SILVER OAK UNIVERSITY

EDUCATION TO INNOVATION

SUBJECT: CORE JAVA

TOPIC: Java-Programming-Constructs

FACULTY NAME: Dr. Nikunj K. Raval

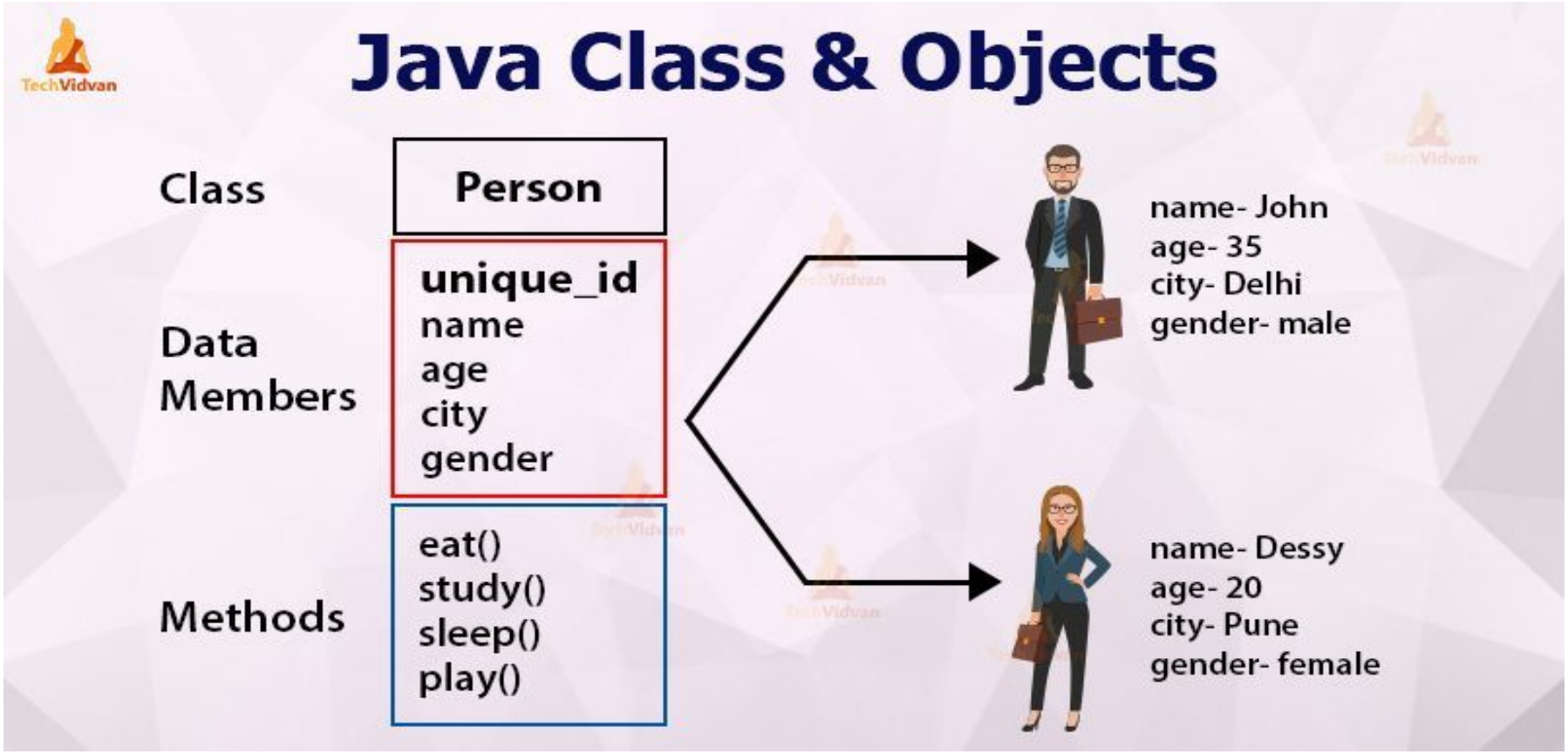
What is JAVA?

Java is a widely used object-oriented programming language and software platform that runs on billions of devices, including notebook computers, mobile devices, gaming consoles, medical devices and many others. The rules and syntax of Java are based on the C and C++ languages.

