CIS 412 L Artificial Intelligence Lab
Topic - 02
Introduction to Python for Machine Learning

☎ +88 01831-661534

🔗 jehadfeni@gmail.com

©MJS Global Ltd by ABK Bhuiyan

Python Identity Operators?

The identity operators are used to decide whether an element certain class or type.

IS: It is evaluated to be true if the reference present at both sides point to the same object.

Is not : It is evaluated to be true if the reference present at both sides do not point to the same object.

Python Membership Operators

Python membership operators are used to check the membership of value inside a Python data structure. If the value is present in the data structure, then the resulting value is true otherwise it returns false.

In: It is evaluated to be true if the first operand is found in the second operand (list, tuple, or dictionary).

Not in: It is evaluated to be true if the first operand is not found in the second operand (list, tuple, or dictionary).

Decision Making (Conditionals)

Decision making is the most important aspect of almost all the programming languages. As the name implies, decision making allows us to run a particular block of code for a particular decision. Here, the decisions are made on the validity of the particular conditions. Condition checking is the backbone of decision making.

In python, decision making is performed by the following statements.

- **If Statement**
- **If - else Statement**
- **If-elif statement**
- **Nested if Statement**

The pass statement: in python if statement cannot be empty but for some reason if need an if statement without having content, you can use **pass** statement to avoid interpreter error.

Iteration in Python

- Like all other programming languages python also have iteration support. There are two way to iterate python statement.
- **For loop:** for loop is used to iterate the statements or a part of the program several times. It is frequently used to traverse the data structures like list, tuple, or dictionary.
- **While loop:** The Python while loop allows a part of the code to be executed until the given condition returns false. It is also known as a pre-tested loop.

