

# Object Oriented Programming with Java

---

DANIEL DANSO ESSEL

DEPARTMENT OF ICT EDUCATION

# Objectives

---

At the end of this lesson, students should be able to:

---

define the term programming

---

explain the history of Java

---

discuss the Java Buzzwords

---

explain the OOP principles

---

differentiate Java Language Specifications

---

explain the concept of JVM

---

download, install & configure Java & NetBeans

---

explain the various parts of a Java program

---

write their first Java program

---

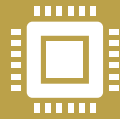
explain the types of errors in Java

---

# Introduction



The term programming means to create (or develop) software, which is also called a program.



In basic terms, software contains the instructions that tell a computer (or a computerized device) what to do.



Give examples of devices that use software



Mention some programming languages you know of

# Brief History of Java



Java was developed by a team led by James Gosling at Sun Microsystems and was originally called Oak



Sun Microsystems was purchased by Oracle in 2010



Oak was designed in 1991 for use in embedded chips in consumer electronic appliances.



In 1995, it was renamed Java and was redesigned for developing Web applications



Java became very popular because it was marketed as “write once, run anywhere”



Currently Java is one of the most popular programming languages.

# Types of Java Programs

## Applications

- An application is a program that runs on your computer, under the operating system of that computer

## Applets

- An applet is an “application” designed to be transmitted over the Internet and executed by a Java-compatible Web browser
- In other words, an applet is a program that can react to user input and dynamically change - not just run the same animation or sound over and over.

# Java Buzzwords

---



SIMPLE



OBJECT-ORIENTED



ROBUST



MULTI-THREADED



ARCHITECTURE-  
NEUTRAL



INTERPRETED AND  
HIGH  
PERFORMANCE



DISTRIBUTED



DYNAMIC

# Object-Oriented Programming Principles



Abstraction



Polymorphism



Inheritance

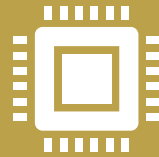


Encapsulation

# Java Language Specification



The Java language specification is a technical definition of the Java programming language's syntax and semantics.



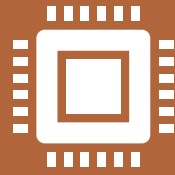
The Application Program Interface (API), also known as library, contains predefined classes and interfaces for developing Java programs.



The API is growing everyday.



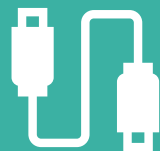
# Editions of Java



Java Standard Edition (Java SE) to develop client-side applications. The applications can run standalone or as applets running from a Web browser.



Java Enterprise Edition (Java EE) to develop server-side applications, such as Java servlets, Java Server Pages (JSP), and Java Server Faces (JSF).

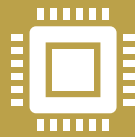


Java Micro Edition (Java ME) to develop applications for mobile devices, such as cell phones.

# Java Development Kit (JDK)



The JDK consists of a set of separate programs, each invoked from a command line, for developing and testing Java programs.

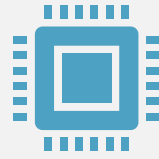


Instead of using the JDK, you can use a Java development tool (e.g., NetBeans, Eclipse, and TextPad): software that provides an integrated development environment (IDE) for developing Java programs quickly



What is the latest version of JDK?

