ACADGILD

LEARN. DO. EARN

JAVA

Basics

Introduction to Java

Session 1

Agenda

ACADGILD

Sl. No.	Agenda Topics
1	Java History
2	Java Version History
3	JDK Editions
4	What is Java Technology?
5	Java Feature: Simple
6	Java Feature: Object Oriented
7	Java Feature: Platform Independence
8	Java Feature: Portable and High Performance
9	Java Feature: Secure
10	Java Feature: Robust
11	Java Feature: Multithreaded
12	Java Feature: Architecture Neutral
13	Structured Vs Object Oriented Approach

Sl. No.	Agenda Topics
14	Need for Object Oriented Approach
15	Introduction to OOPs
16	Concept of Class and Object
17	Setting Up of Environment
18	Writing First Java Program
19	Keywords
20	Data Types and Variables
21	Different Data Types
22	Type Casting
23	Operators in Java
24	User Input in Java
25	Introduction to JDK, JVM and JRE
26	Short Assignments

Java History

- **James Gosling, Mike Sheridan, and Patrick Naughton** initiated the Java language project in June 1991. The small team of sun engineers was called Green Team.

- Originally designed for small, embedded systems in electronic appliances like set-top boxes. It was called Oak and was developed as a part of the Green project.

- In 1995, Oak was renamed as "Java" because it was already a trademark by Oak Technologies.

- Originally developed by James Gosling at Sun Microsystems (which is now a subsidiary of Oracle Corporation) and released in 1995.

James Gosling

Java Version History

ACADGILD

- There are many Java versions that have been released. Current stable release of Java is **Java SE 8.**

  - JDK Alpha and Beta (1995)
  - JDK 1.0 (23rd Jan, 1996)
  - JDK 1.1 (19th Feb, 1997)
  - J2SE 1.2 (8th Dec, 1998)
  - J2SE 1.3 (8th May, 2000)
  - J2SE 1.4 (6th Feb, 2002)
  - J2SE 5.0 (30th Sep, 2004)
  - Java SE 6 (11th Dec, 2006)
  - Java SE 7 (28th July, 2011)
  - Java SE 8 (18th March, 2014)

JDK Editions

- Java Standard Edition (J2SE)

  J2SE can be used to develop client-side standalone applications or applets.

- Java Enterprise Edition (J2EE)

  J2EE can be used to develop server-side applications such as Java servlets and Java ServerPages.

- Java Micro Edition (J2ME)

  J2ME can be used to develop applications for mobile devices such as cell phones.

What is Java Technology?

ACADGILD

- Java technology is both a programming language and a platform.

- The Java programming language is a high-level language that can be characterized by following buzzwords:

  - Simple
  - Object-Oriented
  - Platform independent
  - Portable
  - High Performance
  - Secured
  - Robust
  - Multithreaded
  - Architecture Neutral

Java Feature: Simple

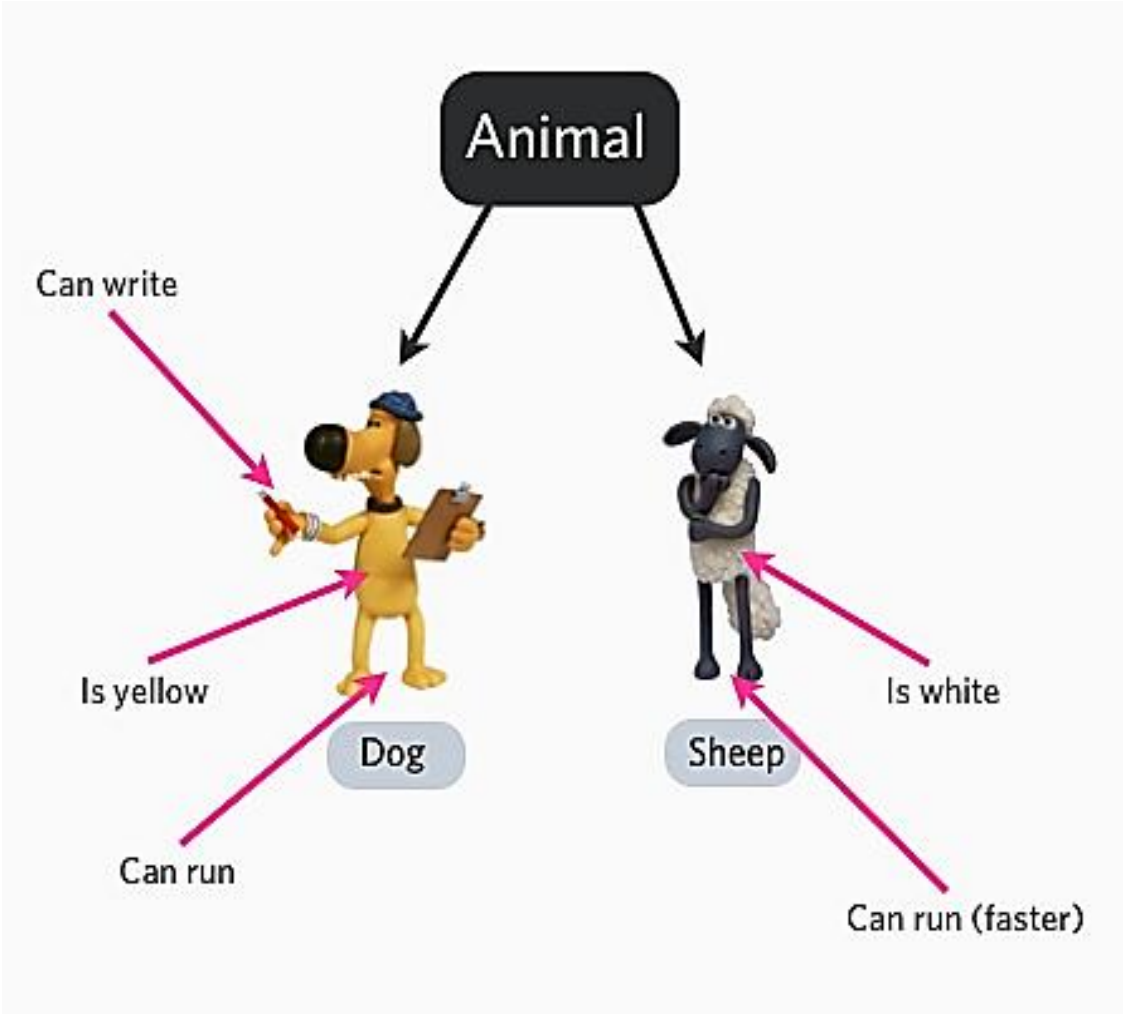
According to Sun, Java language is simple because:

