

C - D A C M U M B A I

JAVA

MANOJ MORE

A U G U S T 2 0 2 5

.....

JAVA

.....

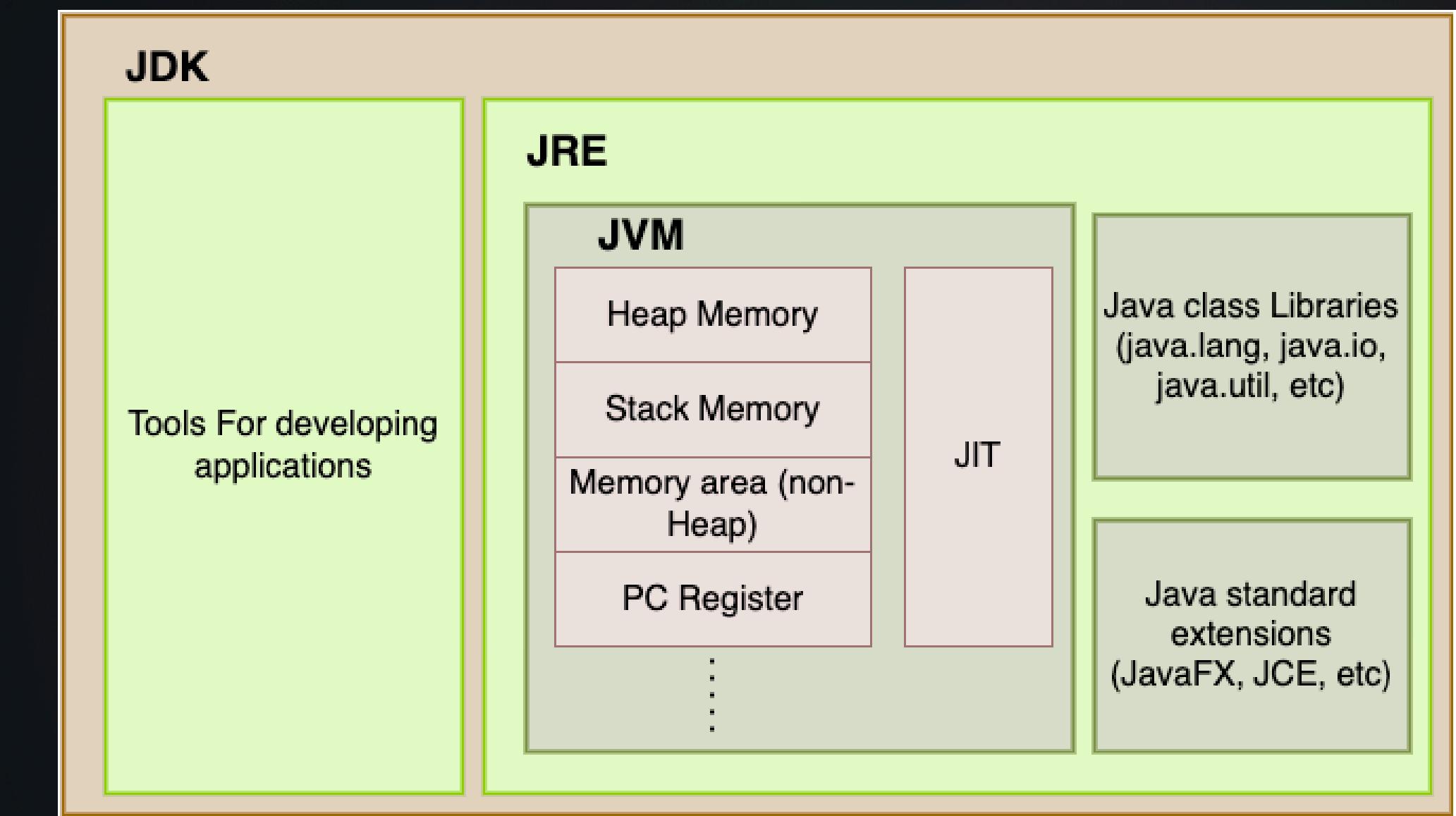
Author : James Gosling

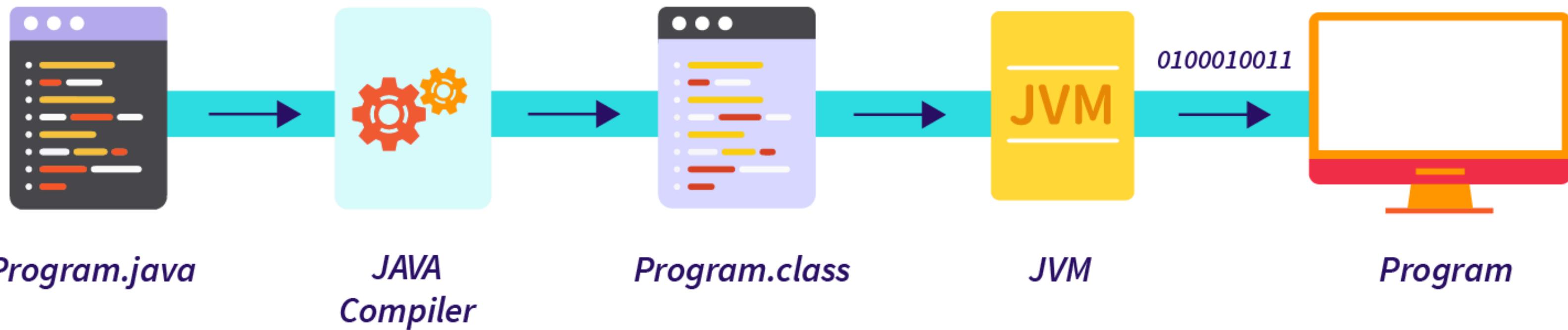
Vendor : Sun Micro System

Project Name : Green Project

Type : Open Source & Free Software

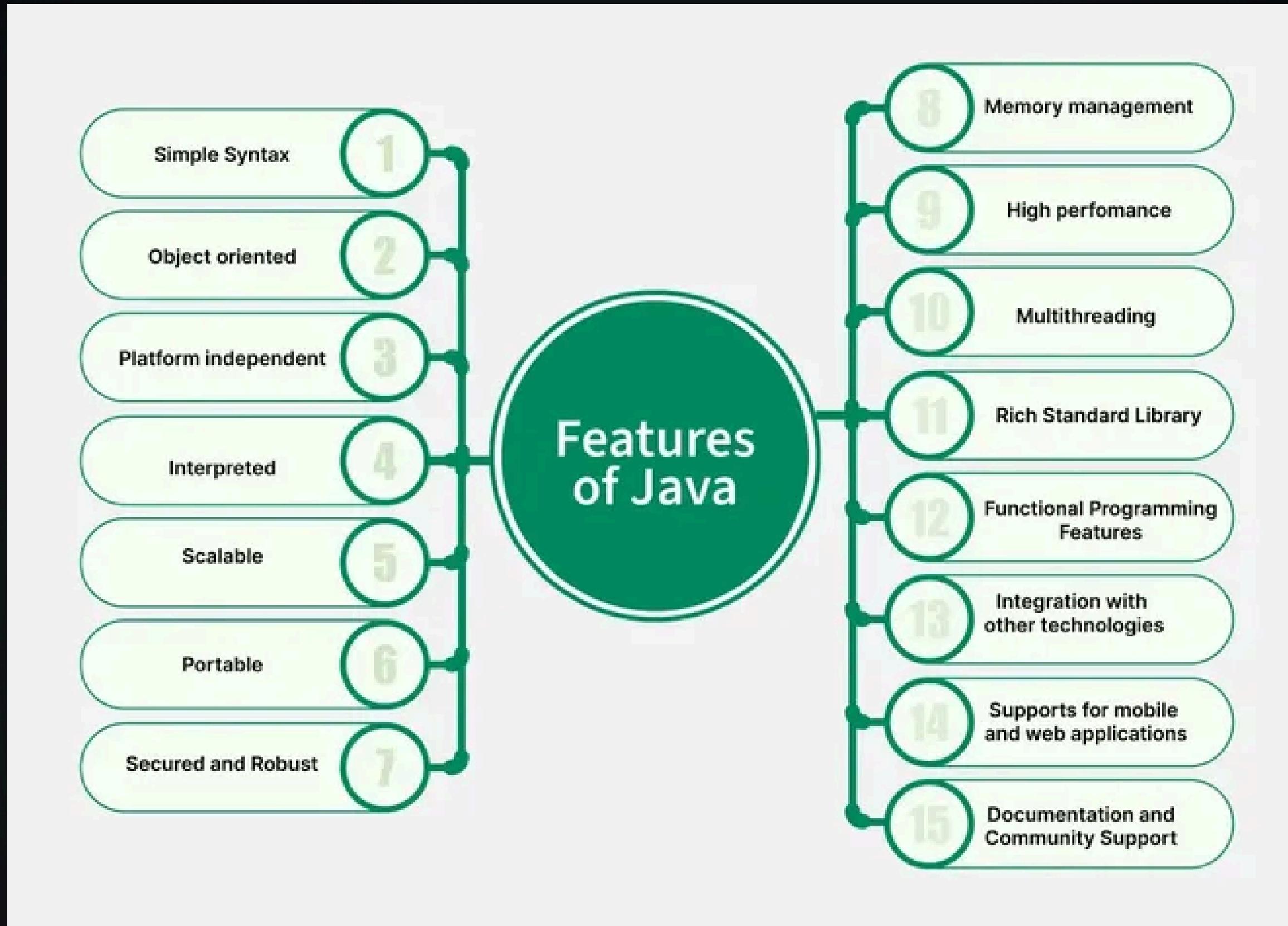
Initial Name : OAK Language





KEY FEATURES OF JAVA

.....



CLASSES AND OBJECTS IN JAVA

.....

- A class is a template to create objects having similar properties and behavior, or in other words, we can say that a class is a blueprint for objects.
- An object is an instance of a class. For example, the animal type Dog is a class, while a particular dog named Tommy is an object of the Dog class.





HELLO WORLD!

.....

- Let's Write Our First Java Project!!!

DATA TYPES

Bits and Bytes in Programming

- Bits and bytes are the smallest units of data in a computer.
- A bit is a single binary digit, with a value of either 0 or 1.
- A byte is a group of 8 bits.
.....

- In computer memory, a bit is stored as electrical voltage, where a voltage above a certain threshold represents a 1, and a voltage below that threshold represents a 0.
