

## The `random` module in Python:

---

### Purpose:

Generates pseudo-random numbers and performs random actions for various tasks in Python.

### Key Functions:

`random.randint(a, b):`

Returns a random integer between `a` (inclusive) and `b` (inclusive).

Example: `random.randint(1, 6)` generates a random roll of a dice.

`random.randrange(start, stop, step):`

Returns a randomly selected element from a range, similar to `range()` but with random selection.

`start` is included, `stop` is excluded.

`step` specifies the step size (default is 1).

`random.choice(sequence):`

Returns a randomly selected element from a non-empty sequence (list, tuple, string).

Raises `IndexError` if the sequence is empty.

`random.shuffle(list):`

Randomly shuffles the elements of a list in-place.

### Other Useful Functions:

`random.random():` Returns a random floating-point number between 0.0 and 1.0.

`random.seed(a):` Sets the seed for the random number generator, ensuring reproducibility.

### Important Notes:

- The module generates pseudo-random numbers, meaning they are not truly random but appear random for practical purposes.
- If you need cryptographically secure random numbers, use the `secrets` module.
- For more advanced random number generation from different distributions, consider the `numpy.random` module.

### Example:

Python

```
import random
```

```
# Generate a random integer between 1 and 10
number = random.randint(1, 10)
print(number)
```

```
# Choose a random element from a list
fruits = ["apple", "banana", "orange"]
random_fruit = random.choice(fruits)
print(random_fruit)
```

```
# Shuffle a list of numbers
numbers = [1, 2, 3, 4, 5]
random.shuffle(numbers)
print(numbers)
```

### Some other functions from random module

The `random.sample()` function in the `random` module is used to generate a random sample from a given sequence (such as a list or range) without replacement. "Without replacement" means that once an element is selected, it is not available for selection again in the same sample.

### Syntax:

```
random.sample(population, k)
```

- **population**: This is the input sequence (e.g., a list or range) from which the sample is taken.
- **k**: This is the size of the sample.

The function returns a list containing unique elements randomly chosen from the population.

**For example:**

```
row = random.sample(range(1, 10), 5)
```

Generates a list `row` containing 5 unique random numbers between 1 and 9 (inclusive). This ensures that each row in the Housie ticket has unique numbers.