**Write 5 differences and similarities between C++ and Java in a tabular form.**

**Comparison Table: C++ vs. Java**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **C++** |  | **JAVA** |
| **1** | C++ is a **statically-typed language** with support for both high-level and low-level programming. |  | Java is a statically-typed, high-level, and **platform-independent language.** |
| **2** | C++ allows manual **memory management** using pointers, which can lead to memory leaks and segmentation faults. |  | Java has automatic **memory management** through garbage collection, reducing the risk of memory-related issues. |
| **3** | C++ code can be **platform-dependent** due to direct hardware interaction, requiring recompilation for different platforms. |  | Java is **platform-independent**, thanks to its "Write Once, Run Anywhere" (WORA) principle using the Java Virtual Machine (JVM). |
| **4** | C++ supports **multiple inheritance**, allowing a class to inherit from more than one base class. |  | Java supports **single inheritance** for classes but allows multiple inheritance through interfaces, promoting a more flexible form of composition. |
| **5** | C++ includes **pointers and references,** enabling direct memory manipulation and efficient low-level operations. |  | Java **does not have pointers** or direct memory manipulation, enhancing security but limiting low-level control. |

**Similarities C++ & JAVA**

|  |  |  |
| --- | --- | --- |
|  |  | **C++ & JAVA SIMILARITIES** |
| **1** |  | Both C++ and Java are **object-oriented languages**, emphasizing the use of objects and classes to model real-world entities and relationships. |
| **2** |  | Both languages support **exception handling** for robust error management and graceful program termination. |
| **3** |  | C++ provides the **Standard Template Library (STL)** for data structures and algorithms, while Java offers the **Java Standard Library** with similar functionality. |
| **4** |  | Both languages support **multithreading**, allowing concurrent execution of tasks, but the threading mechanisms and libraries differ. |
| **5** |  | C++ has a long history and a wide range of **libraries** and frameworks, but it may have a steeper learning curve. Java has a large and mature ecosystem with extensive community support and numerous libraries. |
| **6** |  | While both languages are primarily object-oriented, they also **support other programming paradigms**. C++ allows for procedural and generic programming in addition to OOP, and Java supports aspects of functional programming through features like lambdas and streams. |