



Versión: 1.3.0  
Fecha: 26/09/2021

# Informes de Python 3

Por: Fernando "WildRaven"

## Random miscellaneous

### **Función del programa:**

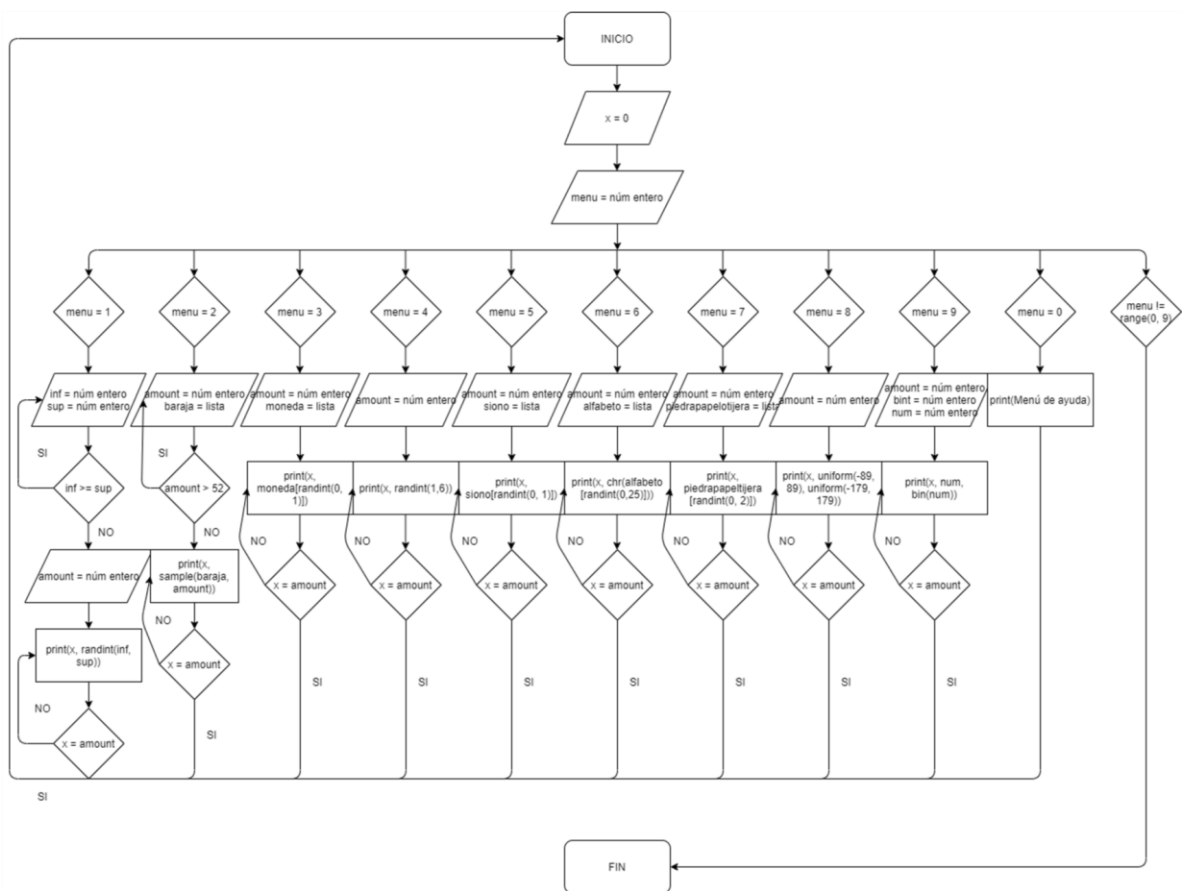
Random miscellaneous puede generar un gran catálogo de variables aleatorias en base a los parámetros que el usuario brinde. Pueden ser desde un número aleatorio en un rango establecido por el usuario, una serie de cartas de la baraja de poquer aleatorio, lanzamiento de monedas aleatorio, lanzamiento de dados aleatorio, decisiones entre sí o no aleatorias, letras aleatorias, piedra papel o tijera aleatorio, coordenadas en grados decimales aleatorios, números binarios aleatorios y un menú de ayuda. En todos los catálogos (a excepción de piedra, papel o tijera y el menú de ayuda) el usuario puede especificar cuantos resultados desea obtener.