- Syntax is based on C++ (so easier for programmers to learn Java after C++).
- Removed many confusing and/or rarely-used features, for example; explicit pointers, operator overloading etc.
- No need to remove unreferenced objects because there is Automatic Garbage Collection in Java.

Java Feature: Object Oriented

ACADGILD

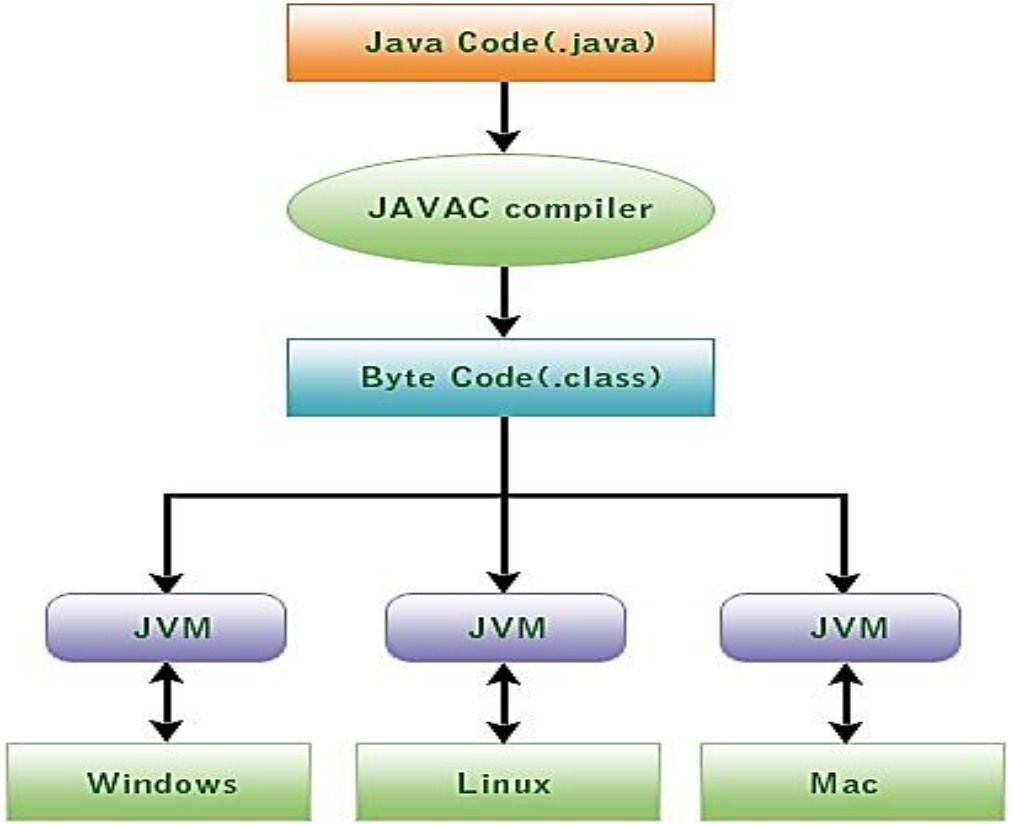
- Object-Oriented Programming Language (OOPs) is the methodology which provides software development and maintenance by using object state, behavior, and properties.



Java Feature: Platform Independence

- **Java Language is platform-independent** due to its hardware and software environment. Java code can run on multiple platforms e.g. Windows, Linux, Sun Solaris, Mac/Os etc.

- **Java code is compiled by the compiler and converted into byte code**. This byte code is a platform independent code because it can be run on multiple platforms i.e. Write Once and Run Anywhere (WORA).



Java Feature: Portable and High Performance

- **Portable:** We may carry the Java bytecode to any platform.

- **High performance:** For all but the simplest or most infrequently used applications, performance is always a consideration for most applications, including graphics-intensive ones such as the ones that are commonly found on the world wide web, the performance of Java is more than adequate.

Java Feature: Secure

- Java apps are used in distributed environments too. Thus, lot of emphasis is on security.