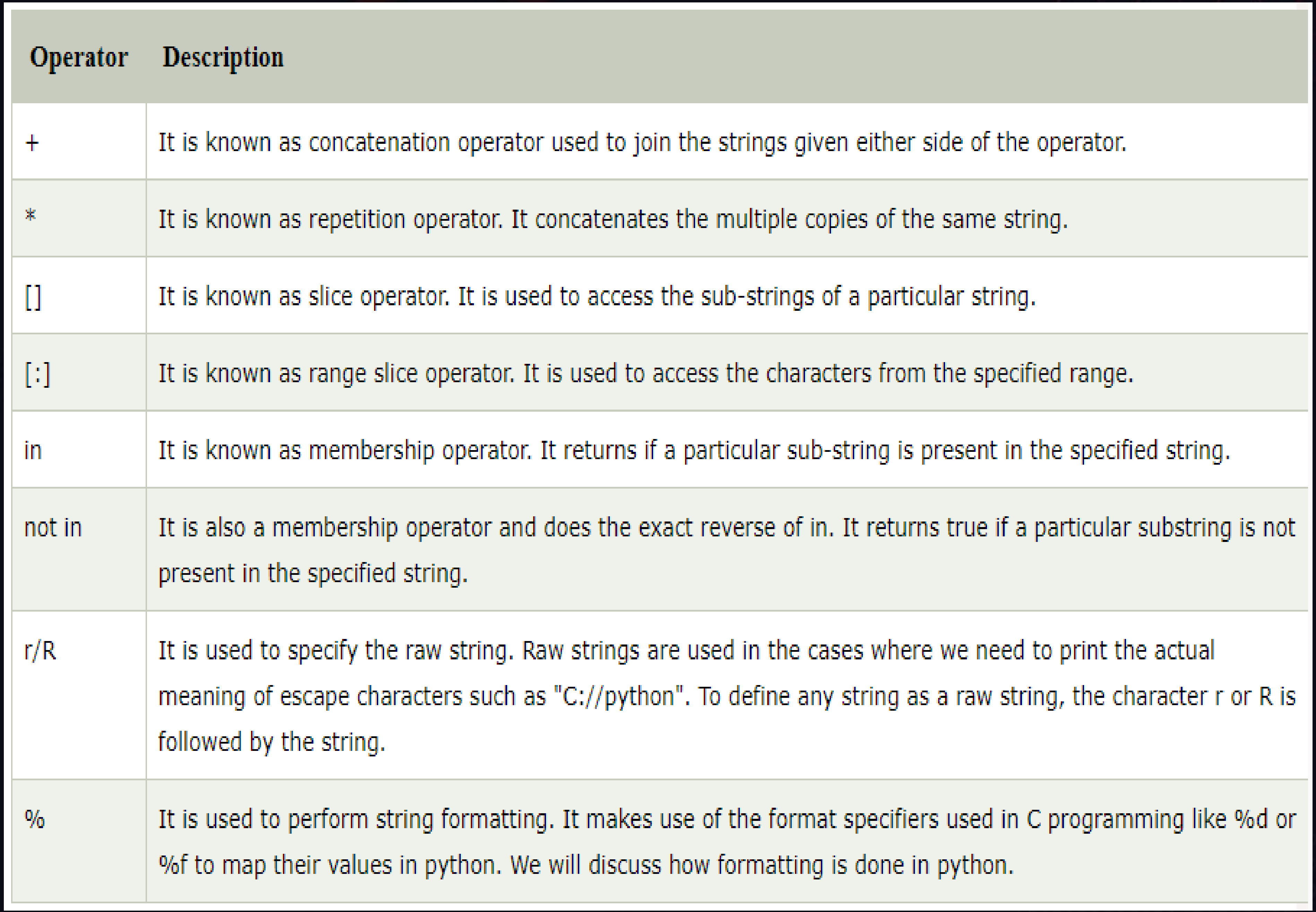
Continue, Break & Pass Statement

- **Continue Statement:** When the continue statement is encountered, the control transfer to the beginning of the loop.
- **Break Statement:** When the break statement is encountered, it brings control out of the loop.
- **Pass Statement:** he pass statement is used to declare the empty loop. It is also used to define empty class, function, and control statement.

Python Strings

- Python string is the collection of the characters surrounded by single quotes, double quotes, or triple quotes. The computer does not understand the characters; internally, it stores manipulated character as the combination of the 0's and 1's.
- Each character is encoded in the ASCII or Unicode character. So we can say that Python strings are also called the collection of Unicode characters.

String Op



Operator	Description
+	It is known as concatenation operator used to join the strings given either side of the operator.
*	It is known as repetition operator. It concatenates the multiple copies of the same string.
[]	It is known as slice operator. It is used to access the sub-strings of a particular string.
[:]	It is known as range slice operator. It is used to access the characters from the specified range.
in	It is known as membership operator. It returns if a particular sub-string is present in the specified string.
