Java Full Stack

Java is a platform independent & object oriented programming language.

Softwares required

* JDK 8
* Eclipse IDE for Enterprise Java & Web developers

Datatypes in Java

Datatypes are set of keywords to represent the type of data a variable can store.

There are two types in datatypes.

1. Primitive types – byte, short, int, long, double, float, boolean, char
2. Derived types – classes, arrays, interface

Type Conversion: Converting from one type to another, since there are two types of conversion for each type of datatypes.

1. Primitive types
   1. Auto-Widening
   2. Explicit-Narrowing
2. Derived types
   1. Auto-Upcasting
   2. Explicit-Downcasting

Type Conversion

**package** com;

**public** **class** PrimitiveTypeConversion {

**public** **static** **void** main(String[] args) {

**byte** a = 10;

**int** b = 20;

**byte** c = (**byte**)b; // Explicit Narrowing

**int** d = a; // Auto-Widening

System.***out***.println("c = "+c);

System.***out***.println("d = "+d);

**int** e = 130;

**byte** f = (**byte**)e;

System.***out***.println("f = "+f);

}

}

Inheritance: Process of acquiring the properties & behaviours of an object from another object

Types of inheritance

1. Single level
2. Multi level
3. Hierarchical
4. Multiple (not supported through class, but possible through interface)

Person.java

**package** com;

**public** **class** Person {

String name;

**void** updateName() {

System.***out***.println("updateName() inside Person");

}

}

Student.java

**package** com;

**public** **class** Student **extends** Person {

String usn;

String grade;

**void** updateGrade() {

System.***out***.println("updateGrade() in Student");

}

}

Employee.java

**package** com;

**public** **class** Employee **extends** Person {

**int** id;

**double** salary;

**void** updateSalary() {

System.***out***.println("updateSalary() inside Employee");

}

}

TestInheritance.java

**package** com;

**public** **class** TestInheritance {

**public** **static** **void** main(String[] args) {

Person p1 = **new** Person();

p1.name = "Alex";

p1.updateName();

System.***out***.println("---------------------");

Employee e1 = **new** Employee();

e1.id = 100;

e1.name = "Bruce";

e1.salary = 35000;

e1.updateName();

e1.updateSalary();

System.***out***.println("-------------------");

Student s1 = **new** Student();

s1.name = "Charles";

s1.usn = "1ABC001";

s1.grade = "A+";

s1.updateName();

s1.updateGrade();

}

}

Note: Above program is to show that a subclass object can access super class members.

Access Modifiers

There are 4 access modifiers in java

1. Private: within the class
2. Package scope: within the package
3. Protected: within package & outside package but only to the subclass
4. Public: everywhere