- The Java language is secure in the sense that it is very difficult to write incorrect code for viruses that can corrupt/steal your data, or harm hardware such as hard disks.

- There are some main lines of defense:

- Interpreter level:
  - No pointer arithmetic
  - Garbage collection
  - Array bounds checking
  - No illegal data conversions
  - Byte Code Verifier

Java Feature: Robust

- Reliable
- Early checking for potential problems.
- Dynamic checking to eliminate error-prone situations.
- Developer doesn’t have to worry about
  - Bad pointers
  - Memory allocation errors
  - Memory leakage

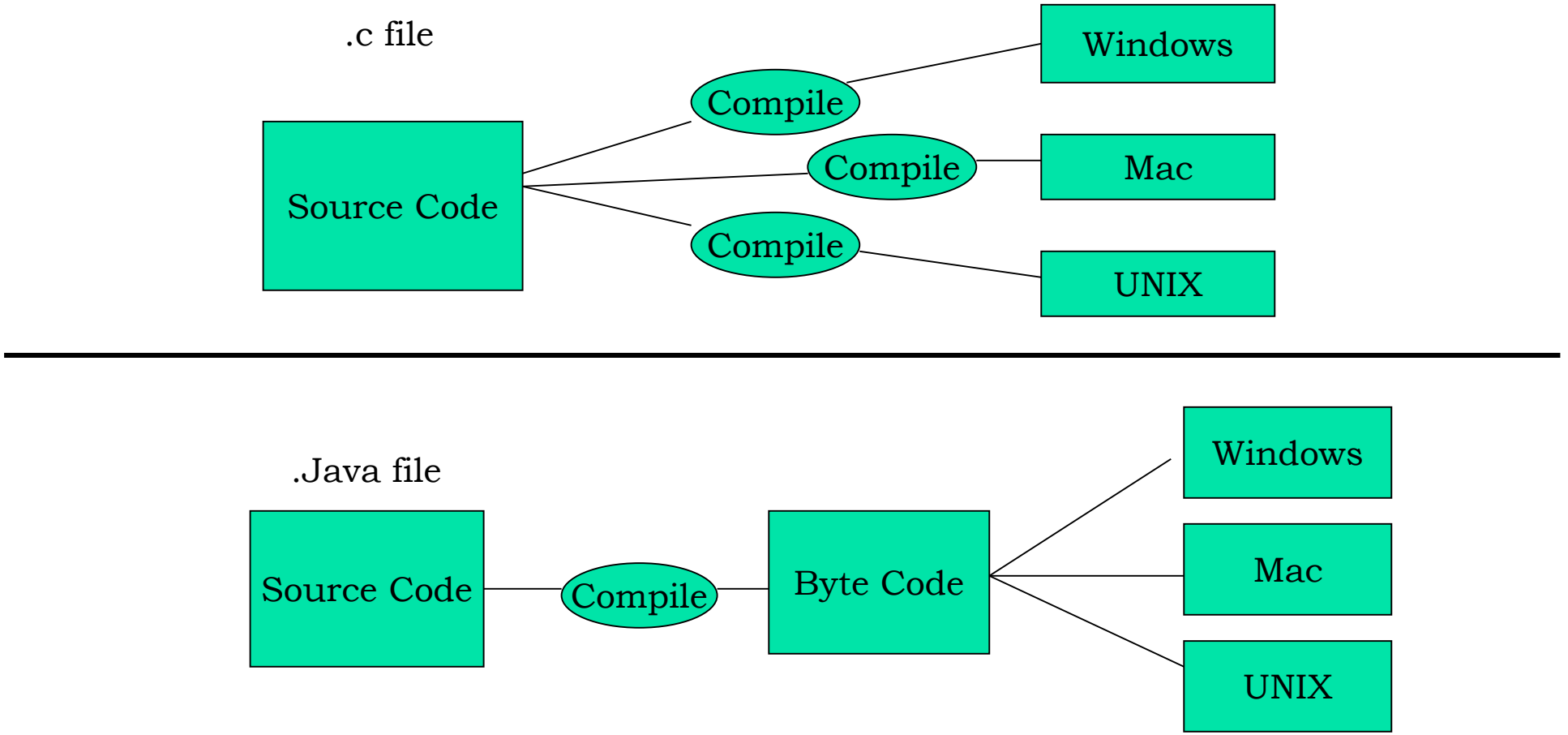
Java Feature: Multithreaded

ACADGILD

- **Thread-safe:** Lib functions implemented such that it can be executed by multiple concurrent threads
- Built-in support for threads