not in	It is also a membership operator and does the exact reverse of in. It returns true if a particular substring is not present in the specified string.
r/R	It is used to specify the raw string. Raw strings are used in the cases where we need to print the actual meaning of escape characters such as "C://python". To define any string as a raw string, the character r or R is followed by the string.
%	It is used to perform string formatting. It makes use of the format specifiers used in C programming like %d or %f to map their values in python. We will discuss how formatting is done in python.

List of ar escape sequenc e

Sr.	Escape Sequence	Description	Example
1.	\newline	It ignores the new line.	print("Python1 \ Python2 \ Python3")**Output:**Python1 Python2 Python3
2.	\\	Backslash	print("\\")**Output:**\
3.	\'	Single Quotes	print('\'')**Output:**'
4.	\\"	Double Quotes	print("\"")**Output:**"
5.	\a	ASCII Bell	print("\a")
6.	\b	ASCII Backspace(BS)	print("Hello \b World")**Output:**Hello World
7.	\f	ASCII Formfeed	print("Hello \f World!") Hello World!
8.	\n	ASCII Linefeed	print("Hello \n World!")**Output:**Hello World!
9.	\r	ASCII Carriege Return(CR)	print("Hello \r World!")**Output:**World!
10.	\t	ASCII Horizontal Tab	print("Hello \t World!")**Output:**Hello World!
11.	\v	ASCII Vertical Tab	print("Hello \v World!")**Output:**Hello World!
12.	\ooo	Character with octal value	print("\110\145\154\154\157")**Output:** Hello
13	\xHH	Character with hex value.	print("\x48\x65\x6c\x6c\x6f")**Output:**Hello

