

# SE2052

## Worksheet 01

1. How does C compare to java in terms of readability?

Features	C	Java
Abstraction level	Low level	High level
Syntax structure	Procedural	Object-oriented
Memory Management	Manual	Automatic (Garbage collector)
Error handling	Manual error checking	Try-Catch blocks, built-in exception handling improves clarity and reduces hidden errors
Library support	basic	Extensive and consistent
Code clarity	Can be cluttered for complex tasks (e.g. string handling, I/O)	Cleaner for complex tasks due to standard libraries and conventions.
Consistency	Flexible, but inconsistent styles reduce readability across projects	Strict conventions (e.g. naming, JavaBeans) ensure uniform readability

Java is more readable to the structured and OOP concepts and C is hard to read because of large / complex programs

2. How does C compare to java in terms of writability?

Aspect	C	java
Syntax simplicity	Simple and minimal syntax. Program can be written with very little boilerplate.	More verbose to object-oriented structure. Although code simple task requires class and method definitions.
Memory Management	Manual (e.g. malloc, free), but increases programming complexity and error chances.	Automatic garbage collection reduces effort and simplifies code writing.

Error Handling	Manual error checking (e.g. return codes) adds complexity and code volume.	Built-in exception handling (try-catch) streamlines error management.
Library Support	Limited standard libraries, often requires external libraries or custom code, slowing development.	Extensive standard libraries (e.g. Java API) reduce need for custom code, speeding up development.

Java is much more writable than C for most application development tasks. However, C may be writable for low-level system programming due to its direct access to hardware and memory.

### 3. How does C compare to java in terms of reliability?

Features	C	java
Memory management	Manual (use malloc, free); prone to memory leaks and pointer bugs	Automatic (garbage collector); safe and more reliable
Error handling	Basic (return codes); easy to miss error	Structured (Try-Catch); better error reporting and handling
Type safety	Weak (allows pointer arithmetic, implicit conversions)	Strong (strict type checking at compile-time and runtime)
Run time checks	Minimum runtime checks (e.g. array bounds not enforced)	Includes runtime checks (e.g. array bounds, null references)
Array bounds checking	No bounds checking (e.g., arr[100] compiles even if array is size 10)	Automatic bounds checking – throws <code>ArrayIndexOutOfBoundsException</code>

Java is more reliable with garbage collector, type safety and runtime checks. But C is less safe due to manual memory handling and fewer safety checks.

### 4. For each of the 9 characteristics, analyze how C and java handle them differently. Provide examples where applicable?

Characteristics	C	java
Simplicity	Simple language low-level and small set of keywords e.g. <code>printf("hello");</code>	More complex syntax and larger standard libraries

		e.g. System.out.println("she");
Orthogonality	Low orthogonality. Operations valid in one context might not work in another. (can't return arrays) e.g. int a[10]; Int *p=a;	Higher orthogonality. Has independent constructs like classes and interfaces (can return arrays) e.g. int[] a= new int [10]
Data type	Basic primitive and structs for complex data.  e.g. struct Point{ Int x, y;;	Rich primitives and objects. Has class-based. e.g. class Point { Int x, y };
Syntax design	Prone to errors e.g. int *p;	Object-oriented and strict rules e.g. int add( int a, int b){ Return a+b;}
Support for abstraction	relies on structs and functions for manual abstraction	Classes, interfaces, and packages for robust abstraction
Expressivity	High (for low-level)	High (for high level) with libraries
Type checking	Weak and allows unsafe conversions.	Strong and enforces strict type safety.
Exception handling	Need manual assistance	Already built-in
Restricted aliasing	Aliasing is common and unrestrictions due to pointers  e.g. void update(int *a, int *b) { *a += 10; *b += 5; }	Has restrictions and safe  E.g. void update(int[] a, int[] b) {  a[0] += 10;  b[0] += 5;  }

5. What type of language would you use for system-level programming and why?

C is better compared to java due to low level access, performance and minimal runtime

6. What type of language would you use for application-level development and why?

Java is better than C because of use cases, memory management and error handling