

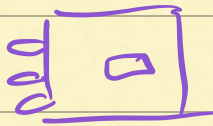
Agenda :

1. Adapter
 2. Facade
 3. Decorator
 4. Flyweight
- } structural design patterns

Adapter Design Pattern :

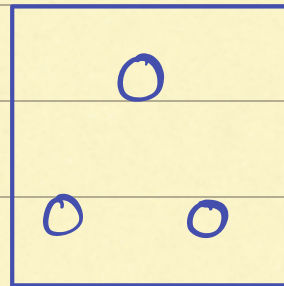
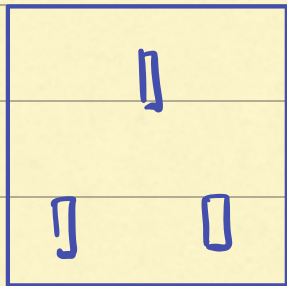
↳ what is an adapter?

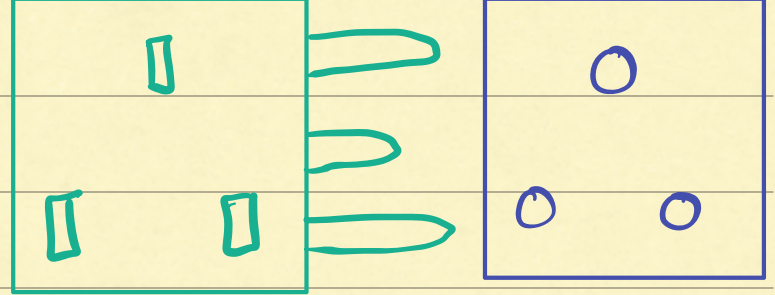
Power Charges



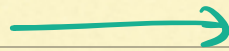
helps to adapt.

type C → USB → ☐
type C → type C → ☐
type C → lightning → ☐
type C → HDMI → ☐



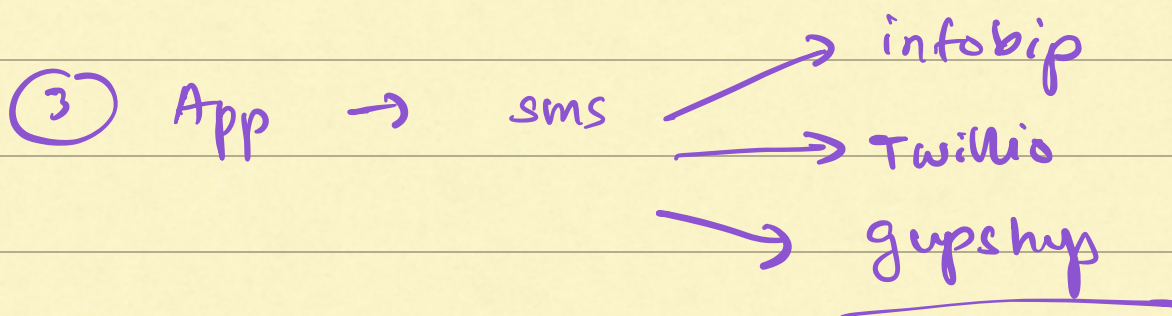
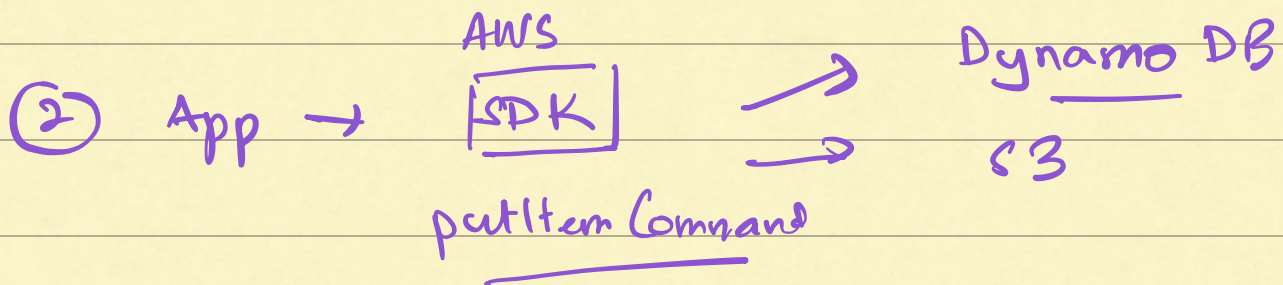
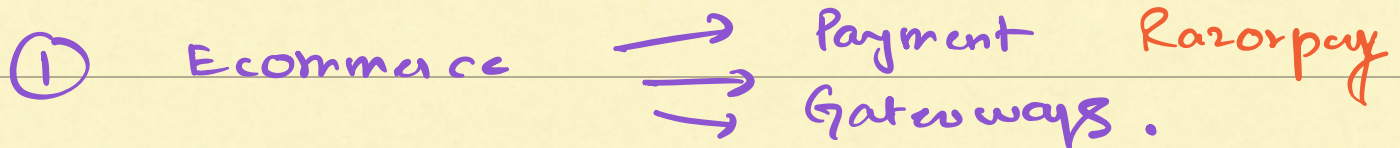


