|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Features of Java**   1. Simple 2. Object-Oriented 3. Platform independent 4. Secured 5. Robust 6. Architecture neutral 7. Portable 8. Dynamic 9. Interpreted 10. High Performance 11. Multithreaded 12. Distributed   **Simple**   |  | | --- | | According to Sun, Java language is simple because: | | syntax is based on C++ (so easier for programmers to learn it after C++). | | removed many confusing and/or rarely-used features e.g., explicit pointers, operator overloading etc. | | No need to remove unreferenced objects because there is Automatic Garbage Collection in java. |   **Object-oriented**   |  | | --- | | Object-oriented means we organize our software as a combination of different types of objects that incorporates both data and behaviour. | | Object-oriented programming(OOPs) is a methodology that simplify software development and maintenance by providing some rules. | | Basic concepts of OOPs are: | | 1. Inheritance 2. Polymorphism 3. Abstraction 4. Encapsulation |   **Platform Independent**   |  | | --- | | A platform is the hardware or software environment in which a program runs. There are two types of platforms software-based and hardware-based. Java provides software-based platform. The Java platform differs from most other platforms in the sense that it's a software-based platform that runs on top of other hardware-based platforms.It has two components:   1. Runtime Environment 2. API(Application Programming Interface) |  |  | | --- | | Java code can be run on multiple platforms e.g.Windows,Linux,Sun Solaris,Mac/OS etc. Java code is compiled by the compiler and converted into bytecode.This bytecode is a platform independent code because it can be run on multiple platforms i.e. Write Once and Run Anywhere(WORA). |   **Secured**   |  | | --- | | Java is secured because: | | * No explicit pointer * Programs run inside virtual machine sandbox. |  |  |  | | --- | --- | |  |  |  |  | | --- | | * **Classloader-** adds security by separating the package for the classes of the local file system from those that are imported from network sources. * **Bytecode Verifier-** checks the code fragments for illegal code that can violate access right to objects. * **Security Manager-** determines what resources a class can access such as reading and writing to the local disk. | | These security are provided by java language. Some security can also be provided by application developer through SSL,JAAS,cryptography etc. |   **Robust**   |  | | --- | | Robust simply means strong. Java uses strong memory management. There are lack of pointers that avoids security problem. There is automatic garbage collection in java. There is exception handling and type checking mechanism in java. All these points makes java robust. |   **Architecture-neutral**   |  | | --- | | There is no implementation dependent features e.g. size of primitive types is set. |   **Portable**   |  | | --- | | We may carry the java bytecode to any platform. |   **High-performance**   |  | | --- | | Java is faster than traditional interpretation since byte code is "close" to native code still somewhat slower than a compiled language (e.g., C++) |   **Distributed**   |  | | --- | | We can create distributed applications in java. RMI and EJB are used for creating distributed applications. We may access files by calling the methods from any machine on the internet. |   **Multi-threaded**  A thread is like a separate program, executing concurrently. We can write Java programs that deal with many tasks at once by defining multiple threads. The main advantage of multi-threading is that it shares the same memory. Threads are important for multi-media, Web applications etc. |