# Java Virtual Machine (JVM)



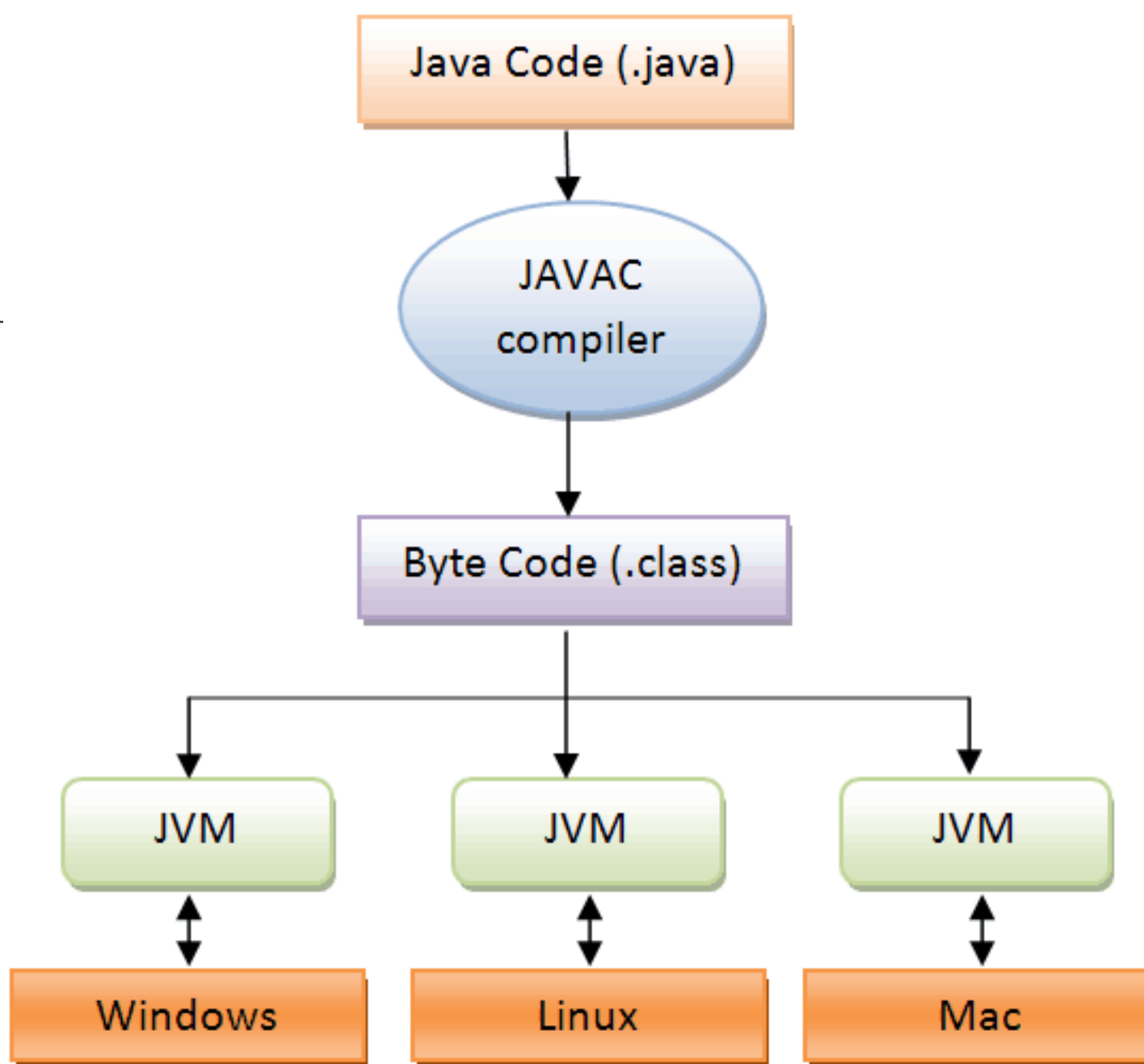
Java Runtime Environment (JRE)



Java is platform independent.



The reason is Java Virtual Machine.



# Downloading & Installing the NetBeans IDE



Go to the NetBeans download page at <https://netbeans.org/downloads>



Select the Java SE and click on download.



Now, run the setup to install (make sure you have already installed the JDK)



We're going to use NetBeans IDE to write our code.

NetBeans IDE 8.1

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help Search (Ctrl+I)

Start Page x

Files Services Projects Navigator

**NetBeans IDE** Learn & Discover **My NetBeans** What's New Show On Startup


## My NetBeans

**Recent Projects** [Install Plugins](#) [Activate Features](#)

<no recent project>

Add support for other languages and technologies by installing plugins from the NetBeans Update Center.

NetBeans turns on functionality as you use it. Start creating and opening projects and the IDE will just activate the features you need, making your experience quicker and cleaner. Alternatively, you can activate features manually.