### **Descripción de variables:**

- $x = 0$ , este número sirve para enumerar los resultados.
- menú = La decisión del usuario para escoger un catálogo dependiendo del número que entregue.
- amount = Es el número de resultados que el usuario desea obtener para cada catálogo, a excepción de piedra, papel o tijera.

- inf y sup (en el catálogo número aleatorio entre dos puntos): Indican los parámetros de la función randint().
- Listas: Son listas con elementos principalmente de valor string que usan los catálogos de baraja aleatoria, lanzamiento de moneda, si o no y piedra, papel o tijera. Los elementos de las listas son variados dependiendo del catálogo.
- bint: Define la base binaria para establecer un límite superior para el catálogo de número binario aleatorio.

## Diagrama de flujo (versión 1.0.0):



## Documentación:

Nombre	Random miscellaneous (Código)
Autor	Fernando "WildRaven"
Versión	1.3.0
Fecha creación	13/05/2020
Fecha última modif.	26/09/2021
Descripción	Uso del módulo random de Python 3 para darle una función más real respecto a varias cosas mezclaneas de la vida cotidiana, por ejemplo, sacar aleatoriamente cartas de una baraja o conseguir coordenadas aleatorias. En cada línea de código, hay un comentario explicando la función de dicha línea.
Funciones	<b>Módulo de random:</b> randint() sample() uniform()  <b>Creadas:</b> NumAleatorio() BarajaAleatoria() LanzamientoMoneda() LanzamientoDado() DesicionAleatoria() LetraAleatoria() PiedraPapelTijera() CoordenadaGeografica() NumBinarioAleatorio() ListaAleatoria() ContraseñaAleatoria() Ayuda()

## Código:

En este apartado explicamos el código dividido en las partes de sus funciones y explicando su código y los resultados que estos compilan. El código está dividido en dos partes, las funciones y el código principal.

## Definición de funciones:

A continuación, se explica a detalle las funciones que se usan en el código:

### Número aleatorio:

Consigue un número aleatorio entre el límite inferior y el límite superior definidos, además de que puede conseguir uno o más números si se solicita.

```
def NumAleatorio():
    x = 0
    inf = int(input("Ingresa el valor inferior del
límite: "))
    sup = int(input("Ingresa el valor superior del
límite: "))
    while inf >= sup:
        print("Perdona, el límite inferior es mayor al límite superior, prueba co
n otros valores")
        inf = int(input("Ingresa el valor inferior del límite:
"))
        sup = int(input("Ingresa el valor superior del límite: "))
    else:
        amount = int(input("¿Cuantos numeros aleatorios entre lo limites quieres
generar? "))
        for i in range(0, amount):
            x = x + 1
            print(x, "El número aleatorio
es:", randint(inf, sup))
```

```
Ingresa el valor inferior del límite: 1
Ingresa el valor superior del límite: 10
¿Cuantos numeros aleatorios entre lo limites quieres generar? 5
1 El número aleatorio es: 8
2 El número aleatorio es: 7
3 El número aleatorio es: 6
4 El número aleatorio es: 3
5 El número aleatorio es: 4
```