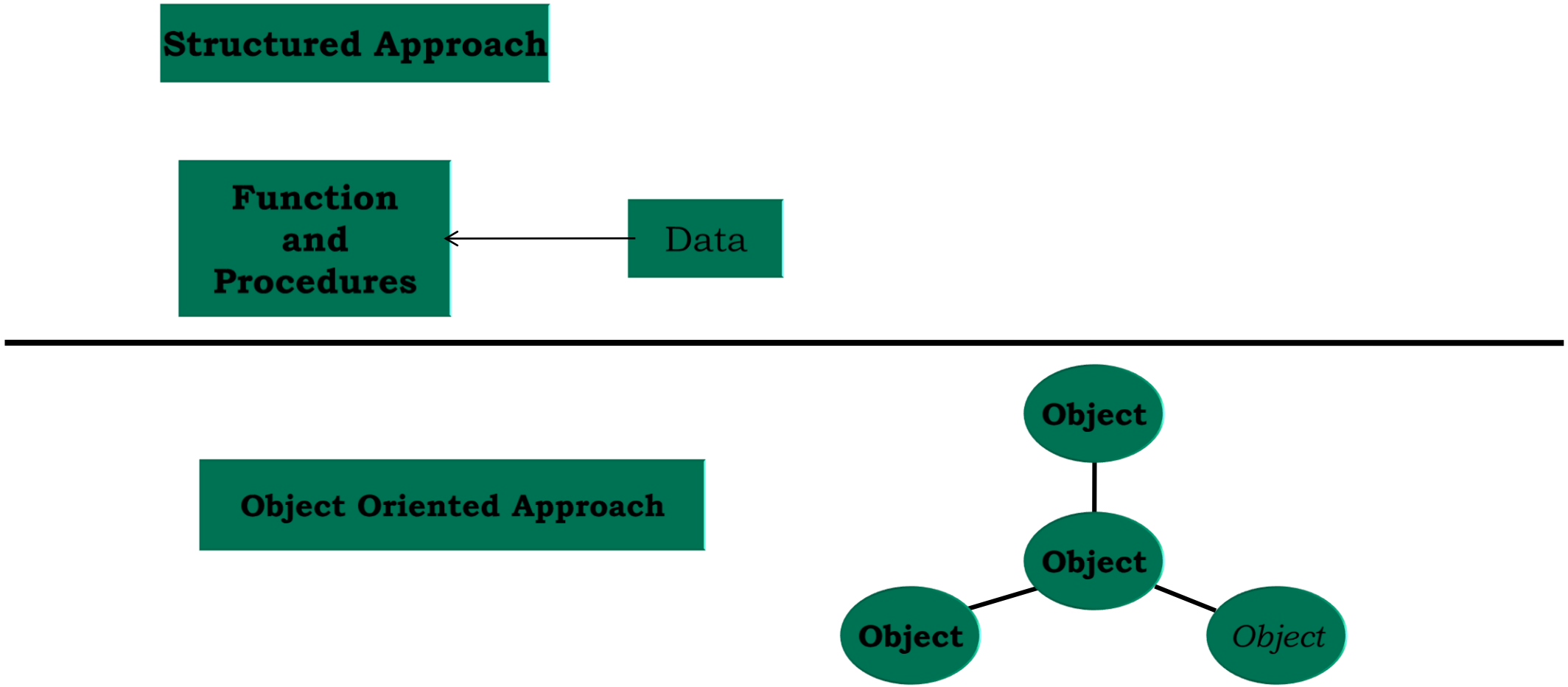
Java Feature: Architecture Neutral

ACADGILD



Structured Vs Object Oriented Approach

ACADGILD



Need for Object Oriented Approach

ACADGILD

- Object-oriented approach is a popular technical approach for analyzing, designing an application and system.
- Applying the object-oriented principles throughout the development life cycles leads to better stakeholder communication and product quality.

Need

- Changing Requirements during development.
- Difficulty of managing software development process.
- Easy User Interface.
- Clients want systems to be adaptable and Extensible.

Need of Object Oriented Approach (Contd.)

The Claim

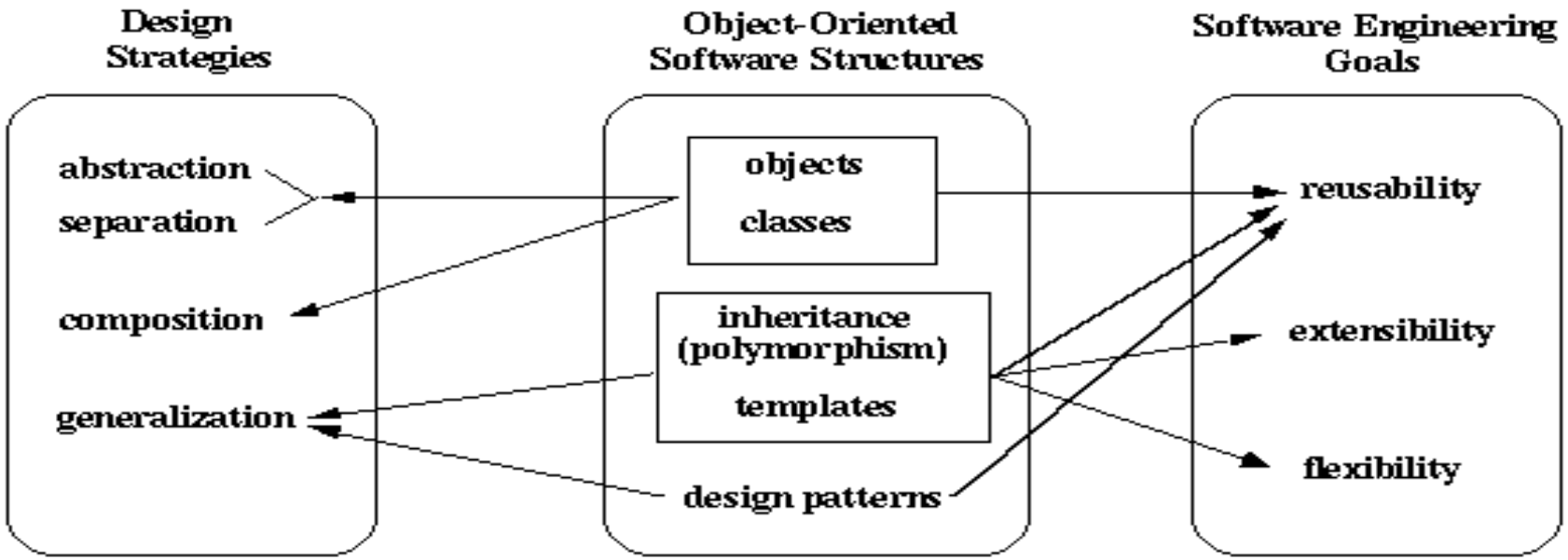
- Object oriented approach helps to handle the complexity of software development and aids in generation of adaptable and extensible systems.