One major advantage of developing software with Java is its portability. Once you have written code for a Java program on a notebook computer, it is very easy to move the code to a mobile device. When the language was invented in 1991 by James Gosling of Sun Microsysts (later acquired by Oracle), the primary goal was to be able to "write once, run anywhere."

JAVA Features:-

Concept of Class:-



Object Oriented Programming in Java

Java

www.educba.com

Introduction to class diagram(UML diagram):-

- The class diagram depicts a static view of an application.

It represents the types of objects residing in the system and the relationships between them.

- A class consists of its objects, and also it may inherit from other classes.
- A class diagram is used to visualize, describe, document various different aspects of the system, and also construct executable software code.

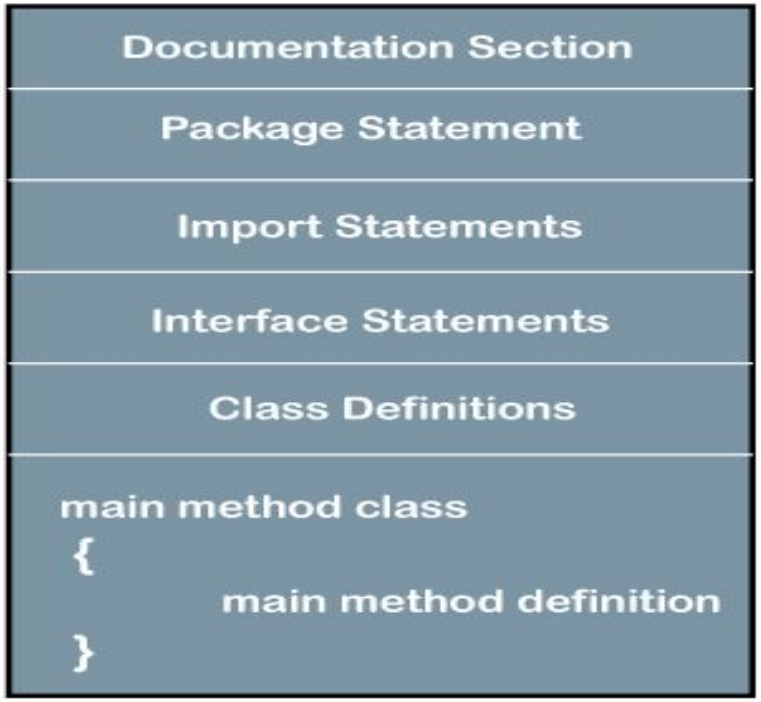
Purpose of Class Diagrams

The main purpose of class diagrams is to build a static view of an application. It is the only diagram that is widely used for construction, and it can be mapped with object-oriented languages.

1. It analyses and designs a static view of an application.
2. It describes the major responsibilities of a system.
3. It is a base for component and deployment diagrams.
4. It incorporates forward and reverse engineering.