## Baraja aleatoria:

Genera una baraja inglesa y escoge aleatoriamente sin reemplazo la cantidad de cartas que se inserte.

```
def BarajaAleatoria():
    x = 0
    amount = int(input("¿Cuántas cartas aleatorias quieres sacar? "))
    while amount > 52:
        print("Perdona, la baraja solo tiene 52 cartas, por lo que solo puedes sacar menos de 52 ó 52")
        amount = int(input("¿Cuántas cartas aleatorias quieres sacar? "))
    else:
        baraja = ["A♠", "2♠", "3♠", "4♠", "5♠", "6♠", "7♠", "8♠", "9♠", "10♠",
                  "J♠", "Q♠", "K♠", "A♦", "2♦", "3♦", "4♦", "5♦", "6♦", "7♦", "8♦", "9♦", "10♦",
                  "J♦", "Q♦", "K♦", "A♣", "2♣", "3♣", "4♣", "5♣", "6♣", "7♣", "8♣", "9♣", "10♣", "J♣",
                  "Q♣", "K♣", "A♥", "2♥", "3♥", "4♥", "5♥", "6♥", "7♥", "8♥", "9♥", "10♥", "J♥",
                  "Q♥", "K♥"]
        print(sample(baraja, amount))
```

```
¿Cuántas cartas aleatorias quieres sacar? 4
['7♥', '10♥', '6♥', 'Q♠']
```

### Lanzamiento de moneda:

Interpreta los resultados en un lanzamiento de moneda el número de veces indicado.

```
def LanzamientoMoneda():  
    x = 0    amount = int(input("¿Cuántas monedas quieres  
lanzar? "))    moneda = ["Cara", "Cruz"]    for i in  
range(0, amount):  
        x = x + 1    print(x, "Lanzaste la moneda y salió:",  
moneda[randint(0, 1)])
```

```
¿Cuántas monedas quieres lanzar? 3  
1 Lanzaste la moneda y salió: Cara  
2 Lanzaste la moneda y salió: Cruz  
3 Lanzaste la moneda y salió: Cara
```

### Lanzamiento de dado:

Interpreta los resultados en un lanzamiento de dado el número de veces indicado.

```
def LanzamientoDado():  
    x = 0  
    amount = int(input("¿Cuántos dados quieres lanzar? "))  
    for i in range(0, amount):  
        x = x + 1    print(x, "Lanzaste el dado y  
salió:", randint(1,6))
```

```
¿Cuántos dados quieres lanzar? 2  
1 Lanzaste el dado y salió: 6  
2 Lanzaste el dado y salió: 4
```

### Decisión aleatoria:

Genera una decisión entre si y no las veces que sea especificado.

```
def DecisionAleatoria():
    x = 0
    amount = int(input("¿Cuántas decisiones aleatorias quieres hacer? "))
    siono = ["Si", "No"]
    for i in range(0, amount):
        x = x + 1
        print(x, "Tu decisión entre si o no es:",
              siono[randint(0, 1)])
```

```
¿Cuántas decisiones aleatorias quieres hacer? 3
1 Tu decisión entre si o no es: Si
2 Tu decisión entre si o no es: Si
3 Tu decisión entre si o no es: No
```

### Letra aleatoria:

Genera una letra aleatoria del abecedario (excluyendo la ñ) las veces que el usuario indique.

```
def LetraAleatoria():
    x = 0
    amount = int(input("¿Cuántas letras quieres obtener? "))
    alfabeto = list(range(ord("A"), ord("[")))
    for i in range(0, amount):
        x = x + 1
        print(x, "La letra aleatoria es:",
              chr(alfabeto[randint(0,25)]))
```

```
¿Cuántas letras quieres obtener? 5
1 La letra aleatoria es: H
2 La letra aleatoria es: U
3 La letra aleatoria es: W
4 La letra aleatoria es: F
5 La letra aleatoria es: G
```

## Piedra, papel o tijera:

Genera una decisión entre piedra, papel o tijera.

```
def PiedraPapelTijera():  
    x = 0    piedra = ["Piedra", "Papel", "Tijera"]  
    print("Tu oponente escogió:", piedra[randint(0, 2)])
```

Tu oponente escogió: Papel

## Coordenada geográfica aleatoria:

Genera las coordenadas aleatorias en base a la medición de grados decimales y las creará cuantas veces e le solicite.

```
def CoordenadaGeografica():  
    x = 0    if  
    menú == 8:  
        amount = int(input("¿Cuántas coordenadas en grados decimales quieres  
obtener? "))    for i in range(0, amount):  
            x = x + 1    print(x, "La coordenada aleatoria  
es latitud:", uniform(-  
89, 89), "y longitud:", uniform(-179, 179))
```

```
¿Cuántas coordenadas en grados decimales quieres obtener? 5  
1 La coordenada aleatoria es latitud: -79.55678882434164 y longitud: 59.93383017923358  
2 La coordenada aleatoria es latitud: 50.33863055818543 y longitud: 137.0337390260354  
3 La coordenada aleatoria es latitud: 3.3510368073328465 y longitud: 82.34159935444637  
4 La coordenada aleatoria es latitud: -72.716547205589 y longitud: 59.610711138552205  
5 La coordenada aleatoria es latitud: 71.69875510086419 y longitud: 0.9761972440571469
```



