# Java Keywords and Their Uses

## 1. Data Type Keywords

* byte: Defines an 8-bit integer variable (range: -128 to 127). Used for saving memory.
* int: Defines a 32-bit integer variable. Commonly used for whole numbers.
* boolean: Defines a variable that can store only true or false.
* short: Defines a 16-bit integer variable. Smaller than int.
* long: Defines a 64-bit integer variable. Used for large integer values.
* float: Defines a 32-bit floating-point variable (for decimals).
* double: Defines a 64-bit floating-point variable (more precise than float).
* char: Defines a single 16-bit Unicode character.

## 2. Access Control Keywords

* public: Specifies that a class, method, or variable is accessible from anywhere.
* private: Restricts access to within the same class.
* protected: Allows access within the same package and subclasses.

## 3. Class, Method, and Object Keywords

* class: Defines a class (a blueprint for objects).
* interface: Defines an interface (a collection of abstract methods).
* extends: Indicates that a class inherits from a superclass.
* implements: Used by a class to implement an interface.
* new: Creates new objects (instances) of a class.
* this: Refers to the current object (useful for differentiating instance and local variables).
* super: Refers to the parent class (used to call superclass constructors or methods).
* transient: Excludes a variable from serialization.
* abstract: Defines an abstract class or abstract method (must be overridden).
* final: Used to make a variable constant, prevent method overriding, or inheritance.
* static: Defines class-level variables or methods shared by all instances.
* synchronized: Used to control thread access to a block/method (for thread safety).
* volatile: Marks a variable as 'may be changed unexpectedly' — ensures visibility across threads.
* native: Declares a method implemented in another language (like C).

## 4. Flow Control Keywords

* if: Executes a block of code if the condition is true.
* else: Executes a block if the if condition is false.
* switch: Selects one of many code blocks to execute.
* case: Defines a branch in a switch statement.
* default: Defines the block that runs if no case matches in a switch.
* for: Defines a loop that runs a fixed number of times.
* while: Defines a loop that runs while a condition is true.
* do: Used with while for a loop that runs at least once.
* break: Exits a loop or switch immediately.
* continue: Skips the current iteration and continues with the next one.
* return: Exits from a method and optionally returns a value.
* yield: Returns a value from a switch expression.

## 5. Exception Handling Keywords

* try: Defines a block of code to test for exceptions.
* catch: Defines a block of code to handle exceptions.
* finally: Defines a block that always executes after try (used for cleanup).
* throw: Used to throw an exception manually.
* throws: Declares the exceptions that a method can throw.

## 6. Package and Import Keywords

* package: Defines a namespace (collection of related classes).
* import: Imports classes or entire packages for use in the file.

## 7. Object Reference and Null Keywords

* null: Represents the absence of any object reference.

## 8. Logical and Conditional Keywords

* true: Boolean literal value representing truth.
* false: Boolean literal value representing falsehood.
* instanceof: Tests whether an object is an instance of a specific class or subclass.

## 9. Miscellaneous and Advanced Keywords

* assert: Used for debugging; tests assumptions about the program.
* enum: Defines a set of named constants (enumeration).
* const: Reserved but not used (use final instead).
* var: Allows local variable type inference (the compiler infers the type).
* record: Defines an immutable data class (used for holding data).
* sealed: Restricts which classes can extend or implement a class/interface.
* permits: Used with sealed to specify allowed subclasses.
* goto: Reserved but not used (for backward compatibility with C).