- An object oriented software is composed of discrete objects interacting with each other to give rise to the overall (complex) behavior of the system.

Introduction to OOPs

ACADGILD

- A methodology or paradigm to design a program using classes and objects.
- It simplifies the software development and maintenance.

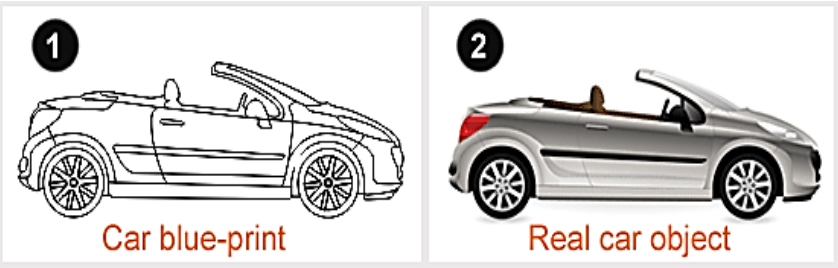


Concept of Class and Object

ACADGILD

Class

- A class can be defined as a template/blue print from which individual objects are created. And that describes the behaviors/states that object of its type support.
- Class is a design whereas object is a real entity based on the class.



Object

- An object is a real world entity that can be a tangible, intangible or a conceptual entity.
- An instance of a class
- An object can be considered as a "thing" that can perform a set of activities. The set of activities that the object performs defines the object's behavior. For example, the hand(object) can Grip something or a Student (object) can give the name or address.

Setting Up of Environment

ACADGILD

For executing any Java program, you need to follow the steps below:

1. Install the JDK. If you don't have it, download the JDK and install it. http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html
2. Set path of the jdk/bin directory on Command Prompt or create Environment variable called path.
3. Create the java program using a Text Editor.
4. Compile and run the java program on Command Prompt.

Otherwise, start using any IDE like eclipse or Netbeans just after installing Java.