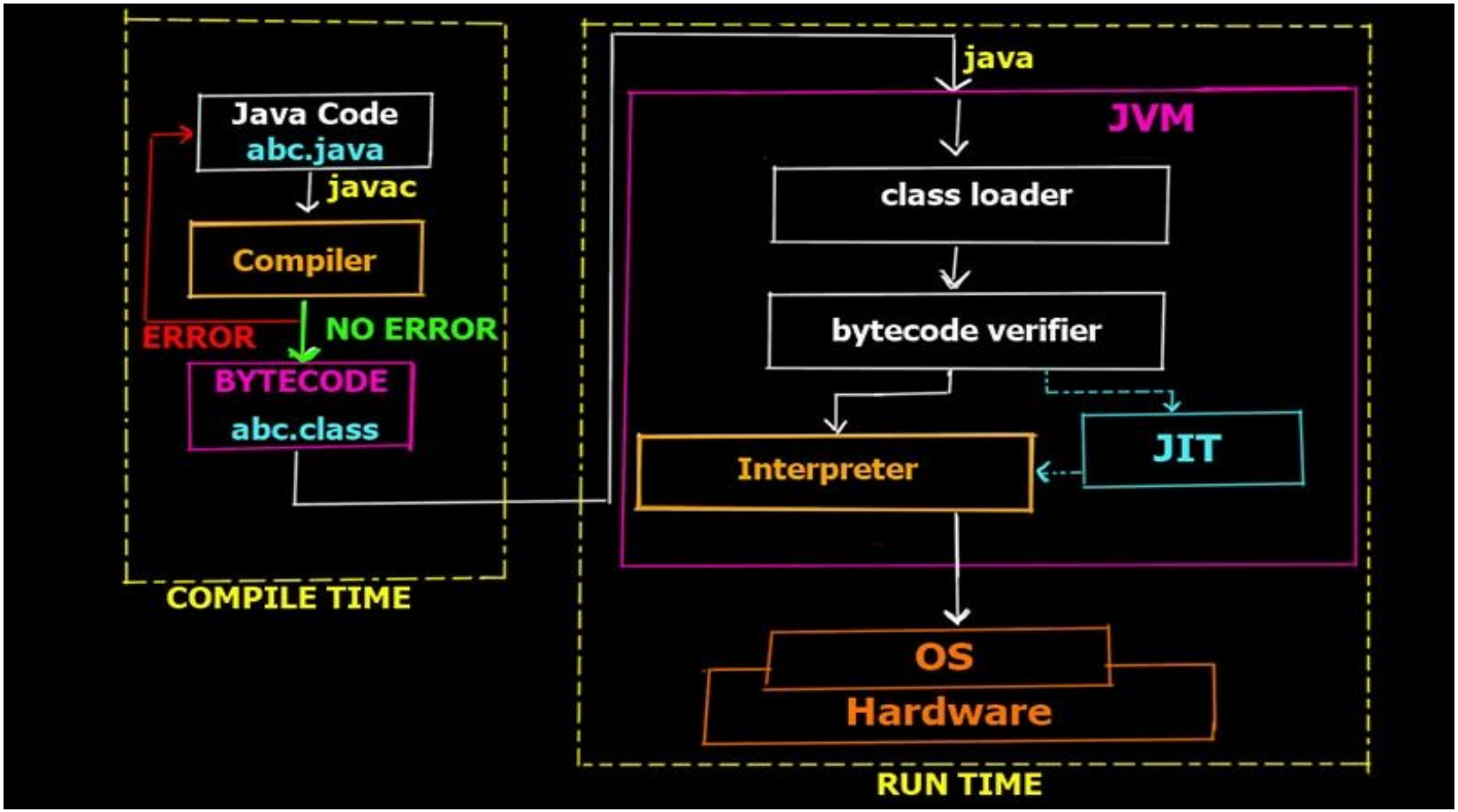
Structure of Java Program

- Java is an object-oriented programming, **platform-independent,** and **secure** programming language that makes it popular.
- Using the Java programming language, we can develop a wide variety of applications. So, before diving in depth, it is necessary to understand the **basic structure of Java program** in detail.



