este realiza el registro en su plataforma.

Formulario:

El primer paso es generar un formulario que almacene la información del usuario:

- Nombres
- Apellidos
- Edad
- Correo Electrónico
- Dirección
- Cédula

El formulario debe validar cada uno de los datos y mostrar algún mensaje de error en caso de que el usuario ingrese algún dato no permitido en cada uno de los campos.

Generador de números:

(Importe la biblioteca random)

Una vez los datos del usuario han sido ingresados, se procede a la generación de varios números telefónicos aleatorios de los cuales **SOLO** el último teléfono generado es el que será asignado para el usuario. Para eso el sistema debe mostrar las siguientes opciones que contienen las siguientes sub-opciones dentro de la

cuales el usuario tiene que seleccionar (Puede usar lo visto en la generación de menús para que el usuario escoja la opción que desea):

- Código del País:
 - No
 - Aleatorio
 - Ingresado
- Número de dígitos (mínimo 6 – máximo 10)
- ¿Con separador guion (-) ? (Booleano)
 - En caso de ser True, mostrar dos opciones de separación:
 - XXX-XXXX
 - XXX-XX-XX
- Cantidad de números aleatorios generados (mínimo 2 - máximo 10)

Se debe mostrar en el terminal todos los teléfonos generados.

Ejemplo 1:

- Código del País:
 - No
- Número de dígitos: 7
- ¿Con separador guion (-) ? False
- Cantidad de números aleatorios generados: 3

Resultado

- 0887122
- 9449302
- 3127701

Ejemplo 2:

- Código del País:
 - Ingresado: 57
- Número de dígitos: 9
- ¿Con separador guion (-) ? True
 - En caso de ser True, mostrar dos opciones:
 - XXX-XX-XX
- Cantidad de números aleatorios generados: 5

Resultado

(57)102-31-2487 (57)480-48-4340 (57)520-73-0971 (57)244-28-2321 (57)580-11-0950

Compresión del número generado:

Una vez seleccionado el último número telefónico del punto anterior, dicho teléfono (sin guiones) debe pasar por un algoritmo de compresión el cual está explicado a continuación:

Considere un algoritmo de compresión de números simple que funciona de la siguiente manera:

1. 111155522500 -> '14_53_22_51_02'

El algoritmo recorre el número de izquierda a derecha y retorna un objeto del tipo str. Cada dígito encontrado se almacena junto con el número de veces que ese dígito se repite hasta que se encuentra otro dígito diferente en el número. Cada uno de estos pares está separado por el carácter '_'.

Implemente una función llamada `compresion()` que comprima el número como se describe arriba.

Ejemplos:

1. [IN]: `compresion(100000)`
2. [OUT]: '11_05'

1. [IN]: `compresion(9993330)`
2. [OUT]: '93_33_01'

1. [IN]: `compresion(6540000)`
2. [OUT]: '61_51_41_04'

Puede implementar la función en un script a parte para verificar que efectivamente está funcionando como se espera entregando los mismos resultados mostrados en el ejemplo.

Una vez funcione correctamente, aplique la compresión al telefono del usuario y almacene el resultado en una variable.

Entregando el resultado:

Finalmente, el Script debe generar un e-mail con el siguiente formato el cual debe ser mostrado en el terminal:

Sr(a) {apellido},

Sus datos han sido ingresados correctamente en Telepython, los cuales se resumen a continuación:

Nombre: {nombre} {apellido}

Edad: {edad}

e-mail: {email}

Cédula {cedula}

Teléfono: {teléfono}

Compresión: {compresión}

Gracias por su preferencia.