Writing First Java Program

ACADGILD

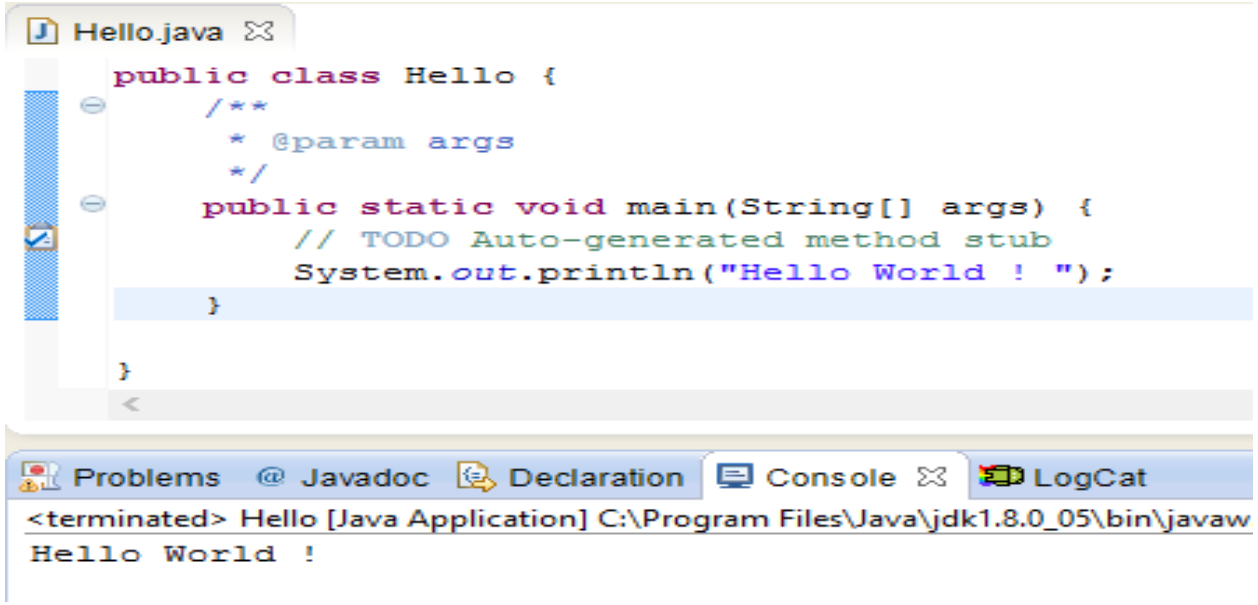
Writing a Java Class

<accessModifier> class <ClassName>{

// data member; // variable declaration and initial assignment of value

// methods & constructors }

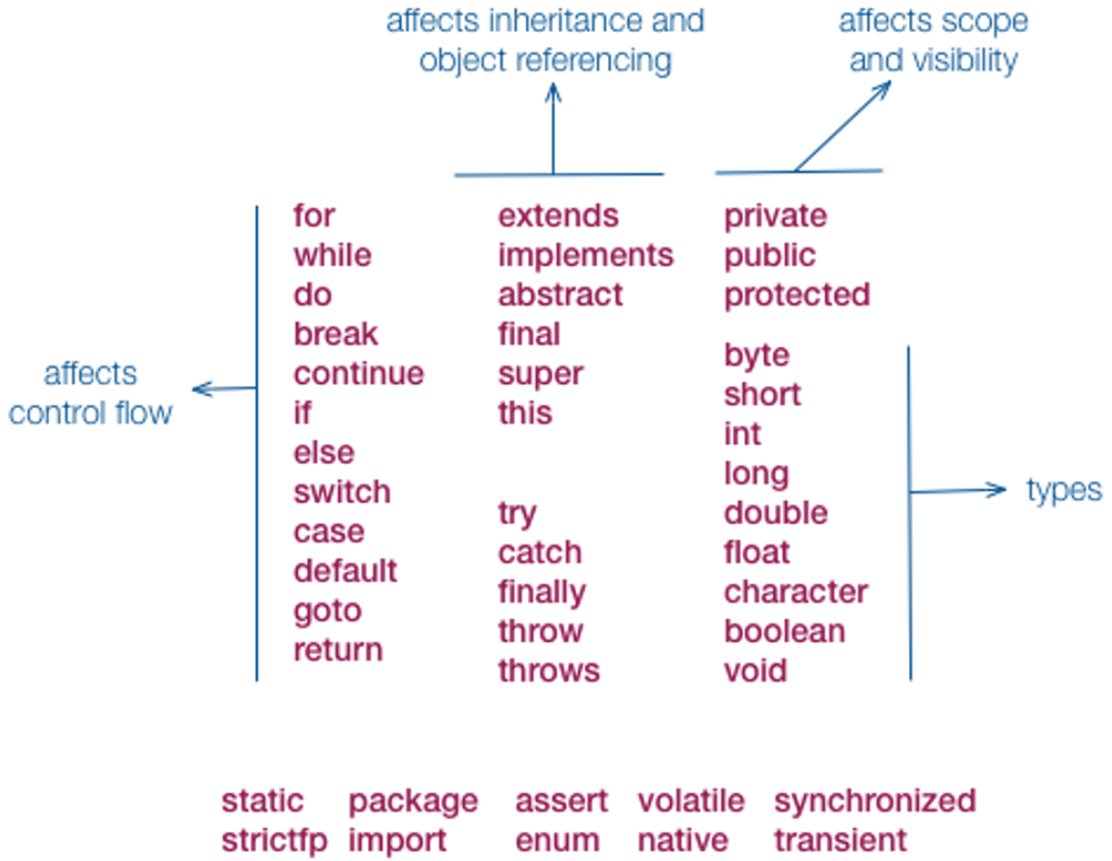
Main method is the point from where the program execution starts



Keywords

ACADGILD

- Every programming language uses some reserved words known as keywords.
- They are an essential part of the language. In Java, keywords are defined in lower case and cannot be used as identifiers.

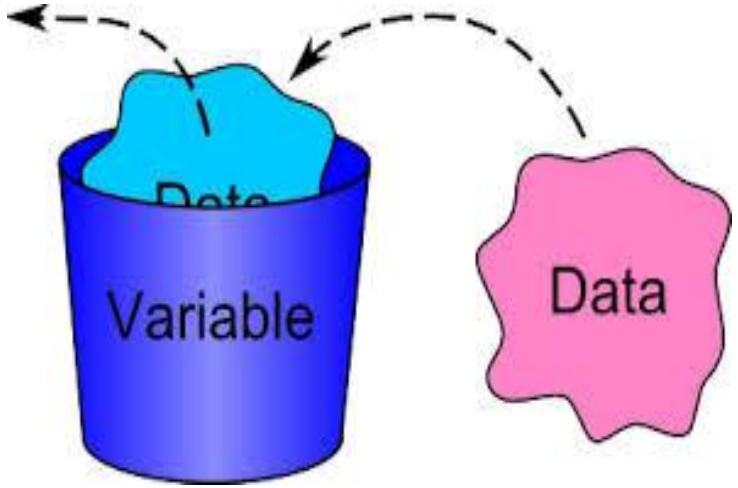


Data Types and Variables

ACADGILD

- Data Types determines the values it may contain, plus the operations that may be performed on it.
- Variables provides us with named storage that our programs can manipulate.

