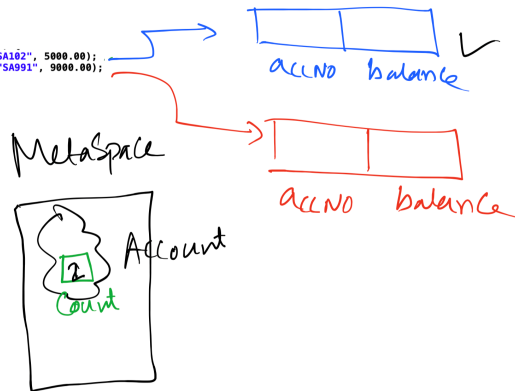


```
Account rahulAcc = new Account("SA102", 5000.00);
Account swethaAcc = new Account("SA991", 9000.00);
```

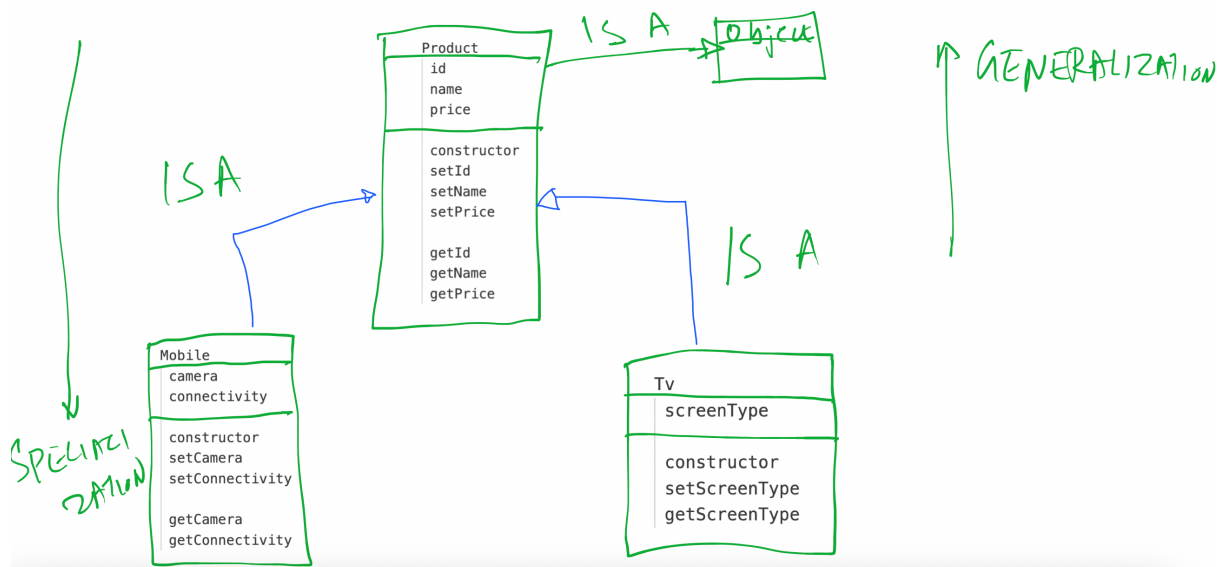
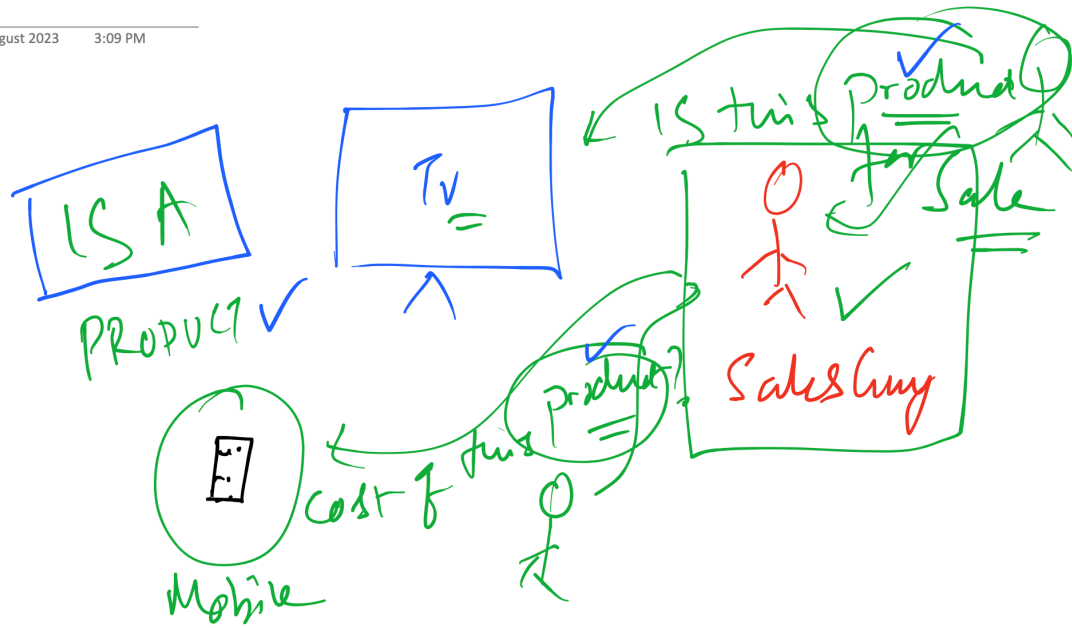