- In hard disk drives, a bit is stored as magnetism, where an area magnetized in one orientation represents a 1, and a magnetized area in the opposite orientation represents a 0.
- In CDs, DVDs, and Blu-ray discs, a bit is stored as either a pit, or a flat area. A pit is an area where the surface is lower than the surrounding surface, and that represents a 1. A flat area is when there is no pit, and that represents a 0.

DATA TYPES

Bits and Bytes in Programming

What is a Byte?

- A byte is a group of 8 bits, like 10001011.
- Each bit can be either 0 or 1, and with 8 bits in a byte, there are $2^8 = 256$ different values a byte can have.
- Using one byte, we can store:
 - A pixel with one out of 256 different colors.
 - An unsigned number from 0 to 255.
 - A signed number from -128 to 127.
 - A character from the ASCII table.

DECIMAL TO BINARY

$123 / 2 = 61$ remainder 1

$61 / 2 = 30$ remainder 1

$30 / 2 = 15$ remainder 0

$15 / 2 = 7$ remainder 1

$7 / 2 = 3$ remainder 1

$3 / 2 = 1$ remainder 1

$1 / 2 = 0$ remainder 1

**Reading the remainders from bottom to top,
the binary representation of 123 is 01111011**

DATA TYPES

Java has eight primitive data types, each with a fixed size and range, ensuring platform independence.

Integer Types:

- byte: 1 byte (8 bits),
 - range -128 to 127.
- short: 2 bytes (16 bits),
 - range -32,768 to 32,767.
- int: 4 bytes (32 bits),
 - range -2,147,483,648 to 2,147,483,647.
- long: 8 bytes (64 bits),
 - range -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807.

Floating-Point Types:

- float: 4 byte (32 bits),
 - Stores fractional numbers. Sufficient for storing 6 to 7 decimal digits.
- double: 8 bytes (64 bits),
 - Stores fractional numbers. Sufficient for storing 15 to 16 decimal digit.

DATA TYPES

Java has eight primitive data types, each with a fixed size and range, ensuring platform independence.

Other Types:

- **char**: 2 byte (16 bits),
 - represents a single Unicode character, range 0 to 65,535
- **boolean**: 4 bytes (32 bits),
 - Size is not precisely defined by the JVM specification but is typically considered 1 byte for storage and manipulation, representing true or false.