### Número binario aleatorio:

Genera un número binario aleatorio ingresando la base binaria y esto lo realizara el número de veces que quiera.

```
def NumBinarioAleatorio():  
    x = 0      bint = int(input("Ingresa la base binaria: "))  
    amount = int(input("¿Cuantos numeros quieres obtener? "))  
    for i in range(0, amount):  
        x = x + 1      num = randint(0, 2 **  
bint)      print(x, num, "en binario es ",  
bin(num))
```

```
Ingresa la base binaria: 6  
¿Cuantos numeros quieres obtener? 5  
1 24 en binario es 0b11000  
2 59 en binario es 0b111011  
3 28 en binario es 0b11100  
4 63 en binario es 0b111111  
5 58 en binario es 0b111010
```

### Lista aleatoria:

Genera los elementos de una lista, y aleatoriamente se escogerán esos elementos (con reemplazo) las veces que lo indique.

```
def ListaAleatoria():
    x = 0      list_ = []      num_elements = int(input("Ingresa el número
de valores de la lista: "))      for i in range(num_elements):
        element = input("Ingresa un elemento a la lista: ")
    list_.append(element)      amount = int(input("¿Cuántos elementos aleatorios de
la lista quiere obtener?
"))      for i in range(0,
amount):
        x = x + 1      print(x, "El elemento aleatorio de la lista es:",
list_[randint(0, num_el ements - 1)])
```

```
Ingresa el número de valores de la lista: 3
Ingresa un elemento a la lista: A
Ingresa un elemento a la lista: B
Ingresa un elemento a la lista: C
¿Cuántos elementos aleatorios de la lista quiere obtener? 5
1 El elemento aleatorio de la lista es: B
2 El elemento aleatorio de la lista es: C
3 El elemento aleatorio de la lista es: A
4 El elemento aleatorio de la lista es: A
5 El elemento aleatorio de la lista es: B
```

### Contraseña aleatoria:

Genera una contraseña aleatoria de 16 caracteres con números, letras en mayúscula y minúscula y símbolos.

```
def ContraseñaAleatoria(): #Función de contraseña aleatoria.
    x = 0 #Variable para numerar los resultados.
    amount = int(input("¿Cuántas contraseñas quieres generar? ")) #Indica el núme
ro de resultados que quieres obtener.
    for i in range(0, amount): #Bucle de incremento iterable.
        password = [] #Genera una contraseña vacía.
```

```

        x = x + 1 #El valor x va reasignandose cada vez que el bucle se repite (aumenta de uno en uno).
    for i in range(0, 16): #Para cada caracter de la contraseña...
        aleatorio = randint(1, 4) #Genera un número del 1 al 4.
        if aleatorio == 1: #Si el número es 1...
            password.append(chr(randint(35, 38))) #Agrega un simbolo.
        if aleatorio == 2: #Si el número es 2...
            password.append(chr(randint(48, 57))) #Agrega un número.
        if aleatorio == 3: #Si el número es 3...
            password.append(chr(randint(65, 90))) #Agrega un caracter en mayúscula.
        if aleatorio == 4: #Si el número es 4...
            password.append(chr(randint(97, 122))) #Agrega un caracter en minúscula.
    print(x, "La contraseña generada es:") #Mensaje de salida.
    print("".join(password)) #Imprime una contraseña aleatoria.

