**Statically Typed Language**

Statically typed languages are those in which data types are explicitly declared for variables at compile-time. This means that the data type of each variable must be known and specified before the program is executed. The compiler checks for type correctness during compilation, and any type errors are caught early in the development process. C, C++, and Java are examples of statically typed languages.

**Dynamically Typed Language**

Dynamically typed languages are those in which data types are determined at runtime, i.e., t he data type of a variable is resolved during program execution. Unlike statically typed languages, we don't need to declare the data type of variables explicitly. Python, JavaScript, and Ruby are examples of dynamically typed languages.

**Strongly Typed Language**

Strongly typed languages are those in which type-checking is strictly enforced, and implicit type conversion is limited. In a strongly typed language, we cannot perform operations that involve incompatible data types without explicit type conversion. This ensures type safety and reduces the risk of unintended errors. Java, C++, and Python are examples of strongly typed languages.

**Loosely Typed Language**

Loosely typed languages are those in which type-checking is more lenient, and automatic type conversion (coercion) between different data types is allowed. Variables can change their data type on the fly, and the language will implicitly perform type conversions when required. PHP is an example of a loosely typed language.

Java falls into the category of a **Statically Typed Language** and a **Strongly Typed Language.** Also Java can be considered as Dynamically typed Language.

**Case Sensitive**

Case sensitivity refers to the distinction made between uppercase and lowercase letters in programming languages. In a case-sensitive language, variables, function names, keywords, and identifiers must be spelled with the exact casing (uppercase or lowercase) as they are declared. This means that "Variable" and "variable" would be considered two different identifiers in a case-sensitive language.

**Case Insensitive**

Case insensitivity means that uppercase and lowercase letters are treated as the same in programming languages. In a case-insensitive language, identifiers are not distinguished based on casing, so "Variable" and "variable" would be considered the same identifier.

**Case Sensitive-Insensitive (Mixed Case Sensitivity)**

Some programming languages are case-sensitive for some parts and case-insensitive for others. For example, variable names might be case-sensitive, while function names are case- insensitive.

Java is a Case-Sensitive language. It makes a clear distinction between uppercase and lowercase letters in identifiers, keywords, and other parts of the code.