Mobile.java
id
name
price
connectivity
camera
constructors
setId
setName
setPrice
setConnectivity
setCamera
getId
getName
getPrice
getConnectivity
getCamera

Tv.java
id
name
price
screenType
smartTv
constructors
setId
setName
setPrice
setScreenType
setSmartTv
getId
getName
getPrice
getScreenType
isSmartTv()

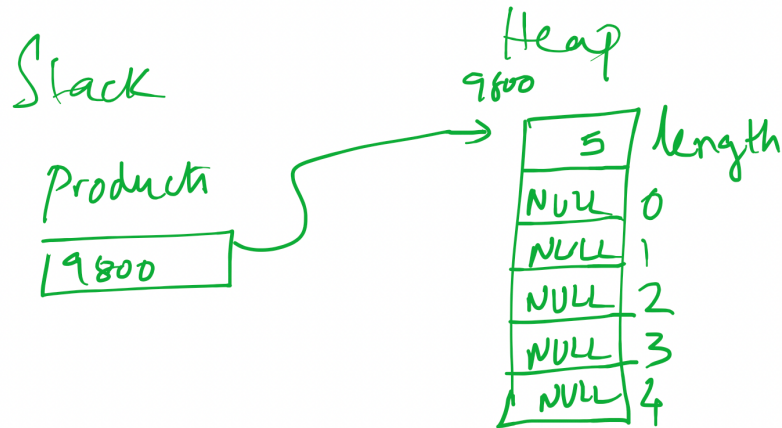
DRY

Copy & Paste



`Product[] products = new Product[5];`

Array of Pointers



```
// OCP
private static void printDetailsOCP(Product[] products) {
    for(Product p : products) {
        System.out.println("###");
        Method[] methods = p.getClass().getMethods(); // methods + inherited methods
        for(Method m : methods) {
            if(m.getName().startsWith("get")) {
                try {
                    Object ret = m.invoke(p); // invoke "method -> m" on "context p"
                    System.out.println(m.getName().substring(3).toUpperCase() + " : " + ret);
                } catch (Exception ex) {
                    ex.printStackTrace();
                }
            }
        }
        System.out.println("-----");
    }
}
```

```
public int getId() {
    return id;
}

public void setId(int id) {
    this.id = id;
}

public String getName() {
    return name;
}

public void setName(String name) {
    this.name = name;
}

public double getPrice() {
    return price;
}

public void setPrice(double price) {
    this.price = price;
}

public boolean isExpensive() {
    return false;
}

public String getScreenType() {
    return screenType;
}

public void setScreenType(String screenType) {
    this.screenType = screenType;
}
```