Object Orientation & Object Creation

Guess Game

+ memory mapping

≥ Structural program Vs

→ Object Oriented

(um) (Gui) (pkg) (main)

(Class) —? Object
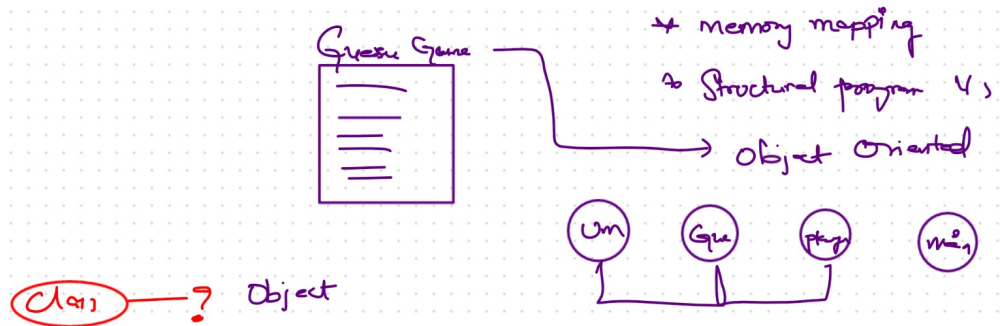
Wrapper Class

[introduced

Primitive types

byte b = 5;

Short s = 10;

int i = 1000;

Long l = 500 000 l;

float f = 569.32f

double d = 123 512.679;

char c = ' A "

boolean b = true

Wrapper Classes

Byte b = new Byte();

Short s = new Short()

Integer i = new Integer()

long l = new Long()
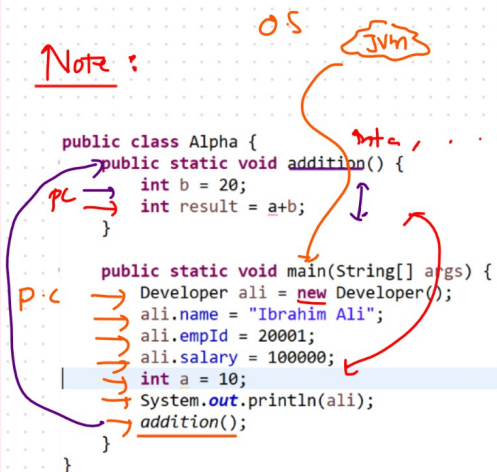
Float f = new Float()
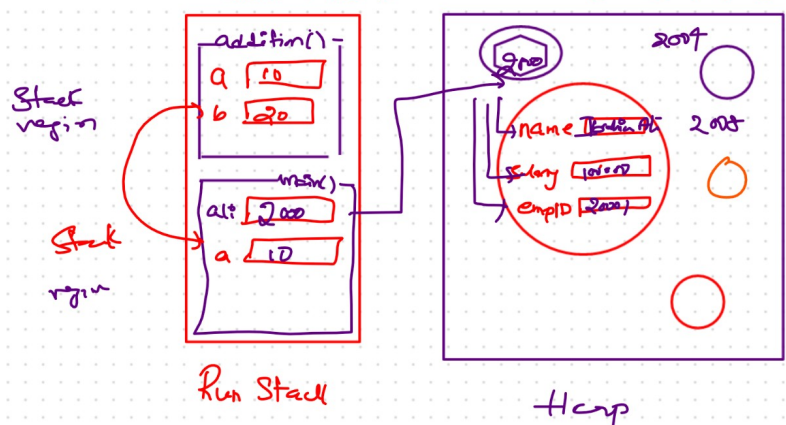
Double d = new Double()

Character c = new Character()

Boolean B = new Boolean()

Note :

OS   JVM

Note: object → Heap

primitive → Stack

types

```java
public class Alpha {
    public static void addition() {
        int b = 20;
        int result = a+b;
    }

    public static void main(String[] args) {
        Developer ali = new Developer();
        ali.name = "Ibrahim Ali";
        ali.empId = 20001;
        ali.salary = 100000;
        int a = 10;
        System.out.println(ali);
        addition();
    }
}
```

PC

Memory Mapping

Local Variables   Instance Variable

addition() —

a [10]

b [20]

Stack

region

main() —

al: [2000]

a [10]

Stack

region

Run Stack

2004

name [Ibrahim Ali] 2004

salary [100000]

empID [20001]

Heap

→ Java is not 100% Object Oriented Programming Language because there is a usage of **primitive data type**

→ we can make our project 100% Object Oriented by using "**Wrapper Classes**"

int a = 0   Integer a = new Integer(11)

Disadvantages

→ The use of wrapper classes occupies more memory