Format

- The **format()** method is the most flexible and useful method in formatting strings. The curly braces {} are used as the placeholder in the string and replaced by the **format()** method argument.

```
# Using Curly braces
print("{} and {} both are the best friend".format("Devansh", "Abhishek"))

#Positional Argument
print("{1} and {0} best players ".format("Virat","Rohit"))

#Keyword Argument
print("{a},{b},{c}".format(a = "James", b = "Peter", c = "Ricky"))
```

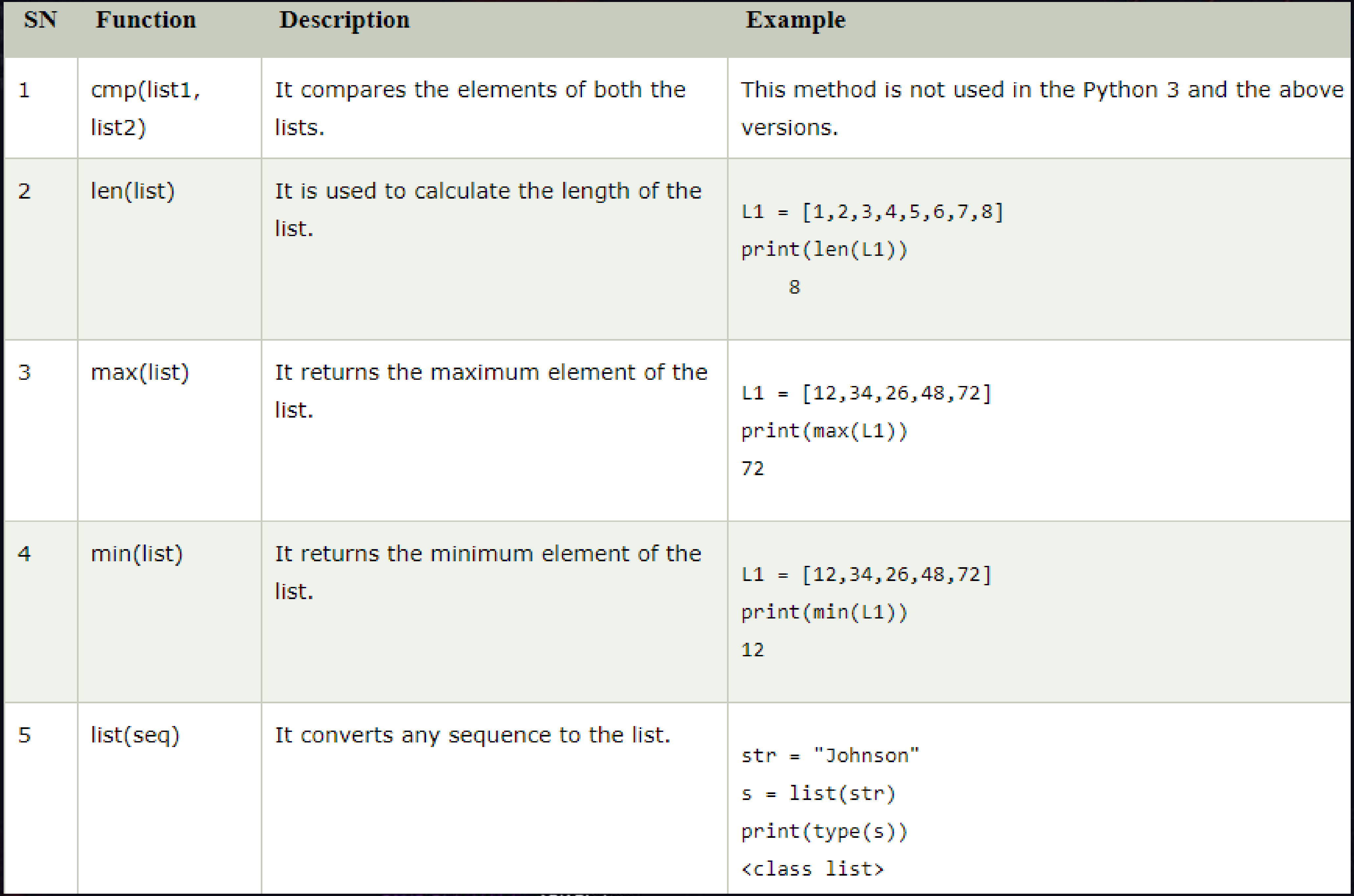
Python List

- A list in Python is used to store the sequence of various types of data. Python lists are mutable type its mean we can modify its element after it created. However, Python consists of six data-types that are capable to store the sequences, but the most common and reliable type is the list.
- A list can be defined as a collection of values or items of different types. The items in the list are separated with the comma (,) and enclosed with the square brackets [].

