2.

Java and Scala are both powerful languages, but they have some key differences:

Platform:

- **Java:** Runs on the Java Virtual Machine (JVM), which means it can run on any platform with a JVM installed. This gives Java great portability.
- **Scala:** Primarily runs on the JVM, but can also target the .NET Common Language Runtime (CLR) with some additional tooling.

Programming Paradigm:

- **Java:** Primarily object-oriented, meaning programs are built around objects that contain data and methods.
- **Scala:** Object-oriented and functional, allowing programmers to use both paradigms in the same project. Functional programming emphasizes functions and immutable data.

Type System:

- **Java:** Statically typed, meaning variable types must be declared before they are used. This can help catch errors early in the development process.
- **Scala:** Statically typed with type inference, which means the compiler can often infer the type of a variable from its context. This can make code more concise.

Verbosity:

- **Java:** Generally considered more verbose, requiring more lines of code to achieve the same results as Scala.
- **Scala:** More concise due to features like type inference and functional programming constructs.

Build System:

- **Java:** Traditionally uses Makefiles or Apache Ant for building projects.
- **Scala:** Uses the Scala Build Tool (sbt) for building and managing dependencies.

Use Cases:

- **Java:** Widely used in enterprise applications, Android development, and big data technologies due to its maturity, large developer community, and vast libraries.
- **Scala:** Popular for big data, machine learning, and functional programming tasks due to its conciseness, powerful type system, and suitability for parallel processing.

Here's a table summarizing the key points:

Feature	Java	Scala
Platform	JVM	JVM, CLR (.NET)

Programming Paradigm	Object-oriented	Object-oriented, Functional
Type System	Statically typed	Statically typed with type inference
Verbosity	More verbose	More concise
Build System	Makefiles, Apache Ant	sbt (Scala Build Tool)
Use Cases	Enterprise applications, Android dev	Big data, machine learning, functional programming