s/w Applications



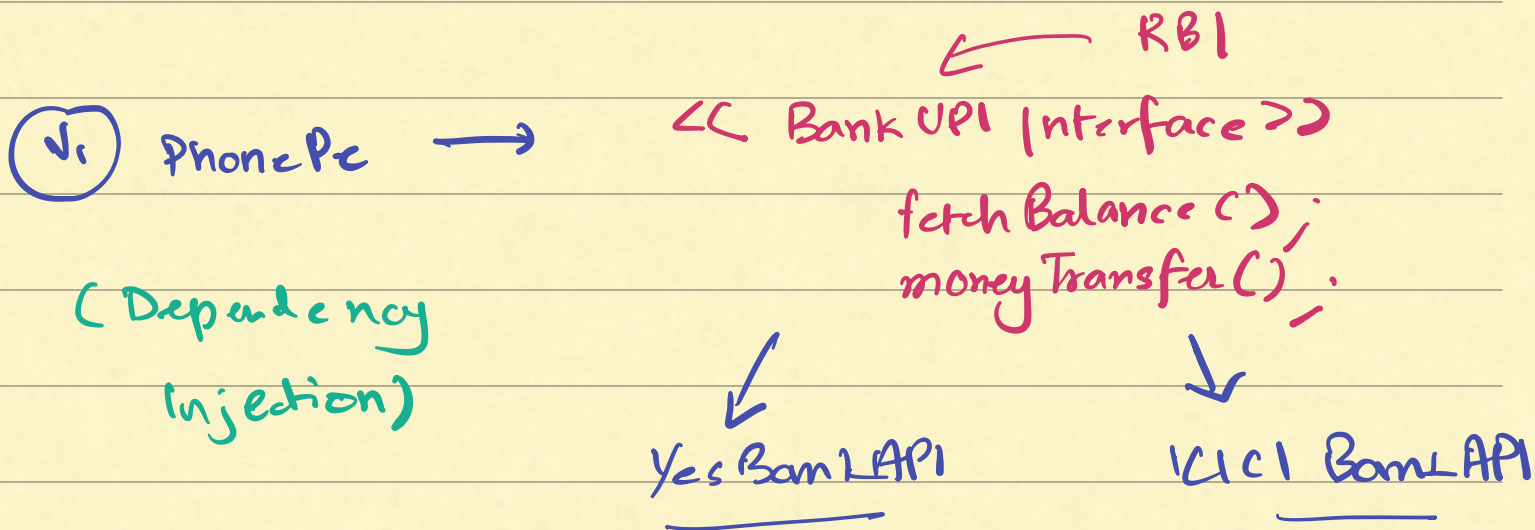
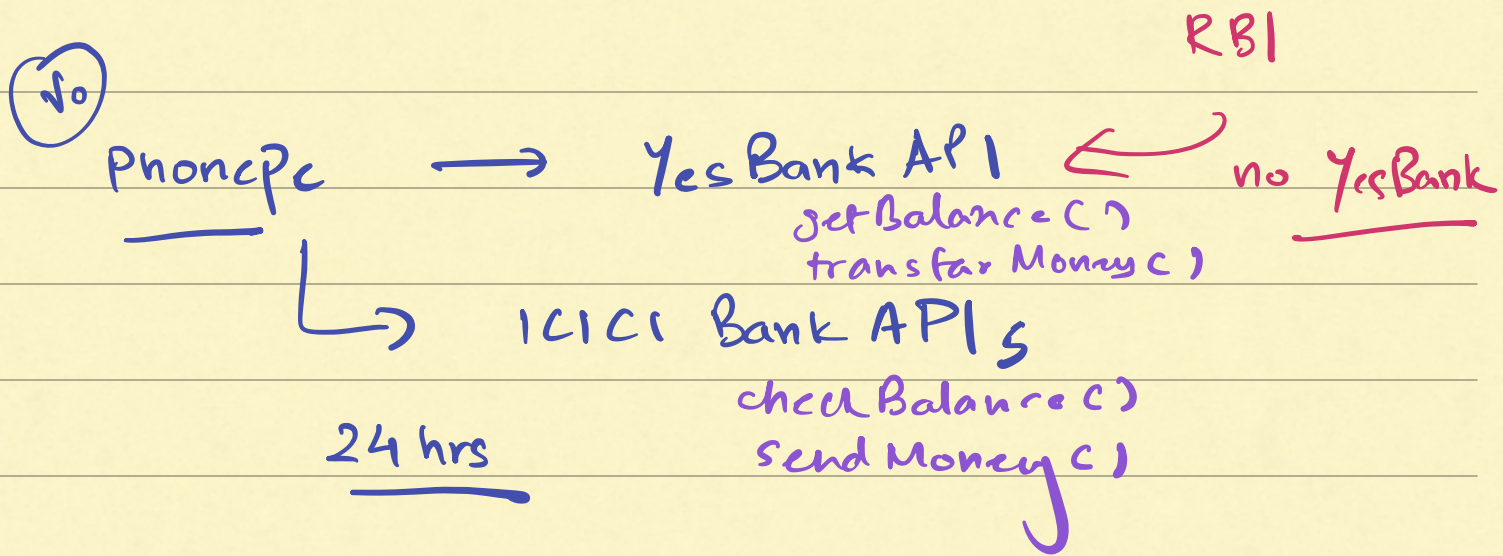
libraries, SDKs

Example 1



Problems of these 3rd Party Softwares:

1. Changing these 3rd Party apps.
2. Service can go out of maintainence.
3. Service Down time.



(v2)

PhonePe



<< UPI Interface >>

→ fetchBalance()

→ moneyTransfer()

take input
acc to PhonePe



fetchBalance()

→ transfer input
to Yes Bank's
need.

Yes Bank API
Adapter

(Yes Bank API)

implements



ICICI Bank API
Adapter

(ICICI Bank
API)

→ use Yes Bank API

→ transform Yes Bank
output to PhonePe need.