Characteristics of List

- The list has the following characteristics:
  - The lists are ordered.
  - The element of the list can access by index.
  - The lists are the mutable type.
  - The lists are mutable types.
  - A list can store the number of various elements.

Built In list functions



SN	Function	Description	Example
1	cmp(list1, list2)	It compares the elements of both the lists.	This method is not used in the Python 3 and the above versions.

2	len(list)	It is used to calculate the length of the list.	`L1 = [1,2,3,4,5,6,7,8]`
			`print(len(L1))`
			`    8`

3	max(list)	It returns the maximum element of the list.	`L1 = [12,34,26,48,72]`
			`print(max(L1))`
			`72`

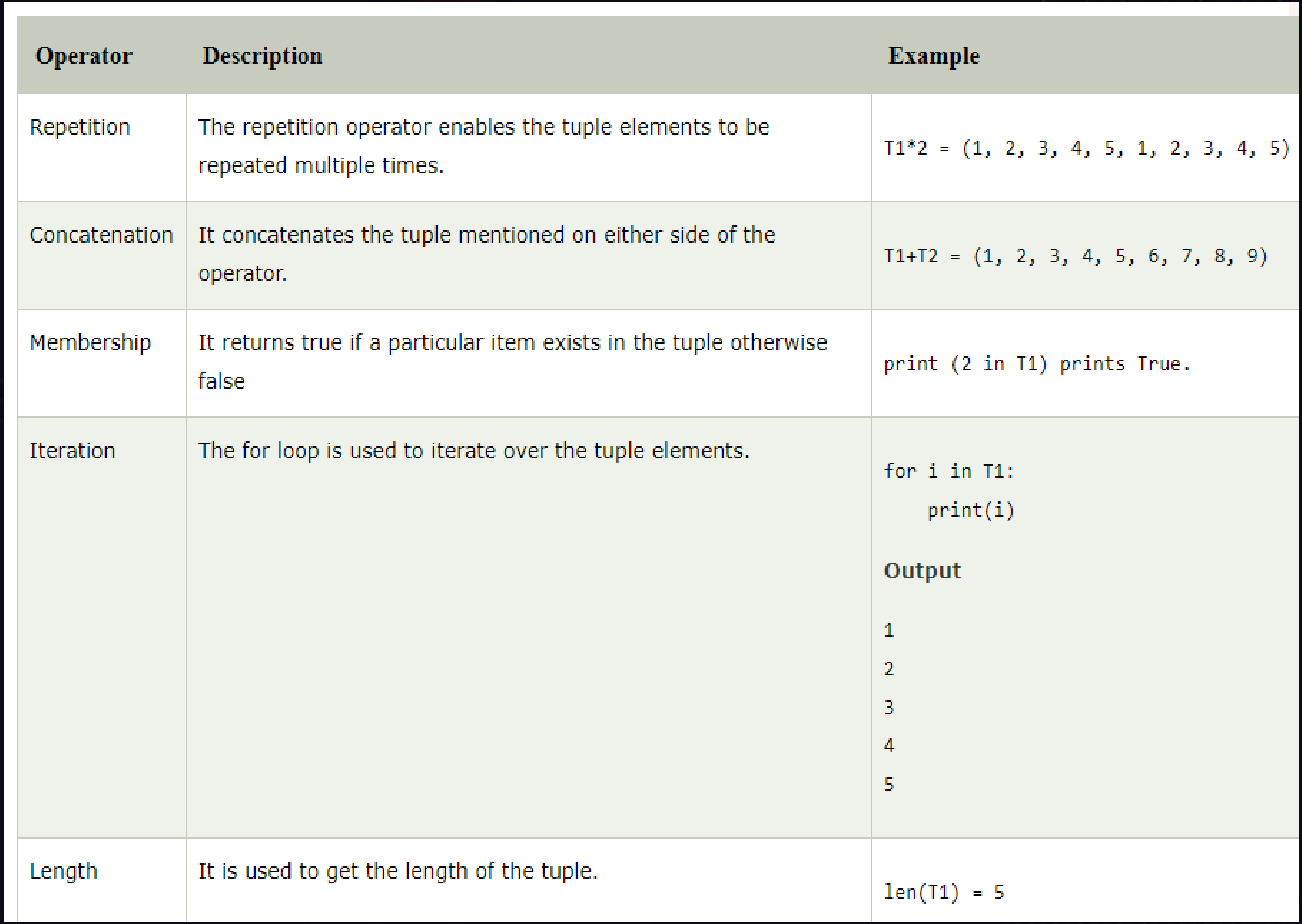
4	min(list)	It returns the minimum element of the list.	`L1 = [12,34,26,48,72]`
			`print(min(L1))`
			`12`

