

Python Random

Textbook

Random



Python has the ability to generate random numbers. Using functions that are built into the Python language, generating random numbers is easy. We use randomness to simulate unexpected factors in the real world. With random numbers, we can represent the way things change and have varying values in the real world.

Let's explore how to get [random](#) numbers in Python.

First, `import random`. This accesses a function that is built in to the Python language that generates random numbers.

Then use `random.random()`

```
1 import random
2 x = random.random()
3 print(x)
```

Try it!

This will print out a random float between 0 and 1. Notice if you hit "run" multiple times it will give you different random values.

Notice that the pseudocode looks different. For the AP CSP pseudocode, you don't need to import `random`.

`randint()`

Now try using `random.randint(0,10)`

```
1 import random
2 x = random.randint(1,6)
3 print(x)
```

Try it!

This prints out a random integer in the specified range. Notice again that if you hit "run" multiple times, you'll get random values. Congratulations! We have just programmed a simulation of rolling a dice! Each time we roll a dice, we get a random value between one and six, which is what our program does! Simple dice rolling apps get thousands of downloads a day. Imagine what you could build!

`random.sample()`

Let's try using `random.sample()`. This allows you to choose how many random numbers you will get in a certain range.

```
1 import random
2 mylist = random.sample(range(0, 10), 5)
3
4 print(mylist)
```

Try it!

This prints out a list of five random numbers between 0 and 10.

Generating a List of Random Numbers

```
1 import random
2 mylist = []
3 for x in range(0,5):
4     myvalue = random.randint(0,10)
5     mylist.append(myvalue)
6 print(mylist)
7
```

Try it!

This will print out a list of random numbers. Changing the values in the parentheses for `range(0,5)` will change the length of your list. Changing the values in the parentheses for `random.randint(0,10)` will change what the numbers can vary between.

Rock Paper Scissors Example

Let's use random to create a game using Python! Have you ever played rock, paper, scissors? Players choose one of those three options and depending on what their opponent chooses, they win or lose.

Paper beats Rock

Rock beats Scissors

Scissors beat Paper

Using random, see if you can create a rock paper scissors game yourself!

Hint: Use this code to get you started:

```
1 import random
2
3 user_action = input("Enter a choice (rock, paper, scissors): ")
4 possible_actions = ["rock", "paper", "scissors"]
5 choice = random.randint(0,2)
6
7 print(choice)
```

Try it!

See if you can create the game yourself! The answer can be found below.

```
1 import random
2
3 user_action = input("Enter a choice (rock, paper, scissors): ")
4 possible_actions = ["rock", "paper", "scissors"]
5 choice = random.randint(0,2)
6 computer_action = possible_actions[choice]
7 print("You chose " + user_action + ", computer chose " + computer_action + ".")
8 if user_action == computer_action:
9     print("Both players selected " + user_action + " . It's a tie!")
10 elif user_action == "rock":
11     if computer_action == "scissors":
12         print("Rock smashes scissors! You win!")
13     else:
14         print("Paper covers rock! You lose.")
15 elif user_action == "paper":
16     if computer_action == "rock":
17         print("Paper covers rock! You win!")
18     else:
19         print("Scissors cuts paper! You lose.")
20 elif user_action == "scissors":
21     if computer_action == "paper":
22         print("Scissors cuts paper! You win!")
23     else:
24         print("Rock smashes scissors! You lose.")
```

Try it!

Real World Connections

Now, let's bridge the gap between what we learn in the classroom and the world around us. In this section, you'll get to apply mathematics to real-world contexts. This means actively looking for ways to connect mathematical concepts to your everyday experiences, seeing how numbers, patterns, and logic play a role in everything from calculating discounts to designing apps.

You'll use various models and methods to understand, represent, and solve problems that pop up in real life. Be ready to perform investigations to gather your own data or test if a certain mathematical approach truly fits the situation.

And remember, the goal isn't always perfection on the first try; you'll have opportunities to redesign your models and methods to make them even more accurate or efficient in solving those real-world challenges.

Real-World Context: Imagine a doctor trying to understand how a new medication might affect different patients. It's impossible to test it on everyone, and real people have countless unpredictable differences (like diet, other medications, stress levels). To simulate these "unexpected factors," doctors and scientists might use computer programs that generate random numbers to represent these variations in a large group of virtual patients. This helps them predict the medication's likely effects and side effects without having to test it on thousands of actual people first, saving time, money, and ensuring safety.

Checkpoint

Python Random

Practice using random numbers!

Import **random**.

Use what was taught in this lesson to do the following.

Print out a random float between **0 and 1**.

Print out a random integer between **0 and 5**.

Print out **3** random integers between **0 and 100**. Make sure they appear as a list.

Requirements:

- Import random
- Print out a random float between 0 and 1.
- Print out a random integer between 0 and 5.
- Print out 3 random integers between 0 and 100. Make sure they appear as a list.

Questions (4)

1. What must you include in your code before you can generate random numbers?

MULTIPLE CHOICE

Choose the correct answer:

- A. import random
- B. generate random
- C. random
- D. import RAND

2. Which of the following gives you a random float between 0 and 1?**Choose the correct answer:**

- A. `random.randint(0, 5)`
- B. `random.random()`
- C. `random.sample(range(0,10), 5)`

3. Which of the following will give you a random integer between 0 and 10?**Choose the correct answer:**

- A. `random.random()`
- B. `random.randint(0,10)`
- C. `random.sample(range(0,10), 5)`

4. Which of the following will give you 3 random numbers between 0 and 10?**Choose the correct answer:**

- A. `random.sample(range(0,10), 3)`
- B. `random.random()`
- C. `random.randint(0, 10)`

Challenges (5)

1. Two Dice Roll

Create a program that simulates two dice rolling.