```

```

¿Cuántas contraseñas quieres generar? 3
1 La contraseña generada es:
1$I5#Ud&%$S&H7Q0
2 La contraseña generada es:
i1g77hj&1#9U09V1
3 La contraseña generada es:
PS#b&&nmiwQ7nw%U

```

### Ayuda:

Muestra la ayuda para el uso de las funciones previamente analizadas.

```

def Ayuda(): #Función de ayuda.
    print("Bienvenido a Random.py, la funcionalidad de este programa radica en generar variables aleatorias, así que te diré como funciona cada una:") #Muestra el menú de ayuda.
    print("1 - Número aleatorio entre dos números: Esta función proporciona la cantidad que desees de números aleatorios entre un límite de enteros que indiques.")

```

```

    print("2 - Baraja aleatoria: De una baraja, podrás obtener cuantas cartas aleatorias quieres de la baraja inglesa. No te quedes con un as bajo la manga.")
    print("3 - Lanzamiento de moneda: Lanza cuantas monedas desees y tendrán dos posibles resultados: Cara o Cruz.")
    print("4 - Lanzamiento de dado: Lanza cuantos dados desees y tendrás seis posibles resultados dependiendo de las caras de los dados.")
    print("5 - Si o No: Similar a lanzar una moneda, pero con las posibilidades de Si y No. Ideal si eres una persona muy indecisa.")
    print("6 - Letra aleatoria: Brinda cuantas letras quieras obtener del alfabeto inglés.")
    print("7 - Piedra papel o tijera: Lanza solo una decisión aleatoria entre Piedra, Papel o Tijera. No, no es tu amigo virtual, solo es un código")
    print("8 - Coordenada geográfica aleatoria: Brinda una ubicación geográfica aleatoria en grados decimales. Sin dudas, el favorito del autor.")
    print("9 - Número binario aleatorio: Genera un número entero entre 0 y una base binaria para mostrar dicho número y el número en binario.")
    print("0 - Lista aleatoria: Genera una lista mediante las entradas de la cantidad de elementos de la lista y sus elementos.")
    print("11 - Contraseña aleatoria: Genera contraseñas aleatorias de 16 caracteres con numeros, letras en mayúscula y minúscula y símbolos.")

```

```

Bienvenido a Random.py, la funcionalidad de este programa radica en generar variables aleatorias, así que te diré como funciona cada una:
1 - Número aleatorio entre dos números: Esta función proporciona la cantidad que desees de números aleatorios entre un límite de enteros que indiques.
2 - Baraja aleatoria: De una baraja, podrás obtener cuantas cartas aleatorias quieres de la baraja inglesa. No te quedes con un as bajo la manga.
3 - Lanzamiento de moneda: Lanza cuantas monedas desees y tendrán dos posibles resultados: Cara o Cruz.
4 - Lanzamiento de dado: Lanza cuantos dados desees y tendrás seis posibles resultados dependiendo de las caras de los dados.
5 - Si o No: Similar a lanzar una moneda, pero con las posibilidades de Si y No. Ideal si eres una persona muy indecisa.
6 - Letra aleatoria: Brinda cuantas letras quieras obtener del alfabeto inglés.
7 - Piedra papel o tijera: Lanza solo una decisión aleatoria entre Piedra, Papel o Tijera. No, no es tu amigo virtual, solo es un código
8 - Coordenada geográfica aleatoria: Brinda una ubicación geográfica aleatoria en grados decimales. Sin dudas, el favorito del autor.
9 - Número binario aleatorio: Genera un número entero entre 0 y una base binaria para mostrar dicho número y el número en binario.
0 - Lista aleatoria: Genera una lista mediante las entradas de la cantidad de elementos de la lista y sus elementos.
11 - Contraseña aleatoria: Genera contraseñas aleatorias de 16 caracteres con numeros, letras en mayúscula y minúscula y símbolos.

```

## Menú (MAIN):

Esta es la función MAIN, es decir, la función principal del programa y nos ayuda a escoger las demás funciones y que el uso de todo el programa funcione como debería.

```

def Menu(menu): #Función menú (MAIN).
    if menu == 1: #Si el menu es 1...
        NumAleatorio() #Ejecuta la función de número aleatorio.
    if menu == 2: #Si el menu es 2...
        BarajaAleatoria() #Ejecuta la función de baraja aleatoria.
    if menu == 3: #Si el menu es 3...
        LanzamientoMoneda() #Ejecuta la función de lanzamiento de moneda aleatorio.

