|  |  |
| --- | --- |
| **SESSION** | **MARCH 2024** |
| **PROGRAM** | **BACHELOR OF COMPUTER APPLICATIONS (BCA)** |
| **SEMESTER** | **V** |
| **COURSE CODE & NAME** | **DCA3104 - PYTHON PROGRAMMING** |
|  |  |
|  |  |

**Set-I**

**1. Explain operators in python with suitable examples of each.**

**Ans 1.**

Operators in Python are special symbols that perform specific operations on one or more operands, and then return a result. They are the building blocks of Python programming, enabling developers to manipulate data and perform calculations. Here’s a concise guide to the different types of operators in Python, along with examples:

**1. Arithmetic Operators**

These operators perform basic arithmetic operations:

* **Addition (+)**: Adds two operands. Example: 5 + 3 results in 8.
* **Subtraction (-)**: Subtracts the second operand from the first. Example: 5 - 3 results in 2.

Its Half solved only

Buy Complete assignment from us

**Price – 190/ assignment**

**MUJ Manipal University Complete SolvedAssignments session FEB 2024**

buy cheap assignment help online from us easily

we are here to help you with the best and cheap help

**Contact No – 8791514139 (WhatsApp)**

**OR**

**Mail us-** [**bestassignment247@gmail.com**](mailto:bestassignment247@gmail.com)

**Our website -** [**www.assignmentsupport.in**](http://www.assignmentsupport.in)

**2. Explain about string slicing with examples. 10**

**Ans 2.**

String slicing in Python is a powerful feature that allows you to extract a portion of a string, using a concise syntax that specifies the start and end positions. This operation is vital in many programming tasks, such as data cleaning, parsing, and transformation. Here’s an in-depth look at string slicing, with examples to illustrate its utility.

**Understanding String Indexing**

Before delving into slicing, it's important to understand string indexing. In Python, strings are

which it allows programmers to perform complex tasks with minimal code.

**3. Differentiate between local and global variables with example.**

**Ans 3.**

In Python, as in many other programming languages, variables that you define in a program can be classified as either local or global. This classification affects how variables are accessed and modified within the program. Understanding the distinction between local and global variables is crucial for managing the scope of data within functions and across the codebase.

**Local Variables**

**Definition**: Local variables are those that are declared inside a function or a block of code and

**Set-II**

**4. What are exceptions in python? Explain the method to handle them with example.**

**Ans 4.**

Exceptions in Python are events that can modify the flow of control through a program. They are used to signal errors or abnormal conditions that occur during the execution of a program. Handling exceptions allows programmers to deal with these anomalies gracefully without causing the program to crash or behave unpredictably.

**Understanding Exceptions**

An exception in Python is a signal that an error or other unusual condition has occurred. When

**5. Differentiate between List, Tuple & Set with suitable examples of each.**

**Ans 5.**

In Python, lists, tuples, and sets are three fundamental data structures that each have unique characteristics and use cases. Understanding the differences between these types can help in choosing the appropriate structure based on the requirements of the application.

**List**

A **list** is a mutable, ordered sequence of items. Being mutable means that you can change, add, or

**6. With suitable example explain the concept of using else statement with loops related to while loop.**

**Ans 6.**

In Python, the else clause can be used with loops, including while loops. This feature is somewhat unique to Python and can enhance the control flow in programs by providing a block of code that runs only when the loop has completed normally, without being interrupted by a break statement. The use of else with a while loop is particularly useful when you want to execute some code after the loop finishes iterating, but only if the loop wasn't exited prematurely