Import **random**.

Generate two random numbers from **1-6**. Use two `random.randint()` statements for this challenge.

Print the random number for each dice.

Create a variable named `total` and set it equal to the value of the dice added together.

Requirements:

- Import Random
- Generate two random numbers from 1-6. Use two `random.randint()` statements for this challenge.
- Create a variable named `total` and set it equal to the value of the dice added together.

2. Random Percentage

Create a program that will generate a random percentage in decimal format.

In other words, create a program that will generate random floats with **two decimal places**.

Requirements:

- Import Random
- Generate values between 0 and 1
- Make sure the values are two decimal places.

3. Number Guesser

Create a number guessing game!

The program will generate a random number between **0 and 5**.

The user will input a number guess between **0 and 5**.

The program will print whether or not the user's guess is the same as the generated number.

Requirements:

- Import Random
- Generate a random number between 0 and 5.
- Ask the user to guess a number between 0 and 5.
- If the user guess is correct, print "You guessed correct!" If not, print "You guessed wrong!"

4. Yahtzee

Create the game of Yahtzee!

Yahtzee is a dice game where you roll **5** dice. You score big points if you roll **5** dice and they are all the same number.

Create a program that produces a list of **5** numbers. Make sure it appears as a single python list.

Each number will be a random number from **1 to 6**.

Each time you hit **"run"** you are shooting for all **5 dice** to be the same.

Note: The syntax "for x in range()" is required to pass.

Note: For this challenge, use randint instead of random.sample.

Requirements:

- Import Random
- Create an empty list
- Generate the list of random numbers.

5. Count the Even Numbers

Create a program that generates a list of **10 numbers**. **These numbers will be random between 0 and 100.**

Print out the **list** of numbers.

Print the **total** number of even numbers that appear in the list as well.

Note: The syntax "for x in range()" is required to pass.

Requirements:

- Import Random
- Generate the list of 10 numbers. Make sure they are random between 0 and 100
- Use a for loop to check if each value in the list is even or not.
- Create a print statement that says "You have _____ even numbers in your list." Add the correct number of even numbers in the blank.

Answer Keys & Solutions

Checkpoint Solutions

Python Random

```
1 import random
2 x = random.random()
3 print(x)
4 y = random.randint(0,5)
5 print(y)
6 z = random.sample(range(0,100), 3)
7 print(z)
```

Questions

1. What must you include in your code before you can generate random numbers?

MULTIPLE CHOICE

Correct Answer:

- A. import random ✓ Correct
- B. generate random ✗ Incorrect
- C. random ✗ Incorrect
- D. import RAND ✗ Incorrect

Explanation:

Remember that Python already has the functions for random numbers, but you need to import them first.

2. Which of the following gives you a random float between 0 and 1?

MULTIPLE CHOICE

Correct Answer:

- A. random.randint(0, 5) ✗ Incorrect
- B. random.random() ✓ Correct
- C. random.sample(range(0,10), 5) ✗ Incorrect

Explanation:

The default for `random()` is to generate a float

3. Which of the following will give you a random integer between 0 and 10?

MULTIPLE CHOICE

Correct Answer:

- A. `random.random()` ✗ Incorrect
- B. `random.randint(0,10)` ✓ Correct
- C. `random.sample(range(0,10), 5)` ✗ Incorrect

Explanation:

The "int" in `randint` means integer.

4. Which of the following will give you 3 random numbers between 0 and 10?

MULTIPLE CHOICE

Correct Answer:

- A. `random.sample(range(0,10), 3)` ✓ Correct
- B. `random.random()` ✗ Incorrect
- C. `random.randint(0, 10)` ✗ Incorrect

Explanation:

`random.sample()` lets you pick out a specific number of random numbers

Challenges

1. Two Dice Roll

Solution:

```
1 import random
2
3 first = random.randint(1,6)
4
5 print(first)
```

```
6
7 second = random.randint(1,6)
8
9 print(second)
10
11 total = first + second
12
13 print(total)
```

2. Random Percentage

Solution:

```
1 import random
2 x = round(random.random(),2)
3 print(x)
```

3. Number Guesser

Solution:

```
1 import random
2
3 guess = int(input("What number do you want to guess between 0 and 5?"))
4
5 secret = random.randint(0,5)
6
7
8 if guess == secret:
9     print("You guessed right!")
10 else:
11     print("You guessed wrong!")
```

4. Yahtzee

Solution:

```
1 import random
2 mylist = []
3 for x in range(1,6):
4     myvalue = random.randint(1,6)
5     mylist.append(myvalue)
6 print(mylist)
```

5. Count the Even Numbers

Solution:

```
1 import random
2 mylist = []
3 for x in range(0, 10):
4     myvalue = random.randint(0, 100)
5     mylist.append(myvalue)
6 print(mylist)
7
8 even = 0
9 for x in mylist:
10     if x % 2 == 0:
11         even += 1
12
13 print("You have " + str(even) + " even numbers in your list.")
```