VARIABLES

Variables are containers for storing data values.



float

stores floating point numbers, with decimals, such as 19.99 or -19.99

M A N L J H U T F N L B T I
O S L I T T A K V W G I Q I
N J E G V V L U C I N E K R
V I S C H I O R U Q U U Q S
Q K B A B B O N A T A L E W
K V P A N E T T O N E I F K
V I U N A S T R I N N E V E
I H R E G A L I V E U Z G J
Y A N G E L O J R A N T S X
S T E L L A Y X B N A S T C
E L F O F F B I S C O T T I
B P I A L B E R O G A H I X
O K D G H I R L A N D A S K
A I P V Q K D K F R E N N A

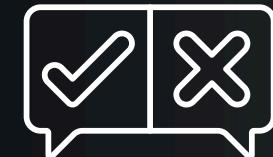
String

stores text, such as "Hello". String values are surrounded by double quotes



int

stores integers (whole numbers), without decimals, such as 123 or -123



boolean

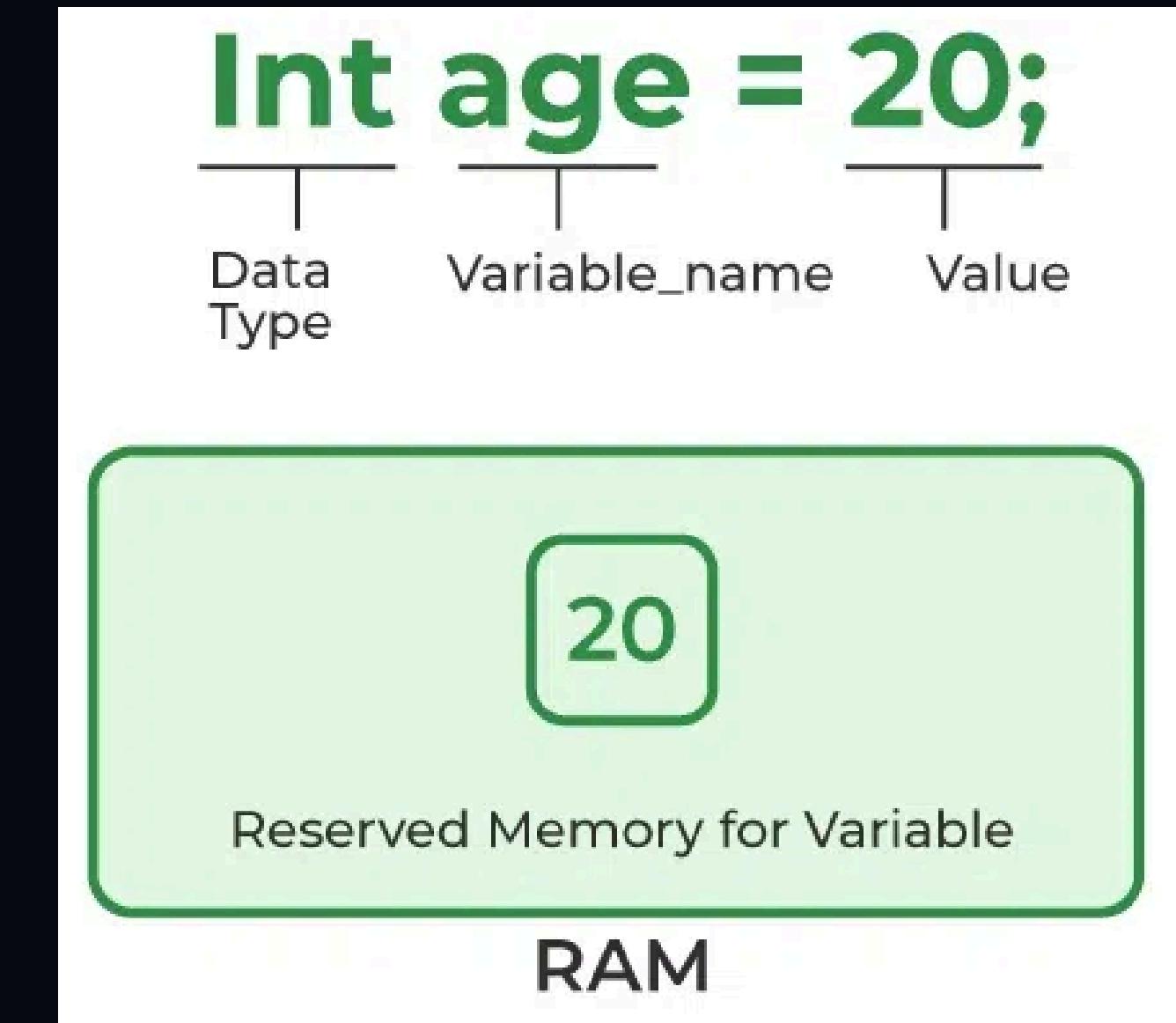
A B C D E F G
H I J K L M N
O P Q R S T U
V W X Y Z

char

stores single characters, such as 'a' or 'B'. Char values are surrounded by single quotes