5	list(seq)	It converts any sequence to the list.	`str = "Johnson"`
			`s = list(str)`
			`print(type(s))`
			`<class list>`

Python Tuple

- Python Tuple is used to store the sequence of immutable Python objects. The tuple is similar to lists, only difference is list is changeable and tuple is not changeable.
- A tuple can be written as the collection of comma-separated (,) values enclosed with the small () brackets. The parentheses are optional but it is good practice to use.

Python Basic Tuple operations

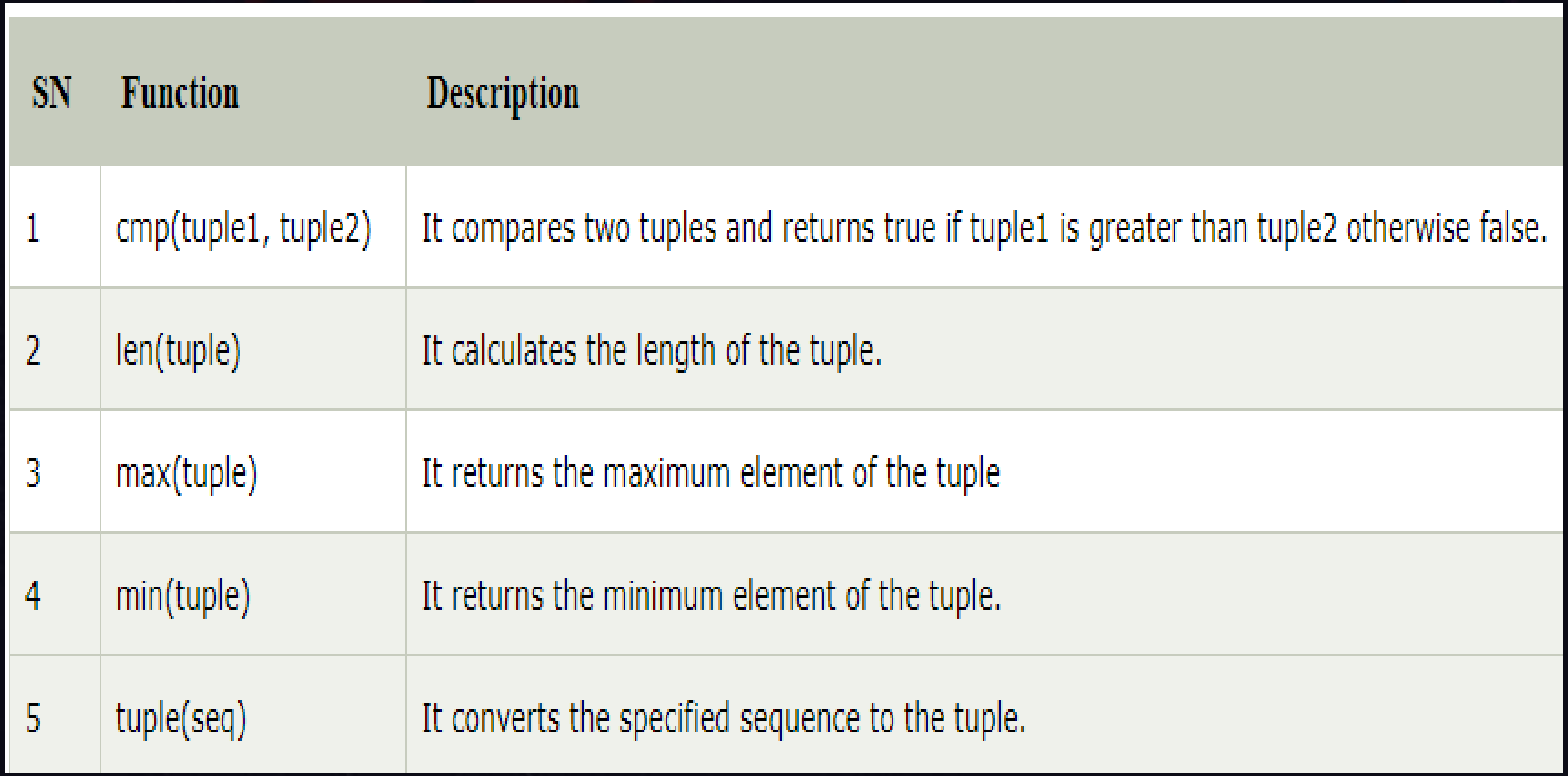


Operator	Description	Example
Repetition	The repetition operator enables the tuple elements to be repeated multiple times.	`T1*2 = (1, 2, 3, 4, 5, 1, 2, 3, 4, 5)`
Concatenation	It concatenates the tuple mentioned on either side of the operator.	`T1+T2 = (1, 2, 3, 4, 5, 6, 7, 8, 9)`
Membership	It returns true if a particular item exists in the tuple otherwise false	`print (2 in T1) prints True.`

Iteration	The for loop is used to iterate over the tuple elements.	`for i in T1:`
		`    print(i)`
		Output
		1
		2
		3
		4
		5

Length	It is used to get the length of the tuple.	`len(T1) = 5`

Python built in tuple functions



SN	Function	Description
1	cmp(tuple1, tuple2)	It compares two tuples and returns true if tuple1 is greater than tuple2 otherwise false.
2	len(tuple)	It calculates the length of the tuple.
3	max(tuple)	It returns the maximum element of the tuple
4	min(tuple)	It returns the minimum element of the tuple.
5	tuple(seq)	It converts the specified sequence to the tuple.

Where to use tuple?

Using tuple instead of list is used in the following scenario:

1. Using tuple instead of list gives us a clear idea that tuple data is constant and must not be changed.

2. Tuple can simulate a dictionary without keys. Consider the following nested structure, which can be used as a dictionary.

```
[(101, "John", 22), (102, "Mike", 28),  (103, "Dustin", 30)]
```

List vs Tuple

SN	List	Tuple
1	The literal syntax of list is shown by the [].	The literal syntax of the tuple is shown by the ().
2	The List is mutable.	The tuple is immutable.
3	The List has the a variable length.	The tuple has the fixed length.
4	The list provides more functionality than a tuple.	The tuple provides less functionality than the list.
5	The list is used in the scenario in which we need to store the simple collections with no constraints where the value of the items can be changed.	The tuple is used in the cases where we need to store the read-only collections i.e., the value of the items cannot be changed. It can be used as the key inside the dictionary.
6	The lists are less memory efficient than a tuple.	The tuples are more memory efficient because of its immutability.