**Misconception Check**

**Select 2 programming language and compare the advantages and disadvantages of their socket**

**programming.**

**C++ and JAVA**

**Advantages of C++**

* **C++** is an object-oriented programming language and includes classes, inheritance, polymorphism, data abstraction, and encapsulation.
* **C++** has a rich function library.
* **C++** allows exception handling and function overloading which is not possible in C.
* **C++** is a powerful, efficient and fast language.

**Disadvantages of C++**

* It has no security.
* Complex in a very large high-level program.
* Used for platform-specific applications commonly.
* For a particular operating system or platform, the library set has usually chosen that locks.
* When **C++** used for web applications complex and difficult to debug.
* **C++** can't support garbage collection.

**Advantages of Java**

* **Java** was designed to be easy to use, write, compile, debug, and learn from other programming languages
* Object-Oriented: Allows you to create modular programs and reusable code
* Platform-Independent: Ability to move easily from one computer system to another

**Disadvantages of Java**

* **Java** is Slow. **Java** is comparatively slower
* **Java** Takes More Memory Space
* Language Limited Latency Critical Tuning
* **Java's** Architecture Code is Inefficient
* No Back-up

**B EXPLORE : API Specifications list**

**1. Give all of socket and server socket methods with their descriptions**

* **socket.bind()** − This method binds the address (hostname, port number) to the socket.
* **socket.listen()** − This method basically listens to the connections made to the socket. It starts TCP listener. Backlog is an argument of this method which specifies the maximum number of queued connections. Its minimum value is 0 and maximum value is 5.
* **socket.accept()** − This will accept TCP client connection. The pair (conn, address) is the return value pair of this method. Here, conn is a new socket object used to send and receive data on the connection and address is the address bound to the socket. Before using this method, the socket.bind() and socket.listen() method must be used.
* **socket.connect(address)** − this method actively intimate server connection or in simple words this method connects the client to the server. The argument address represents the address of the server.
* **socket.recv(bufsize)** − As name implies, this method receives the TCP message from socket. The argument bufsize stands for buffer size and defines the maximum data this method can receive at any one time.
* **socket.send(bytes)** − This method is used to send data to the socket which is connected to the remote machine. The argument bytes will gives the number of bytes sent to the socket.
* **socket.recvfrom(data, address)** − This method receives data from the socket. Two pair (data, address) value is returned by this method. Data defines the received data and address specifies the address of socket sending the data.
* **socket.sendto(data, address)** − As name implies, this method is used to send data from the socket. Two pair (data, address) value is returned by this method. Data defines the number of bytes sent and address specifies the address of the remote machine.
* **socket.close()** − This method will close the socket.
* **socket.gethostname()** − This method will return the name of the host.
* **socket.sendall(data)** − This method sends all the data to the socket which is connected to a remote machine. It will carelessly transfers the data until an error occurs and if it happens then it uses socket.close() method to close the socket.