DATA TYPES

- Data Type: Defines the kind of data stored (e.g., int, String, float).
- Variable Name: A unique identifier following Java naming rules.
- Value: The actual data assigned to the variable.

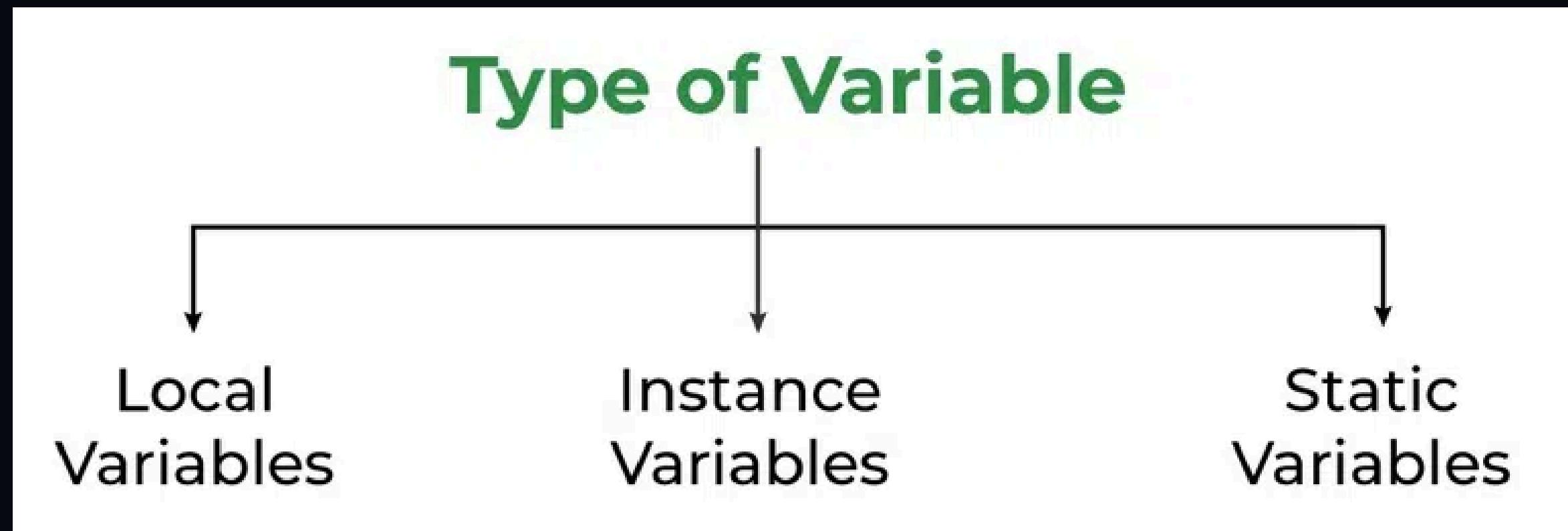


RULES TO DEFINE VARIABLES

- A variable name can consist of Capital letters A-Z, lowercase letters a-z digits 0-9, and two special characters such as _ underscore and \$ dollar sign.
- The first character must not be a digit.
- Blank spaces cannot be used in variable names.
- Java keywords cannot be used as variable names.
- Variable names are case-sensitive.
- There is no limit on the length of a variable name but by convention, it should be between 4 to 15 chars.
- Variable names always should exist on the left-hand side of assignment operators.

TYPES OF JAVA VARIABLES

- Local Variables: Defined within a block or method or constructor
- Instance Variables: non-static variables and are declared in a class outside of any method, constructor, or block
- Static Variables: Declared similarly to instance variables. The difference is that static variables are declared using the static keyword within a class outside of any method, constructor, or block.



JVM MEMORY (RUNTIME DATA AREAS)

These are memory areas created by the JVM during program execution.

Method Area: Stores class-level data, including static variables, methods, and constructor information. There is one Method Area per JVM.