```

```
if menu == 4: #Si el menu es 4...
    LanzamientoDado() #Ejecuta la función de lanzamiento de dado aleatorio.
if menu == 5: #Si el menu es 5...
    DecisionAleatoria() #Ejecuta la función de desición aleatoria.
if menu == 6: #Si el menu es 6...
    LetraAleatoria() #Ejecuta la función de letra aleatoria.
if menu == 7: #Si el menu es 7...
    PiedraPapelTijera() #Ejecuta la función de pierda, papel o tijera.
if menu == 8: #Si el menu es 8...
    CoordenadaGeografica() #Ejecuta la función de coordenada geográfica aleatoria.
if menu == 9: #Si el menu es 9...
    NumBinarioAleatorio() #Ejecuta la función de número binario aleatorio.
if menu == 0: #Si el menu es 0...
    ListaAleatoria() #Ejecuta la función de lista aleatoria.
if menu == 11: #Si el menu es 11...
    ContraseñaAleatoria() #Ejecuta la función de contraseña aleatoria.
if menu == 99: #Si el menu es 99...
    Ayuda() #Ejecuta la función de ayuda.
```

## MENU:

- 1 - Número aleatorio entre dos números
- 2 - Baraja aleatoria
- 3 - Lanzamiento de moneda
- 4 - Lanzamiento de dado
- 5 - Si o No
- 6 - Letra aleatoria
- 7 - Piedra papel o tijera
- 8 - Coordenada geográfica aleatoria
- 9 - Número binario aleatorio
- 0 - Lista aleatoria
- 11 - Contraseña aleatoria
- 99 - Ayuda

## Código principal:

Finalmente, el código principal es el que reúne todas las funciones anteriores y las ordena de modo que nuestro código se ejecute. Contiene básicamente la importación de las funciones que se usan, la interfaz gráfica del menú y la corrección de error.

```
#INICIA EL CÓDIGO:
from random import randint, sample, uniform #Modulo de random (randint genera un
número entero aleatorio, sample elige aleatoriamente una opción de una lista si
n reemplazo, uniform genera un número decimal aleatorio).
print("\n-----\nBienvenido a Random miscellaneous\n-
-----")
") #Mensaje de bienvenida (solo se muestra la primera vez).
while True: #Repite el programa siempre que lo parámetros se cumplan.
    menu = int(input("\nMENU:\n1 - Número aleatorio entre dos números\n2 - Baraj
a aleatoria\n3 - Lanzamiento de moneda\n4 - Lanzamiento de dado\n5 - Si o No\n6
- Letra aleatoria\n7 - Piedra papel o tijera\n8 - Coordenada geográfica aleatori
a\n9 - Número binario aleatorio\n0 - Lista aleatoria\n11 - Contraseña aleatoria\
n99 - Ayuda\n")) #Menu que pide la primer variable para seguir el programa.
    while menu != 1 and menu != 2 and menu != 3 and menu != 4 and menu != 5 and
menu != 6 and menu != 7 and menu != 8 and menu != 9 and menu != 0 and menu != 11
and menu != 99:
        print("\n-----")
        print("Porfavor, ingresa una opción válida") #Mensaje de error.
        print("-----")
        menu = int(input("\nMENU:\n1 - Número aleatorio entre dos números\n2 - B
araja aleatoria\n3 - Lanzamiento de moneda\n4 - Lanzamiento de dado\n5 - Si o No
\n6 - Letra aleatoria\n7 - Piedra papel o tijera\n8 - Coordenada geográfica alea
toria\n9 - Número binario aleatorio\n0 - Lista aleatoria\n11 - Contraseña aleato
ria\n99 - Ayuda\n")) #Menu que pide la primer variable para seguir el programa.
    else:
        Menu(menu) #Ejecuta la función de menú (MAIN).
```