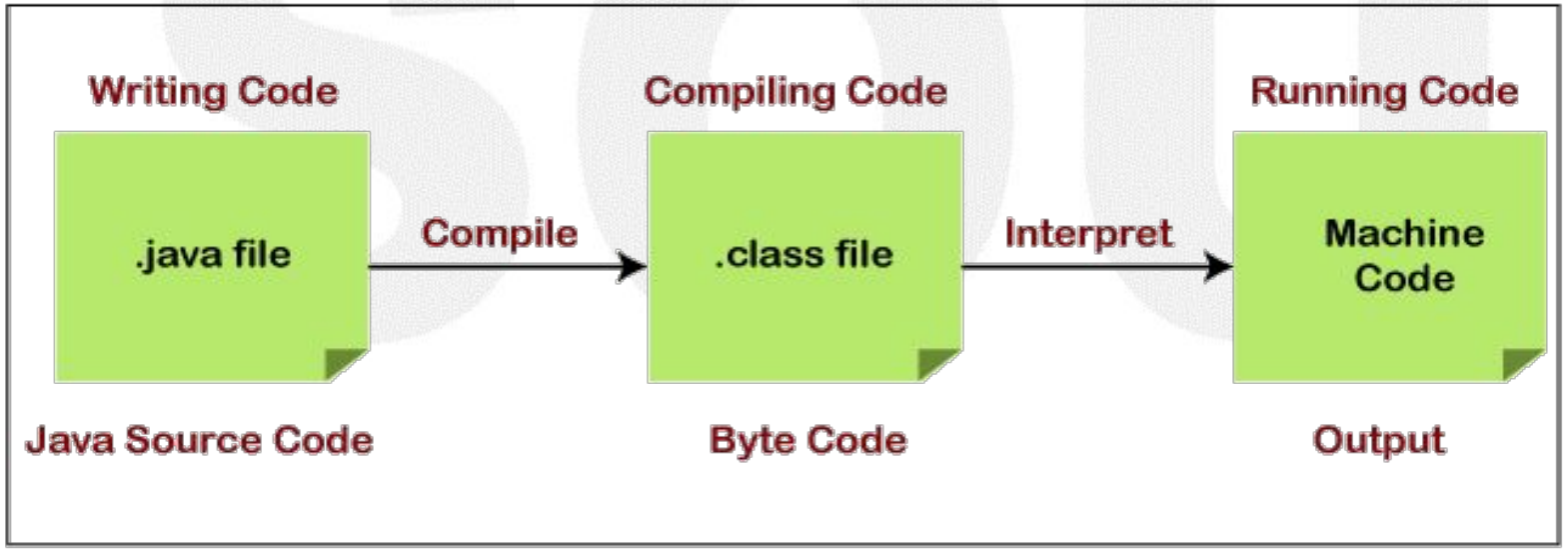
Structure of Java Program

Compiling of JAVA source file and



run the application using JAVA interpreter.

- **Java interpreter** is a computer program (system software) that implements the JVM. It is responsible for reading and executing the program. It is designed in such a way that it can read the source program and translate the source code instruction by instruction.

