MODULES:

Math Module:

The math module provides extensive mathematical functions for calculations, trigonometry, constants and more.

Common Usages

1) • math.pi : Provides access to the value of pi (3.14159)
• math.sqrt(x) : Calculates the square root of x.
• math.ceil(x) : Round x up to the nearest integer.
• math.floor(x) : Round x down to the nearest integer.
• Trigonometric functions:
math.sin(x), math.cos(x), math.tan(x) etc.
• Exponential and logarithmic functions:
math.exp(), math.log(x), etc.

RANDOM MODULE:

The random module generates pseudorandom numbers for various purposes like simulations and games.

Common Usages

random.seed(seed_value) : Set a seed for more predictable randomness.
• random.randint(a, b) : Generate a random integer between a (inclusive) and b (inclusive).
random.random() : Generate a random floating-point number between 0.0 (inclusive) and 1.0 (exclusive).
random.choice(sequence) : Choose a random element from the given sequence (list, tuple, string).

Shuffling list

random.shuffle(mylist)

PAGE NO:

Time and Date Modules

- time Module: Provides functions for working with time values.

Common Usages

- time Module: Provides functions for working with time values.

① datetime.datetime.now(): Create a date time object representing the current date and time

o datetime.datetime(year, month, day, hour, minute, second): It means to create a date-time object from specific components.

Json Module

Functionality: Enables working with JSON (JavaScript Object Notation) data, a popular format for data exchange.

Common Usages

Encoding

json.dumps(data): Converts a python object (dict, list, etc) into a Json string.

Decoding

→ json.loads(json_string): Convert a Json string into a Python object.

Success has a simple formular do your best, and people may like it.

Regular Expression Module:

Functionality: Provides powerful tools for text processing and pattern matching using regular expressions.

- Common usages

import re → to import the regular expression module.

re.compile(pattern): Compile a Regular expression pattern for efficiency.

pattern.search(string): Search for the first occurance of the python anywhere in the string.

pattern.findall(string): Find all non-overlapping occurances of the pattern in the string.

pattern.sub(replacement, string): Substitute occurences of the pattern with replacement string