

# Types

# Introduction

- ❖ The types in Java can be either value types or reference types.
- ❖ When dealing with the native types (e.g. int, float, etc) we are dealing with value types. In all other cases we are dealing with reference types.

# The Native Data Types

- ❖ The following table summarize the native data types in Java:

Type	Values	Number Of Bits	Default Value
boolean	true or false	1	false
byte	integers	8	0
char	unicode values	16	\x0000

# The Native Data Types

Type	Values	Number Of Bits	Default Value
short	integers	16	0
int	integers	32	0
long	integers	64	0
float	real numbers	32	0.0
double	real numbers	64	0.0

# Declaring A Variable

- ❖ Declaring a variable in Java is done by specifying its type followed with its name:

```
type variableName;
```

- ❖ The following are examples for various possible declarations:

```
int number;
```

```
int num, sum, total;
```

# Declaring A Variable

- ❖ We can declare more than one variable in the same line and we can also initialize it with a value.
- ❖ The following is a small example

```
int numOfStudents, numOfTeachers=22;
```

# The Identifiers

- ❖ The identifiers are the names we give to variables, classes and methods.
- ❖ The identifier can start with the dollar sign (\$), with a unicode letter or the underscore sign ('\_').
- ❖ The identifiers are case sensitive.

# The Keywords

- ❖ The keywords in Java are predefined words that already have a special meaning for the compiler.



# Expressions & Operators

## ❖ Mathematic operators:

`+, -, *, \, %, ++, --, <<, >>, >>>, &, |, ~`

## ❖ Logical operators:

`!, &&, ||, ^`

## ❖ Operators that compare between expressions:

`<, <=, ==, !=, >, >=`

# Simple & Compound statements

- ❖ Java allows writing simple statements as well as compound statements (also known as blocks).
- ❖ Wherever we can place a simple statement, a compound statement can be placed as well.

# Assignment Statements

- ❖ The assignment statement in Java uses the '=' operator, and it is the same as in many other programming languages. The name of the variable to which we want to assign a value should be placed on the left, and the value should be placed on the right.

# Assignment Statements

- ❖ Java supports short assignment operators, such as the `+=`, `-=`, `*=`, and `/=`.

# Local Variable Scope

- ❖ Local variables are defined inside a method\block.
- ❖ Local variables are created when the method\block is executed and destroyed when the method\block ends.
- ❖ Local variables must be initialized before they are used.  
If a local variable isn't initialized a compile time error occurs.

# Binary, Octal & Decimal

- ❖ We can easily express integer numbers using the binary, the octal and hexadecimal numbering systems.

```
int numA = 0b10011101; //binary
int numB = 03425; //octal
int numC = 0xE12F; //hexadecimal
```



# Underscores in Numeric Literals

- ❖ We can improve the readability of our code by adding underscores in between the digits.

```
double num = 1_424_234.532;
```