Variable Declaration & Initialization



Syntax for declaration of variable:
DataType variableName;

Syntax for assignment of value to a variable:
variableName=value;

In a single statement declaration and assignment of variable:
DataType variableName=value;

In a single statement declaration and assignment of many variables:
DataType variableName=value, variableName1=value1…;

Different Data Types

ACADGILD

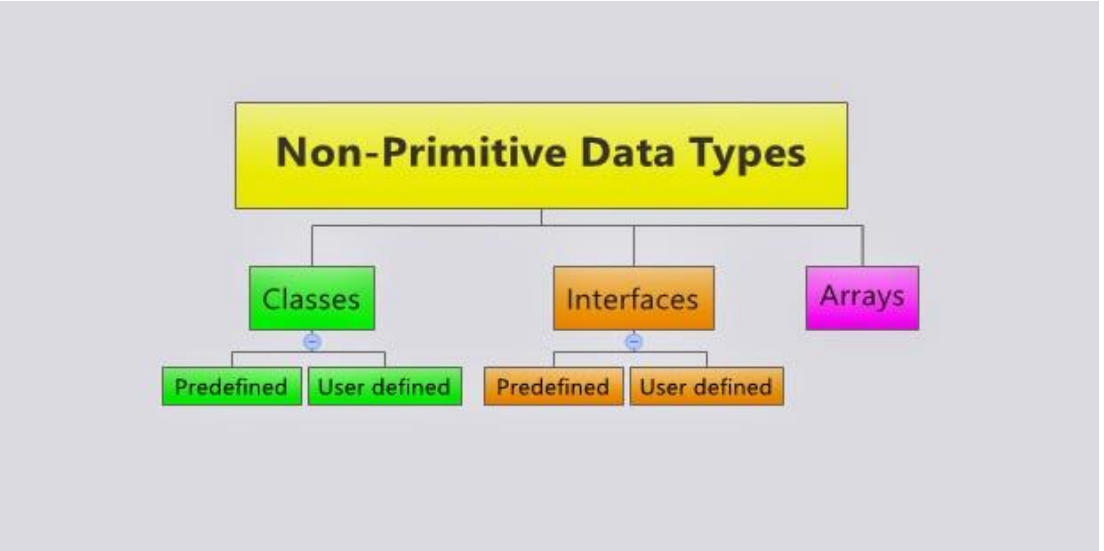
- Java Data Types can be classified into two types:
  - Primitive
  - Non-Primitive

Primitive:

Type Name	Kind of Value	Memory Used	Range of Values
byte	Integer	1 byte	–128 to 127
short	Integer	2 bytes	–32,768 to 32,767
int	Integer	4 bytes	–2,147,483,648 to 2,147,483,647
long	Integer	8 bytes	–9,223,372,036,8547,75,808 to 9,223,372,036,854,775,807
float	Floating-point	4 bytes	$\pm 3.40282347 \times 10^{+38}$ to $\pm 1.40239846 \times 10^{-45}$
double	Floating-point	8 bytes	$\pm 1.79769313486231570 \times 10^{+308}$ to $\pm 4.94065645841246544 \times 10^{-324}$
char	Single character (Unicode)	2 bytes	All Unicode values from 0 to 65,535
boolean		1 bit	True or false

Different Data Types (Contd.)

- **Non-Primitive:**



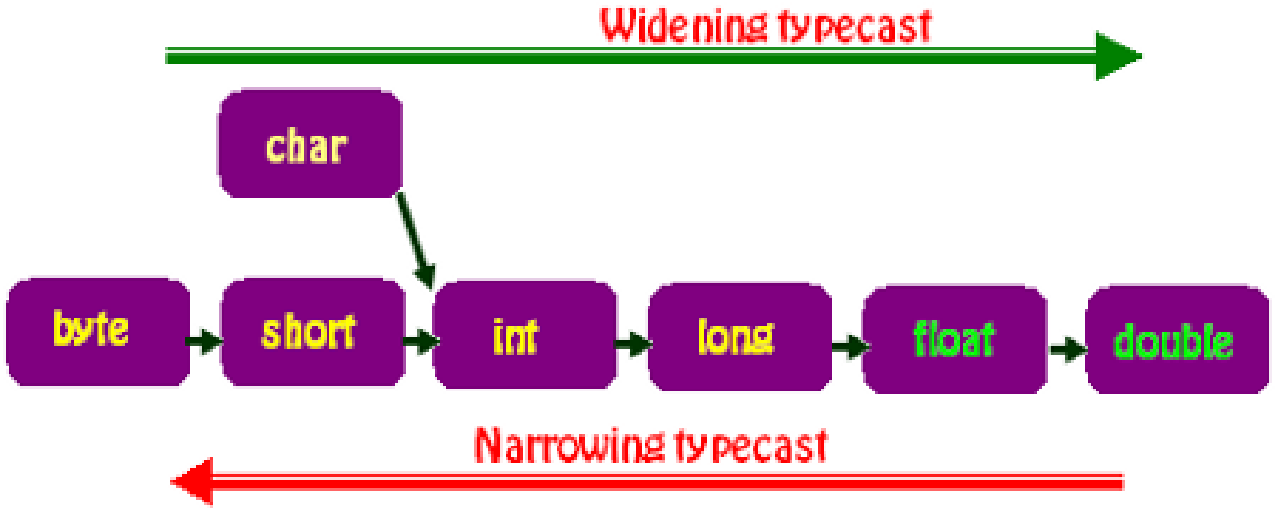
String

- Strings, which are widely used in Java programming, are a sequence of characters.
- It is an inbuilt class in java.
- The most direct way to create a string is to write:
- String greeting = "Hello world!";

Type Casting

Assigning a value of one type to a variable of another type is known as Type Casting.

