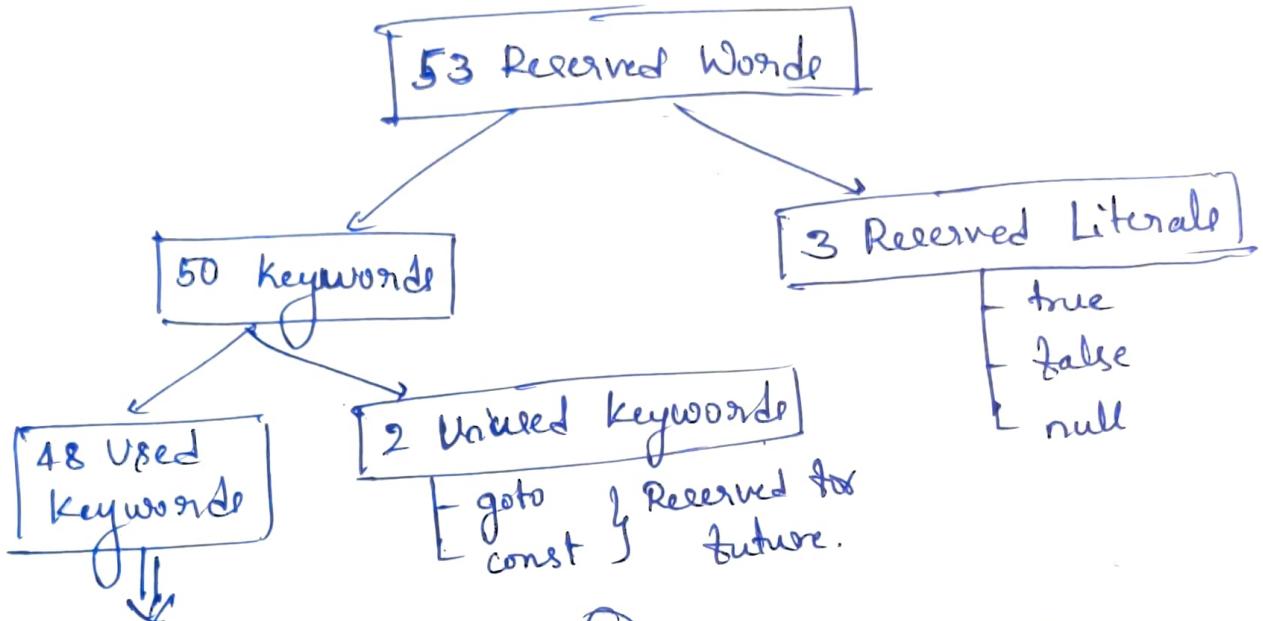


## Reserved Words:

In Java, some identifiers are reserved to represent specific functionality or meaning; such reserved identifiers are called reserved words.



### 1) Access Modifiers (4) → ③

\* public \* private \* protected \* default  
 (Not actually a keyword for default access, but a modifier keyword in default methods of interfaces).

### 2) Class, Object & Interface Declarations (8) → ⑦

\* class \* interface \* enum \* abstract \* final  
 \* implements \* extends \* record → (added in Java 14)  
 as a preview, permanent in Java 16 → still a keyword

### 3) Method, Variable & Flow Modifiers (11) → ⑥

\* static \* volatile \* synchronized \* transient  
 \* native \* strictfp \* default (overlaps with Access modifiers)  
 \* sealed (Java 17) \* non-sealed (Java 17)  
 \* var (not a keyword) in Java 10+, but a reserved type  
 name → excluded in count of 50 keywords  
 \* permit (Java 17)

## ④ Flow Control (12) → 1.1 ~~1.1~~

- \* if \* else \* switch & case \* default
- \* for \* while \* do \* break \* continue
- \* return \* (yield) (added in Java 14 for switch expressions).

## ⑤ Exception Handling (6)

- \* try \* catch \* finally \* throw \* throws
- \* assert.

## ⑥ Primitive Data Types: (8)

- \* byte \* short \* int \* long \* float
- \* double \* char \* boolean.

## ⑦. Miscellaneous (7)

- \* package \* import
- \* super \* this
- \* instanceof
- \* new \* void

### Unused Keywords

- goto → Caused several problems in older programming languages, so banned in Java.
- Const → the final keyword is used instead.

### Enum

- \* The enum keyword (introduced in Java 5) is used to define a group of named constants. Enums are a special type that can hold a fixed set of values, making code safer & more readable.

Eg: enum Day {  
MONDAY, TUESDAY, WEDNESDAY, THURSDAY,  
FRIDAY, SATURDAY, SUNDAY};

```

public class EnumExample {
    public void m(){
        Day today = Day.Monday;
        System.out.println("Today is: " + today);
        System.out.println("All days of the week:");
    }
}

```

```

for (Day d: Day.values()){
    System.out.println(d);
}

```

```

}
}

Day[] days = Day.values();
for (int i=0; i<days.length; i++){
    System.out.println(days[i]);
}

```

### Key Facts about Java Reserved Words:

#### ① All lowercase by Design

↳ Every Java reserved word is written in lower case letters.

#### ② Keywords added over time

↳ Java has introduced new keywords in specific versions:

- strictfp — Java 1.2
- assert — Java 1.4
- enum — Java 1.5

#### ③ No keyword Deletions

↳ Java never removes keywords. Memory cleanup is handled by the Garbage Collector, so there's no need for a keyword like delete.

#### ④ Case Sensitive Matters:

↳ Java keywords are case-sensitive & must match exactly:

- \* instanceof → not instanceOf
- \* strictfp → not strictFp
- \* const → not constant

#### ⑤ Exact spelling is required

↳ Misspellings make the word a normal identifier, not a keyword:

- \* synchronized → not synchronize
- \* extends → not extends
- \* implements → not implement
- \* import → not Imports
- \* int → not Int