Heap: The runtime data area where objects and arrays are allocated. It is shared among all threads and is subject to garbage collection.

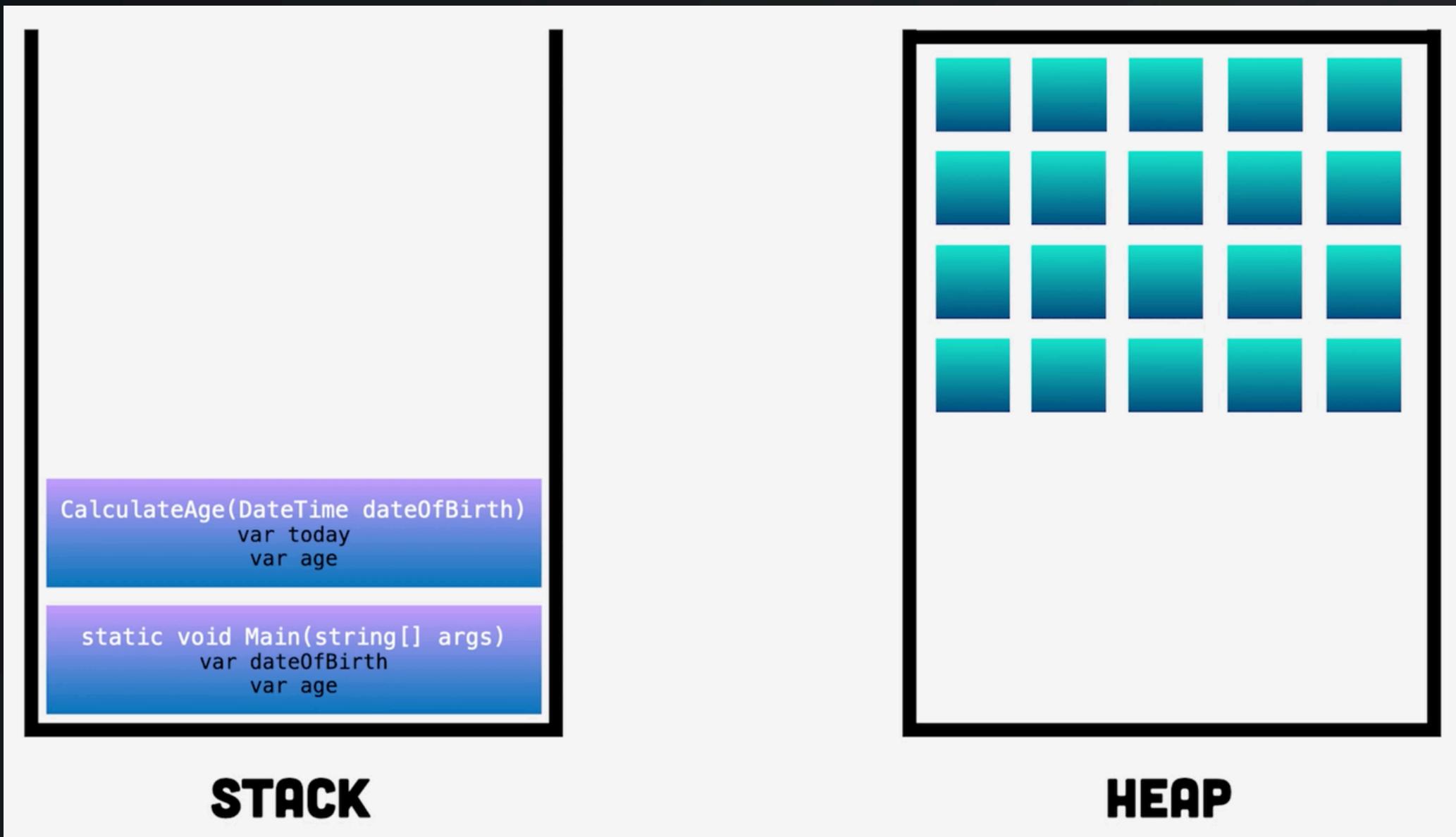
Stack: Each thread in the JVM has its own private Stack, used to store local variables, partial results, and data for method calls and returns.

PC (Program Counter) Register: Each thread has its own PC Register, which stores the address of the currently executing instruction.

Native Method Stacks: Stores information related to native methods (methods written in languages like C/C++).