- Widening Casting/Implicit/Casting/Automatic Type casting - It takes place when the two types are compatible & the target type is larger than the source type.
- Narrowing Casting/Explicit type casting - When you are assigning a larger type value to a variable of smaller type, then you need to perform explicit type casting.



Operators in Java

ACADGILD

- Java provides a rich set of operators to manipulate variables. We can divide all the Java operators into the following groups:

Category	Operator	Associativity
Postfix	() [] -> . ++ - -	Left to right
Unary	+ - ! ~ ++ - - (type) * & sizeof	Right to left
Multiplicative	* / %	Left to right
Additive	+ -	Left to right
Shift	<< >>	Left to right
Relational	< <= > >=	Left to right
Equality	== !=	Left to right
Bitwise AND	&	Left to right
Bitwise XOR	^	Left to right
Bitwise OR	|	Left to right
Logical AND	&&	Left to right
Logical OR	||	Left to right
Conditional	?:	Right to left
Assignment	= += -= *= /= %= >>= <<= &= ^= |=	Right to left
Comma	,	Left to right

User Input in Java

ACADGILD

- Java provides a rich API for data input(java.io) from user but to start with, lets discuss java.util.Scanner class.
- Scanner class allows you to take input for different types of variables by providing different method for each type. For e.g.,
  - nextInt() for input in int variable
  - nextFloat() for input in float variable

//for more see the documentation of Scanner class

https://docs.oracle.com/javase/7/docs/api/java/util/Scanner.html

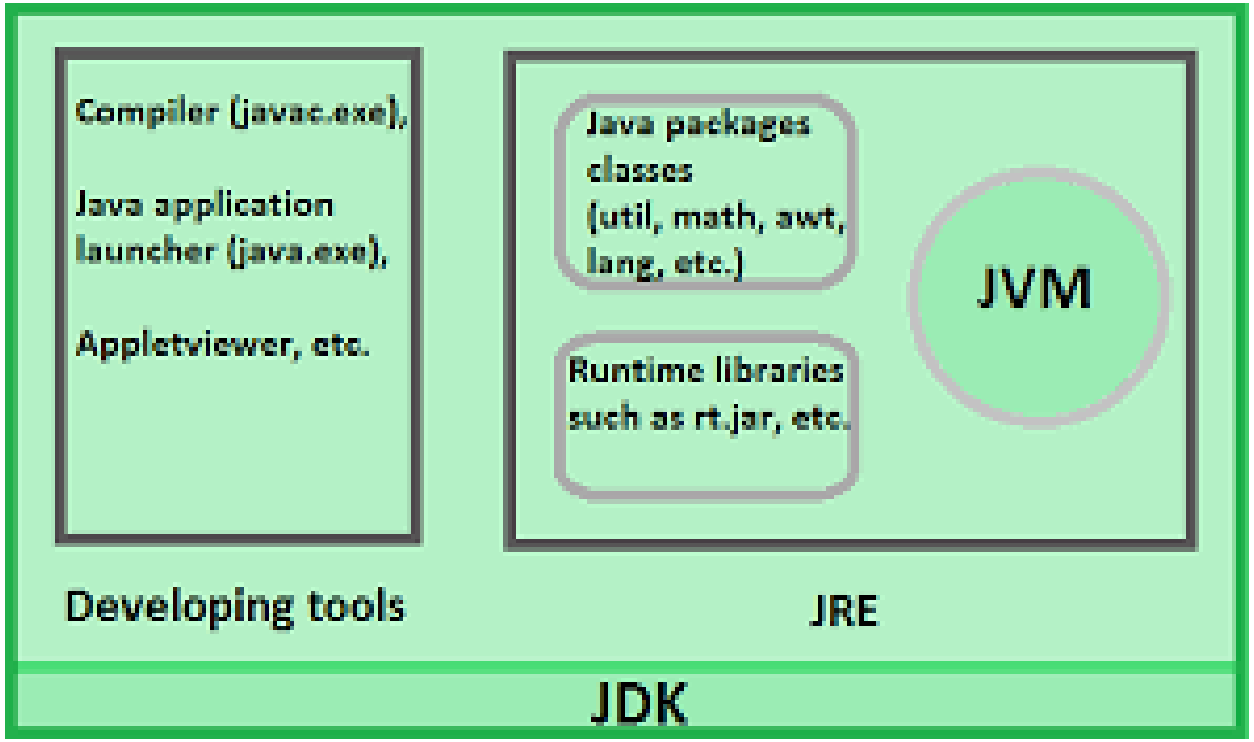
Steps of Using Scanner class:

- First of import java.util.Scanner class.
- Create object of Scanner class
- Start using its methods.

Introduction to JDK, JVM and JRE

ACADGILD

- JDK is Java Developer Kit which you need to compile Java source code
- JVM is Java Virtual Machine which runs Java bytecode.
- JRE is Java Runtime Environment which you need to run a java program. It contains a JVM, among other things.



Short Assignments

ACADGILD

- A Vehicle has an owner and a registration number. A Car is a Vehicle that has a steering wheel and a model name. Choose the fields which must be included in the Car class. (The Vehicle class is already defined) Choose at least one answer.

a. Steering s;

b. Person owner;

c. String registrationNo;

d. String modelName;

e. Vehicle c;

- A blueprint for a software object is called a ___.

THANK YOU

Email us at - support@acadgild.com