ORACLE 

Output

# Creating Your First Program

### New Project

**Steps**

1. Choose Project
2. ...

### Choose Project

Filter:

Categories:

- Gradle
- Java
- JavaFX
- Maven
- Android
- Groovy
- NetBeans Modules
- Samples

Projects:

- Java Application
- Java Class Library
- Java Project with Existing Sources
- Java Free-Form Project

Description:

**Creates a new Java SE application** in a standard IDE project. You can also generate a main class in the project. Standard projects use **an IDE-generated Ant build script** to build, run, and debug your project.

< Back   Next >   Finish   Cancel   Help



New Java Application

**Steps**

1. Choose Project
2. **Name and Location**

**Name and Location**

Project Name:

Project Location:

Project Folder:

Use Dedicated Folder for Storing Libraries

Libraries Folder:






Different users and projects can share the same compilation libraries (see Help for details).

Create Main Class

< Back   Next >   **Finish**   Cancel   Help

```
FirstProject.java
Source History
1  /*
2   * To change this license header, choose License Headers in Project Properties.
3   * To change this template file, choose Tools | Templates
4   * and open the template in the editor.
5   */
6
7  package firstproject;
8
9  /**
10   *
11   * @author Ken
12   */
13  public class FirstProject {
14
15     /**
16      * @param args the command line arguments
17      */
18     public static void main(String[] args) {
19         // TODO code application logic here
20     }
21
22 }
23
```

Run

	Run Project (FirstProject)	F6
	Test Project (FirstProject)	Alt+ F6
	Build Project (FirstProject)	F11
	Clean and Build Project (FirstProject)	Shift+ F11
	Set Project Configuration	▶
	Set Main Project	▶
	Generate Javadoc (FirstProject)	
	Run File	Shift+ F6
	Test File	Ctrl+ F6
	Compile File	F9
	Check File	Alt+ F9
	Validate File	Alt+ Shift+ F9
	Repeat Build/Run	Ctrl+ F11
	Stop Build/Run	

# The Structure of Java Code

---

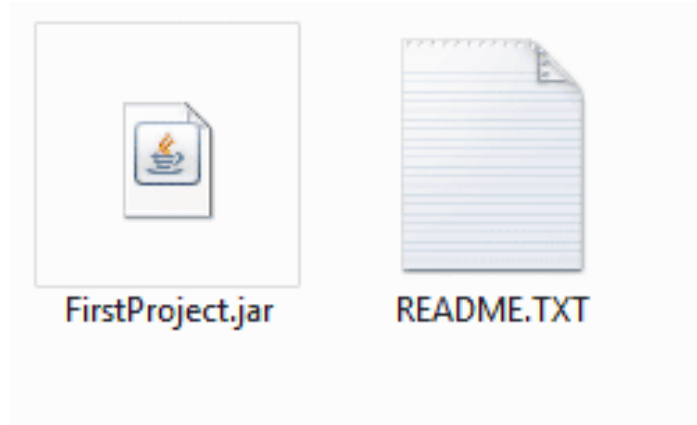
- ❖ Package name
- ❖ Class name
- ❖ Main method

```
package firstproject;  
  
public class FirstProject {  
  
    public static void main (String[ ] args) {  
        System.out.println( "My First Project" );  
    }  
}
```

# Sharing your Java programs

---

- ❖ You can send your programs to other people so that they can run them.
- ❖ To do this, you need to create a JAR file (Java Archive).
- ❖ From the Run menu at the top, select Clean and Build Main Project.
- ❖ NetBeans saves your work and then creates all the necessary files.
- ❖ It will create a folder called **dist** and place all the files in there.



# Programming Style



Spacing (legibility)



Indenting (consistency)



Commenting (clarity,  
maintainability)



Blocking (legibility)

Block style

End of line style

# Programming Errors



Syntax Errors



Runtime Errors



Logic Errors

# Assignment

---

- ❖ Download and Install JDK and NetBeans
- ❖ Write a Java program that can display the days of the week.
  - ❖ (Use any technique of your choice)
- ❖ Explain any 8 Java buzzwords in your own words.
  - ❖ (Handwritten: To be submitted to class next week)