

Day 15

Value type

reference type

eg:-  
→ int

eg:-  
→ string.

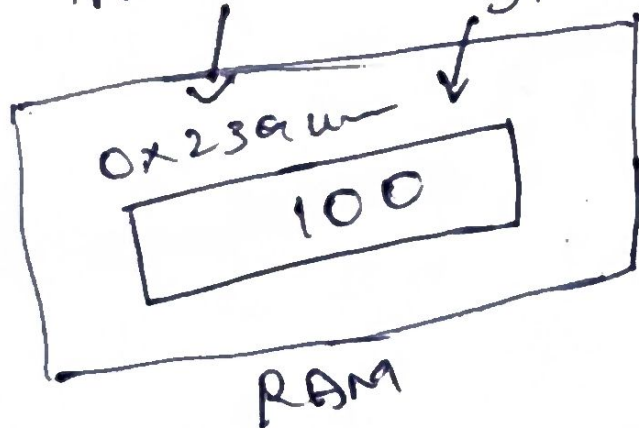
↓  
Immutable.

heap memory

Stack memory

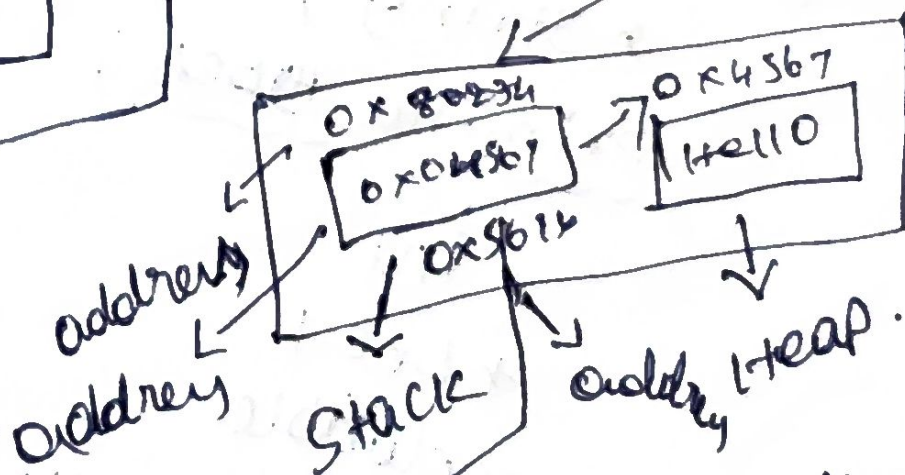
\* All the value type will be stored in stack memory.

eg:- `int i = 100;`

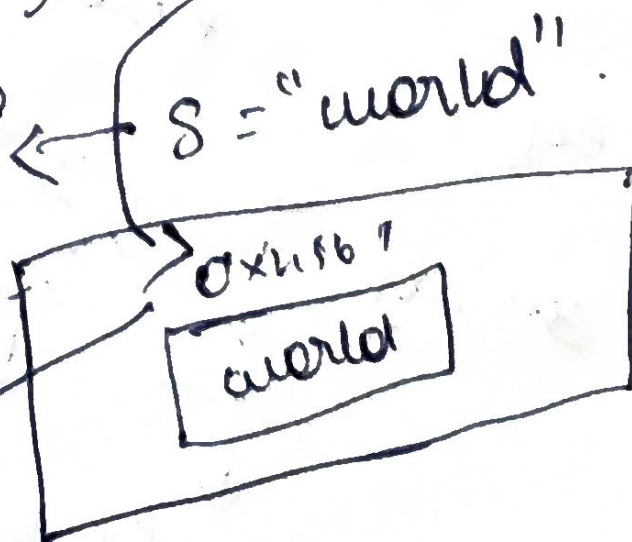


\* All the reference type will be stored in both heap & stack memory.

eg:-  
`String s = "Hello";`



It will create a new address.



mutable - data types that can be change after they are created

immutable - data type that cannot be change after they are created.

mutable eg:-

\* Array.

\* Lists

\* all the custom classes with public creators.

\* String Builder.

immutable eg:-

\* String,

primitive types:

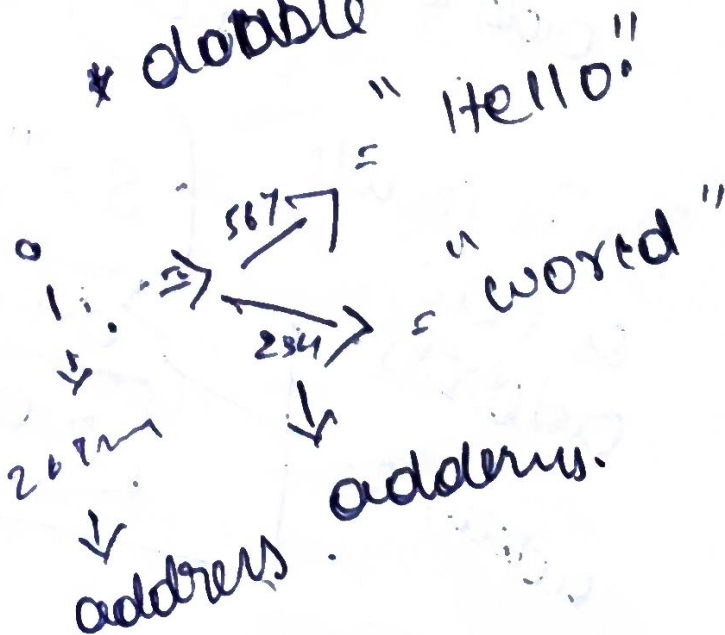
\* int

\* bool

\* double

eg

String





```
StringBuilder sb = new StringBuilder();  
String name = "Hello"  
sb.append(name);
```

↓  
add

```
String namevalue = "world";  
sb.append(namevalue);
```

output;

Hello world.

---

```
String Hi = "Hello {name}, How are {you?}";
```

```
Hi.replace("{name}", "Navindra");
```

```
Hi.replace("{you?}" , "me");
```

output:

Hello Navindra, How are me;

---

```
String Hi = String.format("Hello {0},
```

String how are {1?}

"Navin", "you");

output:

Hello Navindra, How are you

value in order of 0, 1, 2, 3

string hi = \$"Hello {name},  
How are {you} at  
the age {age} ).

string name = "Nauindra"  
string you = "me";  
string age = "2.1";