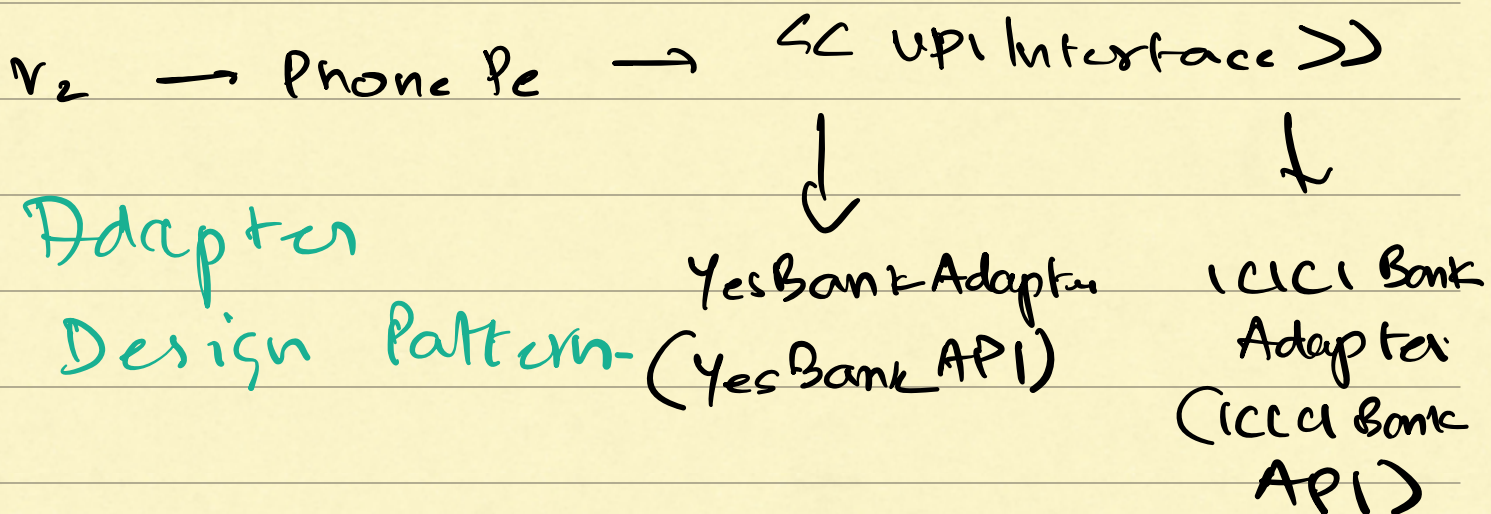
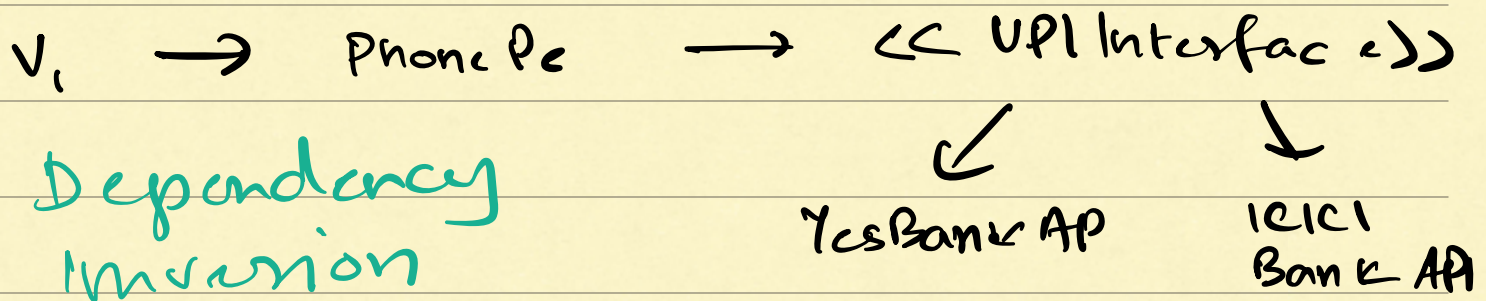
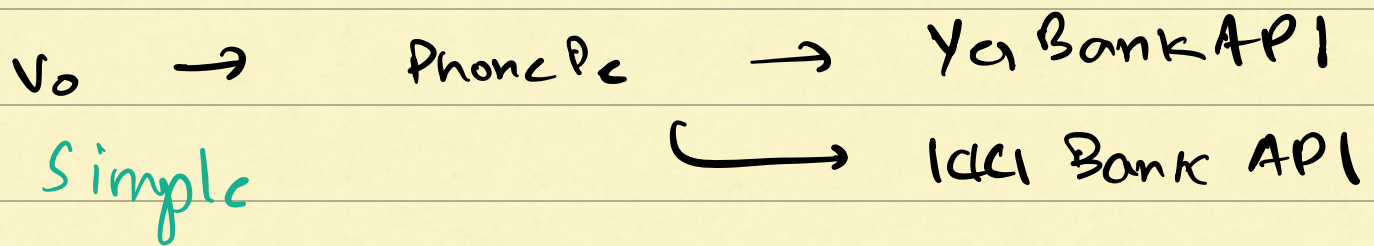
→ return output

