# Important Inbuilt Libraries

### **Math Library**

The *math* module is a standard Python module providing mathematical functions. It includes a wide range of functions for basic arithmetic, trigonometry, logarithmic, exponential, and more. Here are some commonly used functions from the *math* module:

### **Basic Arithmetic Functions**

### 1. 'math.ceil(x)`:

Returns the smallest integer greater than or equal to x.

```
Python3

1 import math
2 result = math.ceil(4.2)
3 print(result)

Output - 5
```

# 2. math. floor(x):

Returns the largest integer less than or equal to x.

```
Python3

1 import math
2 result = math.floor(4.9)
3 print(result)

Output - 4
```

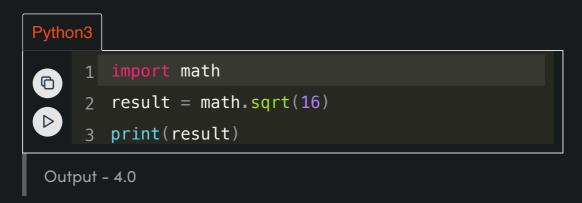
# 3. math.trunc(x):

Returns the truncated integer value of x.

# Python3 1 import math 2 result = math.trunc(4.9) 3 print(result) Output - 4

4. math. sqrt(x):

Returns the square root of x.



# **Trigonometric Functions**

5. math. sin(x), math. cos(x), math. tan(x):

Return the sine, cosine, and tangent of x (measured in radians).

```
Python3

1 import math
2 angle = math.radians(45) # Convert degrees
3 sin_result = math.sin(angle)
4 cos_result = math.cos(angle)
5 tan_result = math.tan(angle)
6 print("sin_result :",sin_result, "cos_result

Output - sin_result : 0.7071067811865475 cos_result :
```

0.7071067811865476 tan\_result : 0.9999999999999999

# **Logarithmic and Exponential Functions**

# 6. math. log(x, base):

Returns the natural logarithm of x with the specified base.

```
Python3

1 import math
2 result = math.log(16, 2)
3 print(result)

output - 4.0
```

# 7. math. exp(x):

Returns e raised to the power of x.

```
Python3

1 import math
2 result = math.exp(2)
3 print(result)

Output - 7.3890560989306495
```

### **Constants**

## 8. *math. pi*:

Mathematical constant representing the ratio of the circumference of a circle to its diameter.

### 9. *math.e*:

Mathematical constant representing the base of the natural logarithm.

# Python3 1 import math 2 print(math.pi) 3 print(math.e) Output - 3.141592653589793

You can explore more functions and constants in the official Python documentation: https://docs.python.org/3/library/math.html

### **Random Library**

2.718281828459045

The *random* module in Python is a standard library module that provides functions for generating random numbers, selecting random elements, and performing various randomization tasks. It is commonly used in applications involving simulations, games, statistical sampling, cryptography, and more.

Here are some commonly used functions from the *random* module:

### **Generating Random Numbers:**

# 1. *random()*:

Returns a random floating-point number in the range [0.0, 1.0).

```
Python3

1 import random
2 value = random.random()
3 print(value)

Output - 0.14277691938900905
```

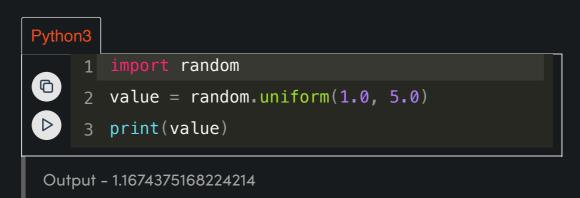
# 2. randint(a, b):

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

# Python3 1 import random 2 value = random.randint(1, 10) 3 print(value) Output - 7

# 3. uniform(a, b):

Returns a random floating-point number in the range [a, b) or [a, b] if b is included.