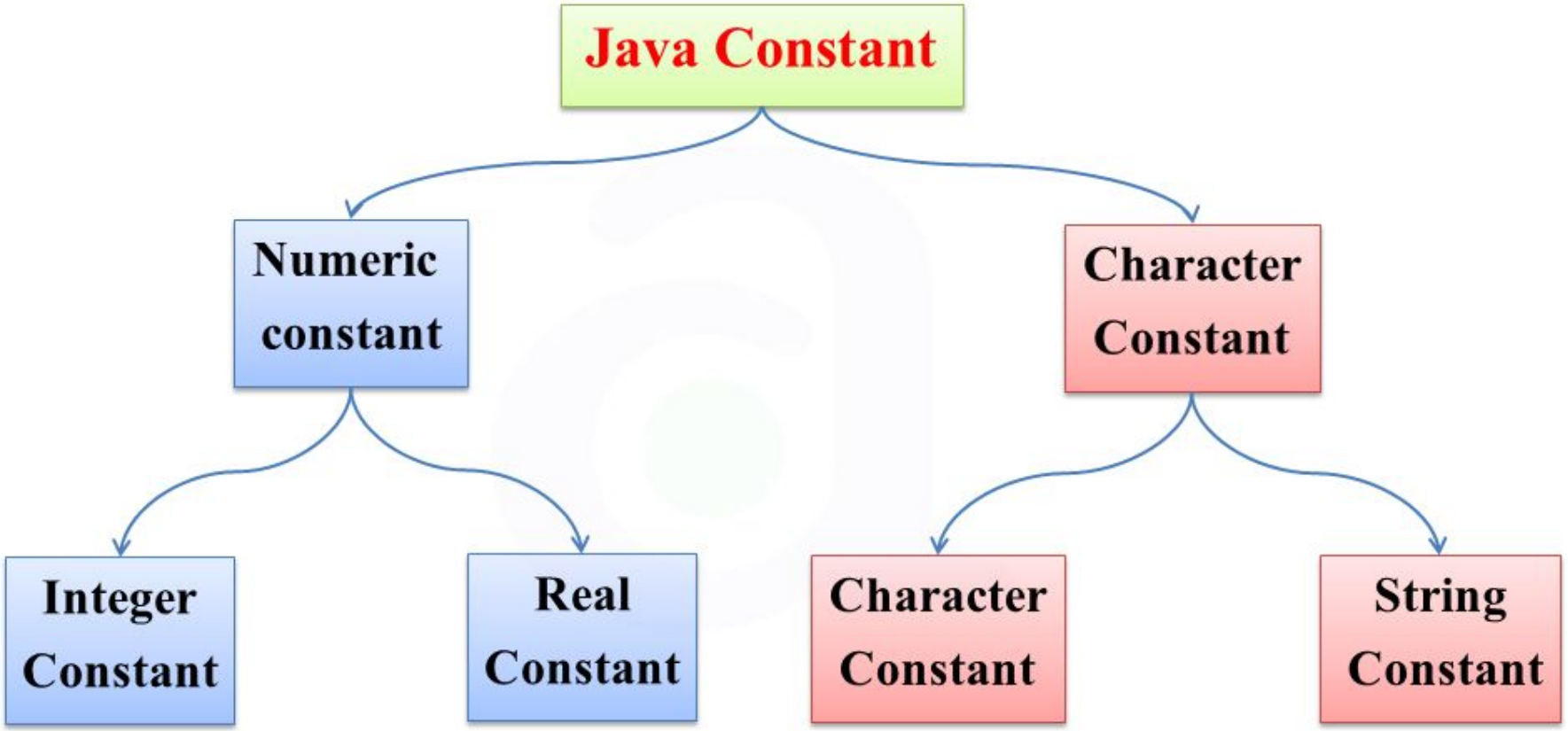


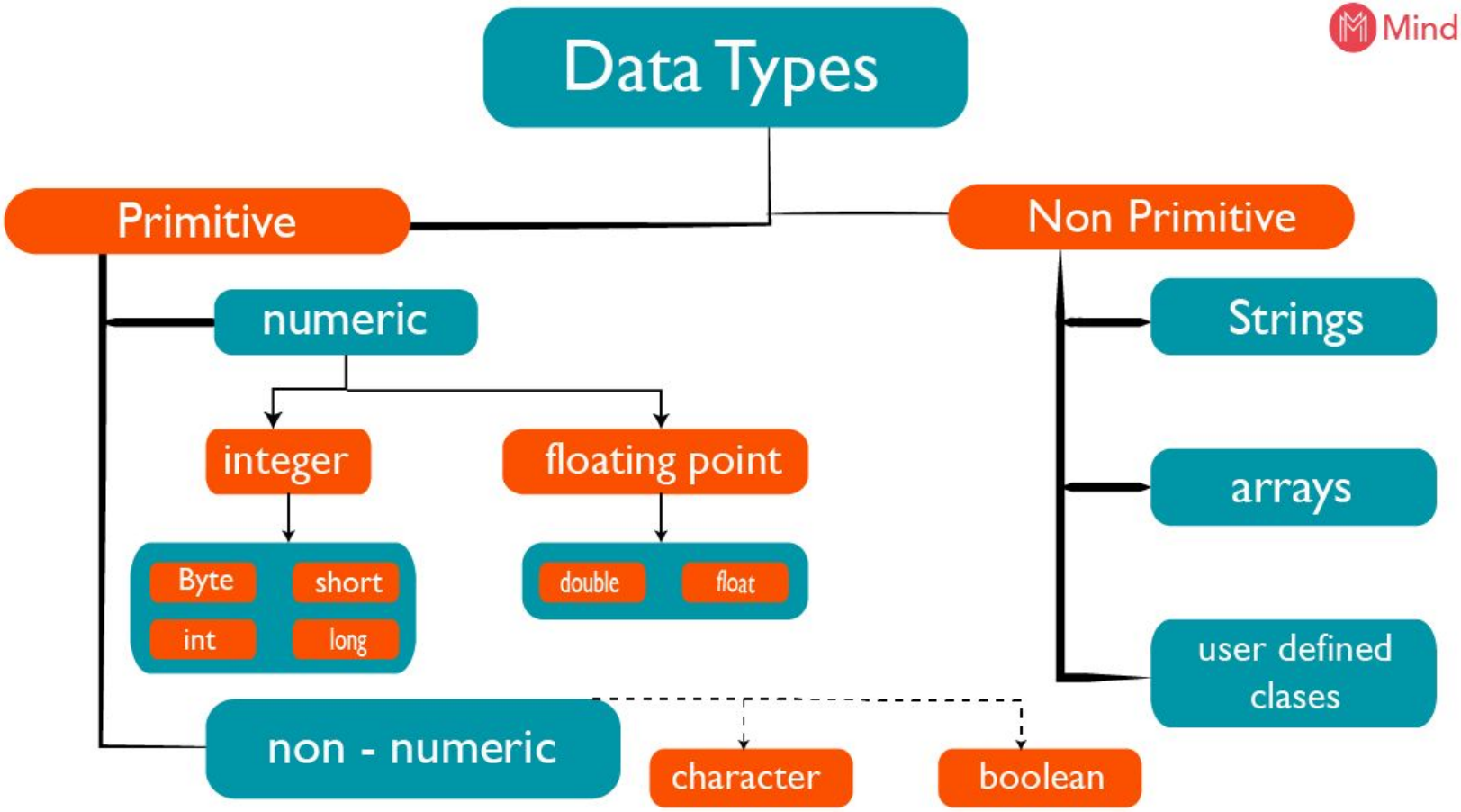
Features of Interpreter

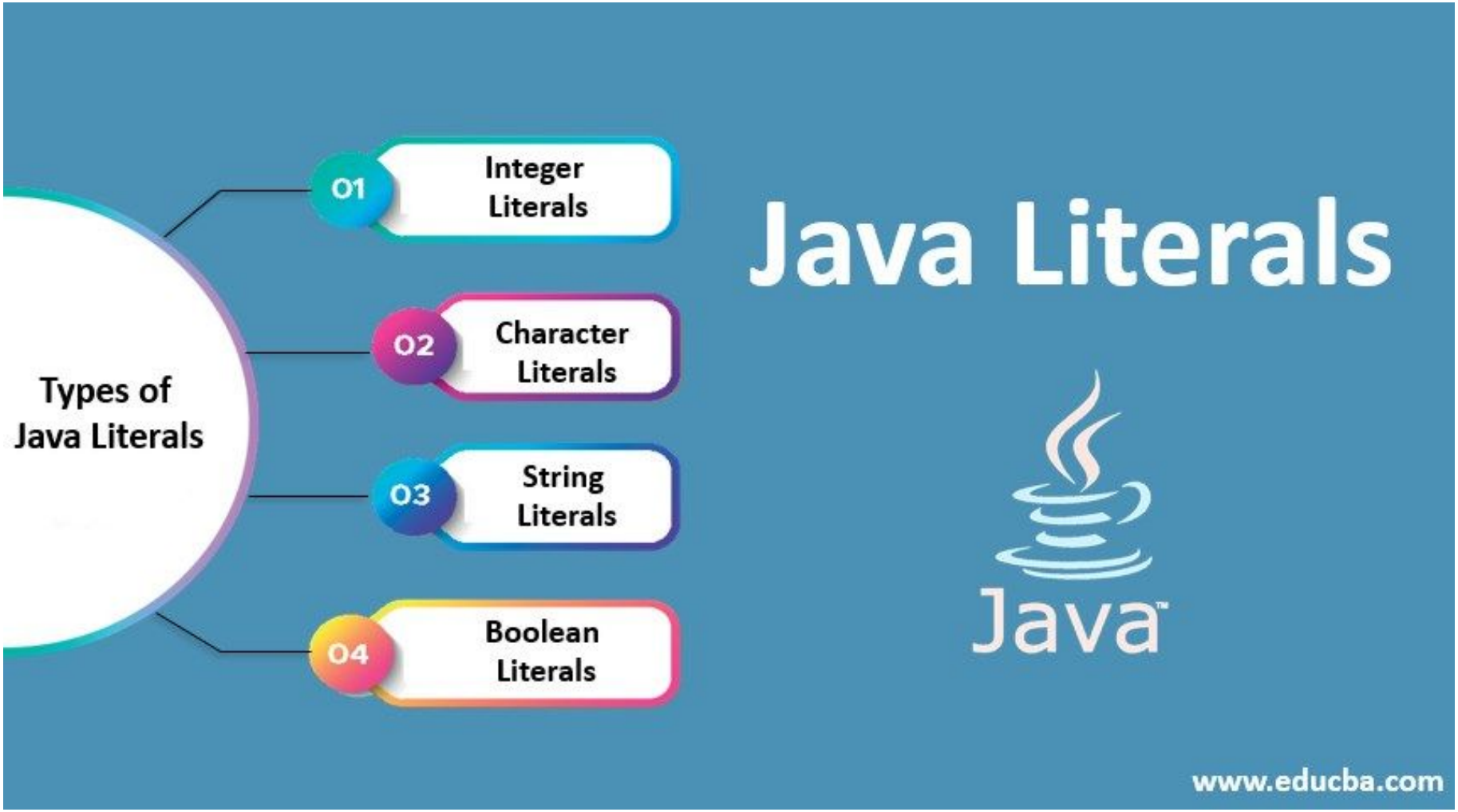
It converts the source code into machine language, line by line at run time, without changing the sequence.