}



return output
acc to phonePe

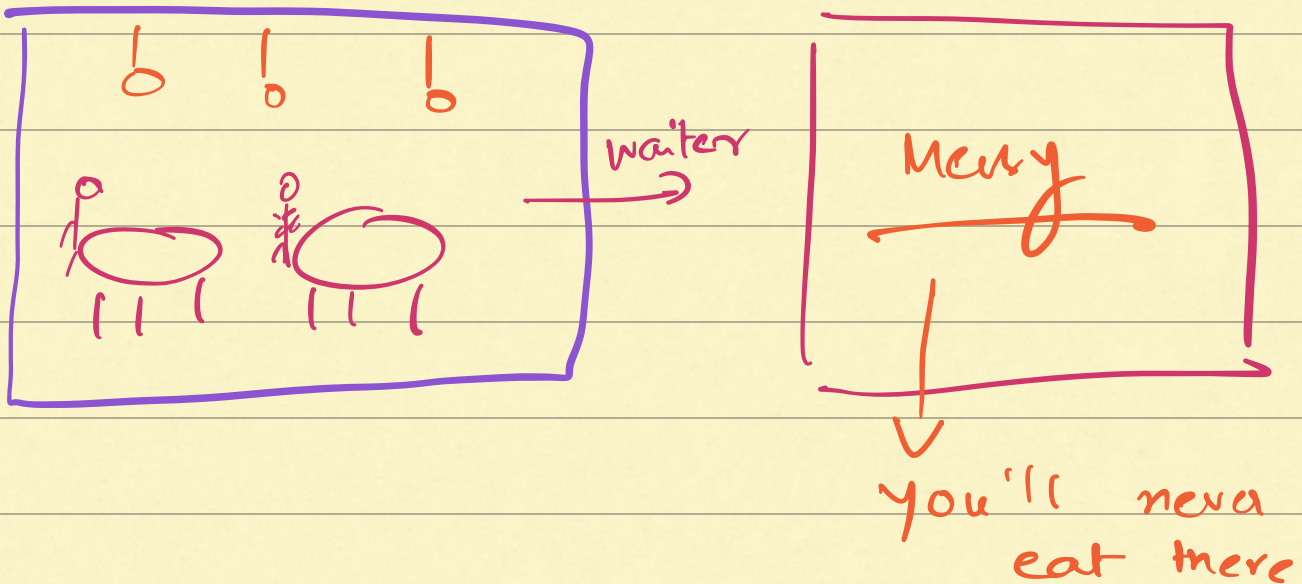
Diagrammatic View :



Facade Design Pattern

↳ Front part / outside of building
(Visible)

Restaurant :



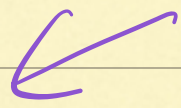
Amazon

Place an order

place Order () {

Order.save("u-u");

or



- Update Inventory
- Notify the seller
- Notify the buyer
- Notify logistics
- Send Invoice Email

>

↓

```
place Order ( ) {
```

```
    Order.save();
```

```
    afterOrder("And He")
```

```
}
```

after Order () {

```
    updateInv();
```

```
    notify();
```

```
    sendEmail();
```

```
}
```