

Módulo 2.4: Aleatoriedad

Prof. Carlos Cedeño

La aleatoriedad en Python te sirve para introducir imprevisibilidad y variabilidad en tus programas. Es útil en un montón de situaciones, siempre que necesites que tu programa tome decisiones no deterministas, simule incertidumbre, o genere datos únicos y variados, la aleatoriedad es tu aliada.

En Python, los números aleatorios se generan principalmente utilizando el módulo `random`.

Para empezar, necesitas importar el módulo `random`:

```
import random # Esta linea va al inicio del script
```

Generar números aleatorios

Una vez importado, puedes usar varias funciones según lo que necesites:

Enteros aleatorios

- `random.randint(a, b)` : Devuelve un entero aleatorio N tal que `a <= N <= b` . Es inclusivo en ambos extremos.

```
numero_entero = random.randint(1, 10) # Genera un número entre 1 y 10 (ambos incluidos)
print(numero_entero)
```

- `random.randrange(inicio, fin, paso)` : Devuelve un elemento seleccionado aleatoriamente de `range(inicio, fin, paso)` . No incluye el valor de `fin` .

```
numero_rango = random.randrange(0, 10, 2) # Genera un número par entre 0 y 8 (0, 2, 4, 6, 8)
print(numero_rango)
```

Números de punto flotante aleatorios

- `random.random()` : Devuelve el siguiente número de punto flotante aleatorio en el rango `[0.0, 1.0)` (incluye 0.0 pero no 1.0).

```
numero_flotante = random.random() # Genera un número decimal entre 0.0 y casi 1.0
print(numero_flotante)
```

- `random.uniform(a, b)` : Devuelve un número de punto flotante aleatorio N tal que `a <= N <= b` .

```
numero_uniforme = random.uniform(1.5, 5.5) # Genera un número decimal entre 1.5 y 5.5
print(numero_uniforme)
```

Selecciones aleatorias de secuencias

El módulo `random` también te permite hacer selecciones aleatorias de listas, tuplas u otras secuencias.

- `random.choice(secuencia)` : Devuelve un elemento aleatorio de una secuencia no vacía.

```
mi_lista = ["manzana", "banana", "cereza"]
elemento_aleatorio = random.choice(mi_lista)
print(elemento_aleatorio)
```

- `random.shuffle(sequencia)` : Mezcla la secuencia `sequencia` en el lugar. Es decir, modifica la lista original.

```
numeros = [1, 2, 3, 4, 5]  
random.shuffle(numeros) # Mezcla la lista 'numeros'  
print(numeros)
```

Recuerda que estos números son **pseudoaleatorios**, lo que significa que son generados por un algoritmo determinista. Para la mayoría de los propósitos (simulaciones, juegos, etc.), son suficientemente aleatorios. Si necesitas números aleatorios para propósitos criptográficos, deberías usar el módulo `secrets` .