- An interpreter does not generate an intermediate machine code
- Each error of every line is displayed one by one
- When compared to a compiler, the program execution speed is slower
- Less amount of time is spent on analyzing and processing the program

Interpreter	Compiler
It translates the code instruction by instruction.	It translates the entire program at once.
Its execution is slower.	Its execution is faster.
Its compile time is less.	It takes more time to compile the code.
It does not generate the intermediate object code.	It generates the intermediate object code.
It compiles the program until an error is found.	All the errors show once at the end of the compilation.
Python, PHP, Ruby, and Perl use an interpreter.	Java, C++, Scala, and C uses a compiler.


Java Constant
Numeric constant
Character Constant
Integer Constant
Real Constant
Character Constant
String Constant


MindMajix
Data Types
Primitive
Non Primitive
numeric
Strings
integer
floating point
arrays
Byte
short
int
long
double
float
user defined clases
non - numeric
character
boolean


Types of Java Literals
01 Integer Literals
02 Character Literals
03 String Literals
04 Boolean Literals
Java Literals
Java
www.educba.com

Keywords

- **keyword**: An identifier that you cannot use because it already has a reserved meaning in Java.

abstract	default	if	private	this
boolean	do	implements	protected	throw
break	double	import	**public**	throws
byte	else	instanceof	return	transient
case	extends	int	short	try
catch	final	interface	**static**	**void**
char	finally	long	strictfp	volatile
class	float	native	super	while
const	for	new	switch	
continue	goto	package	synchronized	

```
class DogName
{
int Age;
}
```

Correct

```
class Dog Name
{
int AGE;
}
```

error, because identifier can't have space

```
class DogName
{
int dogAge;
}
```

Correct

```
class Dog Name
{
int dog AGE;
}
```

error, because identifier can't have space


Operators in Java
Arithmetic Operators
Unary Operators
Assignment Operators
Relational Operators
Logical Operators
Ternary Operators
Bitwise Operators
Shift Operators
1
2
3
4
5
6
7
8

Installation of JDK 1.7

Java Development Kit 1.7 is an essential tool for developers to write and run Java applications. In this presentation, we will guide you through the installation process in simple steps.

Why use JDK 1.7?

1 **Compatibility**

JDK 1.7 can run software written in earlier versions of Java while providing support for new features.

Improved performance

The new features of JDK 1.7 like the "fork/join" framework, multicore JIT, and G1 garbage collector improve the performance of Java applications.

Bug fixes and security patches

Using the latest version ensures access to all bug fixes and security patches released by Oracle.

Steps to install JDK 1.7

1

System requirements

Check that your system is compatible with JDK 1.7 and has enough memory available.

2

Downloading the JDK installer

Download the installer from the recommended site. The file size is approximately 80MB.

Execute the installer and follow the instructions on the wizard. Accept the terms of use and select your preferred installation path.

Setting up environment variables

Add the JDK installation path to the PATH environment variable. Follow the instructions on the Oracle website for your operating system.

JDK 1.7 installation made easy

Choose the correct OS version

Select the installation package compatible v operating system.

Configure environment variable

Add the JDK installation path to the PATH environment variable to access it from the c line.

Execute the installer and follow the prompts. It's that simple!

Ensure the installation is successful by running simple Java code snippets. Troubleshoot issues by reading error messages and browsing online forums.

SILVER OAK UNIVERSITY

www.silveroakuni.ac.in