* Aggregation in java AOAPT

Laptop – Composite

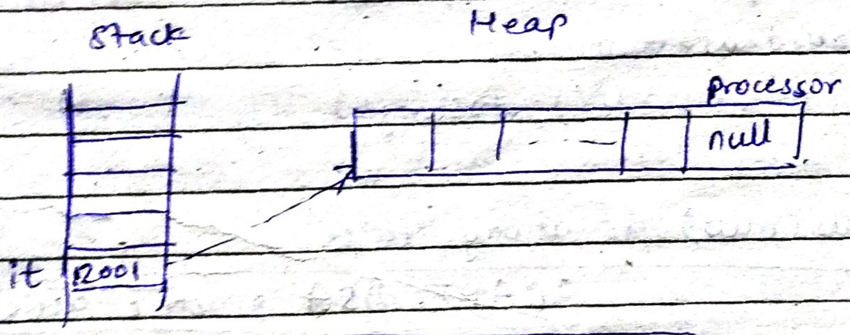
Processor – Constituent

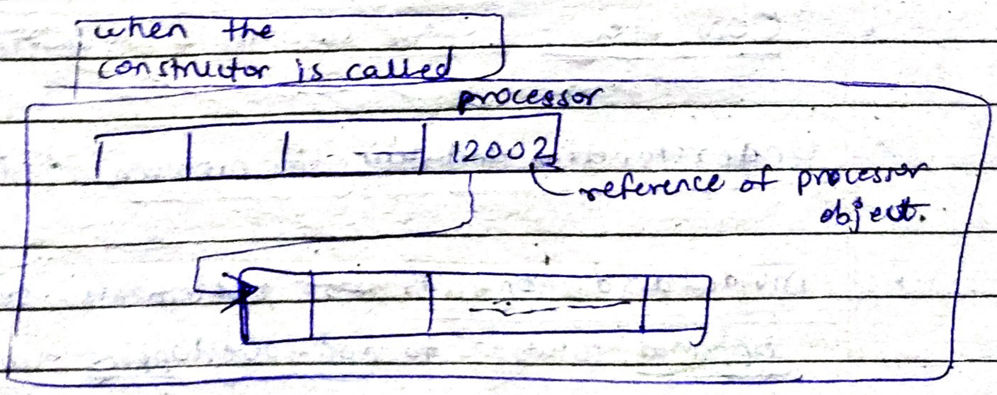
Aggregation is created by declaring a reference of constituent type in composite

```
public class Laptop {
    — ——
   —— ———

    private Processor processor;
    public Laptop () {
        // —— ——
        processor = new Processor();
        // ———
    }
    // setters/getters
}
```

```
Laptop lt = new Laptop ();
```





when the constructor is called
processor
12002
reference of processor object.

Laptop laptop = -----;
System.out.print(laptop.getProcessor().getModel());

Decorator method to act as delegate - to frequently access the members of constituent using reference of a composite.

To watch → 1) Garbage collection
2) cosmic class

Garbage collection

- 2 major problems - Memory leak, dangling pointer
- Garbage collection = process in which Java recollects the space occupied by dead/abandoned objects.

Garbage collector - allocates memory for new objects
- ensuring live objects remain in memory
- recovering memory used by dead objects

non-generational garbage collector
- Mark, sweep, ~~costa~~ compact

- java.lang.object has no superclass.
It's called as cosmic class
equals(), hashCode(), toString()

- x..equals(null) is always false

OOAPT DSA practice assignment

default equals () function

e1.equals(e1) → it checks the references of both
object e1 ↖ true

So, we've to override it and check the fields of these objects in order to compare them.

```
Public boolean equals (Object obj) {
      if (this == obj)
           return true;                    if both objects are
                                               of different classes
                                            ↓
      If ( Obj == null || this.getClass() != obj.getClass())
           return false;

      Employee emp = (Employee) obj;      // typecasting
      return this.name.equals(emp.name) && this.age == emp.age;
}
```

- if hashcodes are different, objects' fields are not equal
  if hashcodes are equal, equals() can be used to check equality

```
public int hashCode()
{ ——
}
```

Set, map मध्ये objects add करताना hashCode आणि equals override करू शकतो.