## **Randomizing Sequences:**

# 4. shuffle(sequence):

Randomly shuffles the elements of a sequence in place.

```
Python3

1 import random
2 my_list = [1, 2, 3, 4, 5]
3 random.shuffle(my_list)
4 print(my_list)

Output - [5, 3, 2, 4, 1]
```

# 5. choice(sequence):

Returns a random element from a sequence.

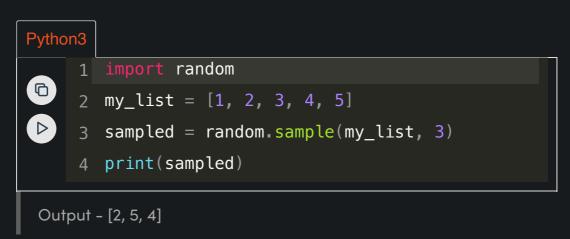
```
Python3

1 import random
2 my_list = [1, 2, 3, 4, 5]
3 value = random.choice(my_list)
4 print(value)

Output - 4
```

# 6. sample(sequence, k):

Returns a new list with  ${\it k}$  unique random elements sampled from the sequence without replacement.



### **DateTime Library**

In Python, the *datetime* module provides classes for working with dates and times. It allows you to represent dates, times, and intervals, and perform operations on them. Here are some key components of the *datetime* module:

### 1. datetime Class:

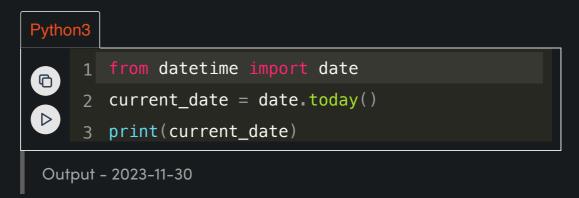
The *datetime* class represents a date and a time. It has attributes like year, month, day, hour, minute, second, and microsecond.

```
Python3

1 from datetime import datetime
2 current_datetime = datetime.now()
3 print(current_datetime)
```

### 2. date Class:

The *date* class represents a date (year, month, and day) without a time component.



### 3. time Class:

The *time* class represents a time (hour, minute, second, and microsecond) without a date component.

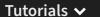
### Formatting and Parsing:

You can format *datetime* objects as strings and parse strings into *datetime* objects using the *strftime* and *strptime* methods.

```
Python3

1 from datetime import datetime
2 current_datetime=datetime.now()
3 # Format a datetime object as a string
4 formatted_date = current_datetime.strftime("s
5 print(formatted_date)
6
7 # Parse a string into a datetime object
8 parsed_date = datetime.strptime("2023-01-01 : 9 print(parsed_date)
```





















All



Article



Videos



Probler



Quiz



Next >>

### **Date Arithmetic:**

Performing arithmetic operations with dates is possible using the timedelta class.

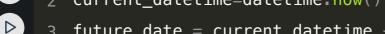




# Python3



- 1 from datetime import datetime, timedelta
- 2 current\_datetime=datetime.now()



- 3 future\_date = current\_datetime + timedelta(date)
- 4 print(future\_date)

Output - 2023-12-07 08:29:56.154919

For more details and examples, refer to the official Python documentation on the *datetime* module https://docs.python.org/3/library/datetime.html

## **Challenge Galore**

Problem Statement - Write a Python program that calculates the remaining days, hours, and minutes until the event. Additionally, provide the percentage completion of the event based on the current time compared to the event's start time.

# Python3

```
current_datetime = datetime.now()
   time_difference = event_date - current_datet
8
   remaining_days = time_difference.days
   remaining_hours, remaining_seconds = divmod(
10
   remaining_minutes = remaining_seconds // 60
11
12
13
   total_seconds_in_event = (event_date - event_
14
   elapsed_seconds = (current_datetime - event_
15
   percentage_completion = (elapsed_seconds / t
16
17
18
   print(f"Remaining time until the event: {rem
19
   print(f"Percentage completion of the event:
20
```

Output - Remaining time until the event: 31 days, 9 hours, 0 minutes.

Percentage completion of the event: -3971%

Mark as Read



If you are facing any issue on this page. Please let us know.