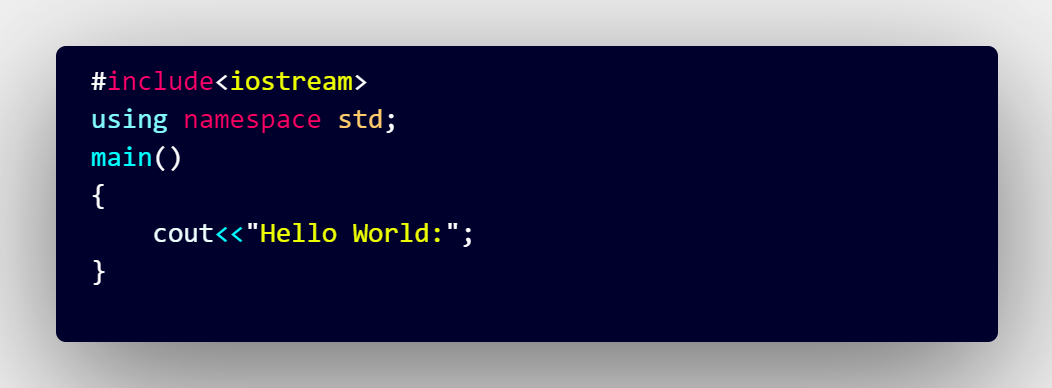
**Module: 1 (C++ Basic)**

**1.** **WAP to print “Hello World” using C++ :**

**Ans:**

OUTPUT:

Hello World:

**2.** **What is OOP? List OOP concepts:**

* OOP, or Object-Oriented Programming, is a programming paradigm that revolves around the concept of "objects," which can contain data (attributes) and behavior (methods). The primary goal of OOP is to organize code into reusable, self-contained units called objects.
* **Class and Objects**:
* **Class:** A blueprint or template for creating objects. It defines the properties (attributes) and behaviors (methods) that the objects will have.
* **Object:** An instance of a class, which represents a specific entity with its own set of attributes and behaviors.
* **Encapsulation:**
* The concept of bundling the data (attributes) and the methods (behaviors) that operate on that data within a single unit (class).
* Access to the data is controlled through methods, allowing for better control and security.
* **Inheritance:**
* A mechanism that allows a class (subclass) to inherit properties and behaviors (attributes and methods) from another class (superclass).
* It promotes code reusability and the creation of a hierarchy of classes.
* **Polymorphism:**
* The ability of a method to do different things based on the object it is acting upon or the input it receives.
* Enables methods with the same name to be used in different classes, exhibiting different behaviors.
* **Abstraction:**
* The process of hiding the complex implementation details and showing only the essential features of an object.
* It helps in reducing complexity and allows the programmer to focus on interactions at a higher level.
* **Association:**
* A relationship between two classes, representing how they are connected or interact with each other.
* Associations can be one-to-one, one-to-many, or many-to-many.
* **Composition:**
* A form of association where one class contains another class as a part.
* The contained class cannot exist without the container class.
* **Aggregation:**
* A special form of association where one class is composed of multiple instances of another class.
* The contained class can exist independently of the container class.
* **Interfaces:**
* A contract that defines a set of methods that a class must implement if it implements that interface.
* It allows for the implementation of multiple inheritances in languages that do not support it directly.
* **Overloading and Overriding:**
* Overloading: Defining multiple methods in the same class with the same name but different parameters (method signature).
* Overriding: Providing a specific implementation of a method in a subclass that is already defined in its superclass.

**3.**  **What is the difference between OOP and POP?**

* The main difference between Object-Oriented Programming (OOP) and Procedural Programming (POP) is in how they structure and organize code:
* OOP (Object-Oriented Programming)**:**
* Organizes code into objects, which encapsulate both data (attributes) and behavior (methods).
* Utilizes principles such as encapsulation, inheritance, and polymorphism.
* Focuses on modeling real-world entities as objects and their interactions.
* POP (Procedural Programming)
* Organizes code into procedures or functions that operate on data.
* Typically uses global variables for data, which can lead to issues with data integrity and maintenance.
* Lacks the built-in concepts of encapsulation, inheritance, and polymorphism.