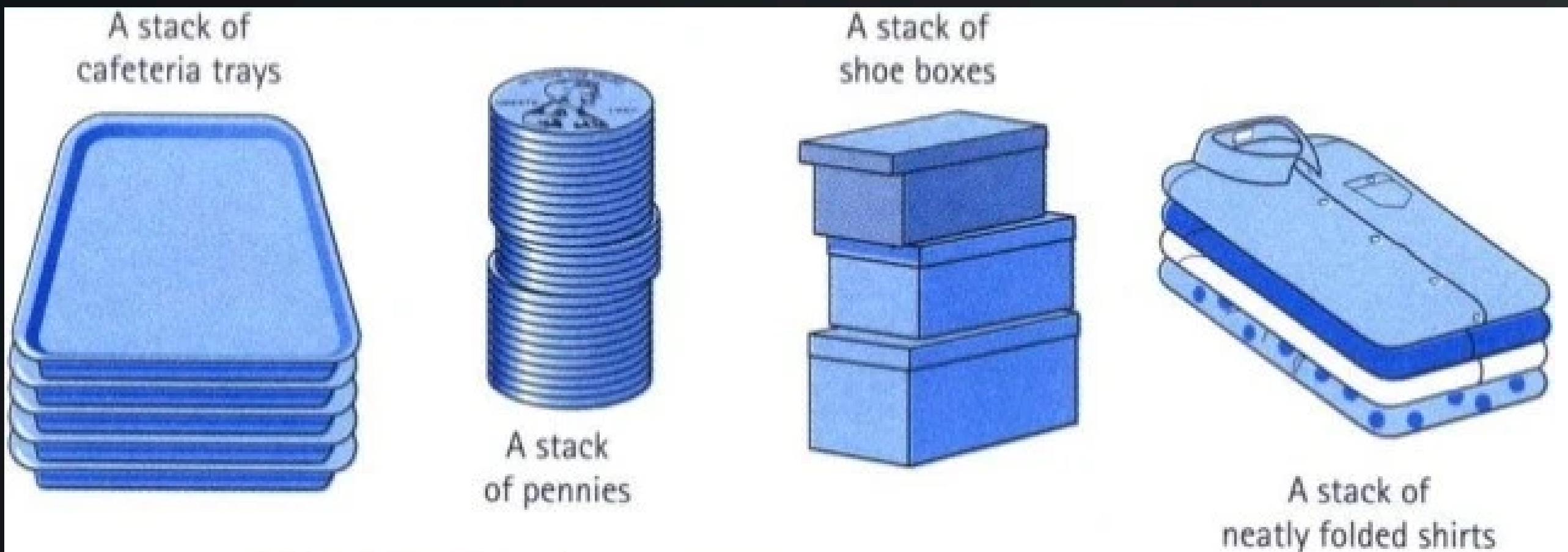
MEMORY

In Java memory is managed by JVM.
Stack and Heap are created by JVM and stored in RAM.



STACK DATA STRUCTURE

- The stack can be viewed similarly to a stack of books. You can add and remove books from the top of the stack but you can't access any in the middle or the bottom.
- The stack works on the Last In First Out (LIFO) principle. You can only ever read the data from the item on the top of the stack.



MEMORY

What is the stack?

The stack has 2 main responsibilities when your program is running:

1. To keep track of the method that control should return to once execution has finished for the current method.
2. To hold the values (or pointers) of local variables used in the methods.

How does the call stack work?

- Each time you call a method in your application it is added (pushed) to the call stack along with any local variables that are declared in the call stack.
- Once the execution of that method has finished it is removed (popped) from the call stack and execution is returned to the previous method.

PACKAGES

Java Packages

- A package in Java is used to group related classes.
- Think of it as a folder in a file directory. We use packages to avoid name conflicts, and to write a better maintainable code.

Packages are divided into two categories:

- Built-in Packages (packages from the Java API)
- User-defined Packages (create your own packages)

- Import a specific Class
 - `import java.util.Scanner;`
- Import a Package
 - `import java.util.*;`
 - `import java.lang.*;`

OBJECT CLASS IN JAVA

.....

- Object class in Java is present in `java.lang` package.
- Every class is directly or indirectly derived from the Object class. If a class does not extend any other class then it is a direct child class of the Java Object class and if it extends another class then it is indirectly derived.
- The Object class acts as a root of the inheritance hierarchy in any Java Program.

