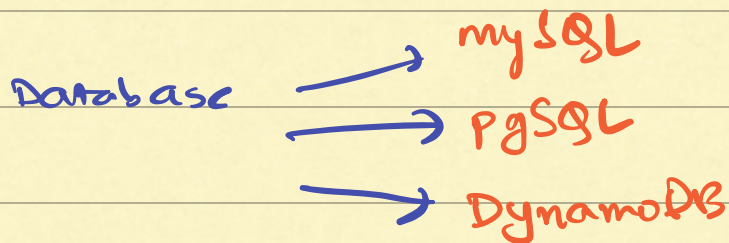


Agenda

Factory Method

Abstract Factory



```
Database db = new MySQL();
```

```
void createStudent ( Database db ) {
```

```
    if ( db is instance of MySQL ) {
```

```
        Query q = new MySQLQuery();
```

```
    }
```

```
    elseif ( db is instance of PostgreSQL ) {
```

```
        Query q = new PostgreSQLCreateQuery();
```

```
    }
```

```
        :
```

```
        q.execute();
```

```
    }
```

SRP + OCP

~~Abstract~~

<<Database>>

<<createQuery>>

createQuery()

MySQL

PgSQL

Dynamo

MySQLQuery()

PgSQLQuery

- createQuery

- createQuery

- createQuery

mysql {

createQuery() {

return new mysqlQuery()

}

class pgSQL {

createQuery() {

return new pgSQLQuery()

}

class UserService {

Database db = new mysql();

void createStudent (Database db)

{
Query q = db.createQuery();
q.execute();

}

}

class Bird

creat = FBC()

Sparrow

Crow

Penguin

<Flying Bird>

Low()

High()

class Sparrow {

Flying Bird
create FBC() {

new LowFlying();

}

createFB

createFB.

Transaction T = db.createTransaction();

g1.execute(T);

g2.execute(T);

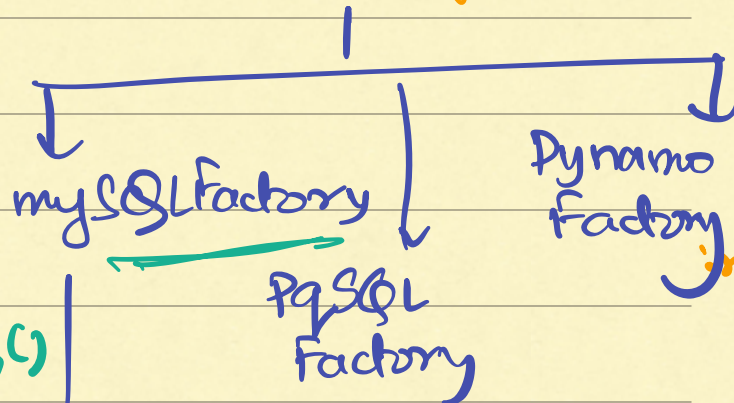
<< Database >>

create Query();
create Transaction();
create Updator();
connect();
increase Pool Size();
close Connection();
time Out();

<< Database Component Factory >>

creatQuery();
create Transaction();
create Updator();

won't exist after solution



create release
create Trans
create Updator

create DBComp Factory();
↓
mysql
↓
create DB comp Factory();
return new
mysqlFactory();
}

createQuery();
return new
mysqlCreateQuery();
}
create Transaction();
return new
mysqlTransaction();



class PgSQL {

create DBCompFactory() {

return new PgSQLFactory();

}

}

Flutter / React Native

↳ UI Framework

→ cross-platform

Button



abstract Platform

createButton;

Android

- createButton() {

return new

AndroidButton();

}

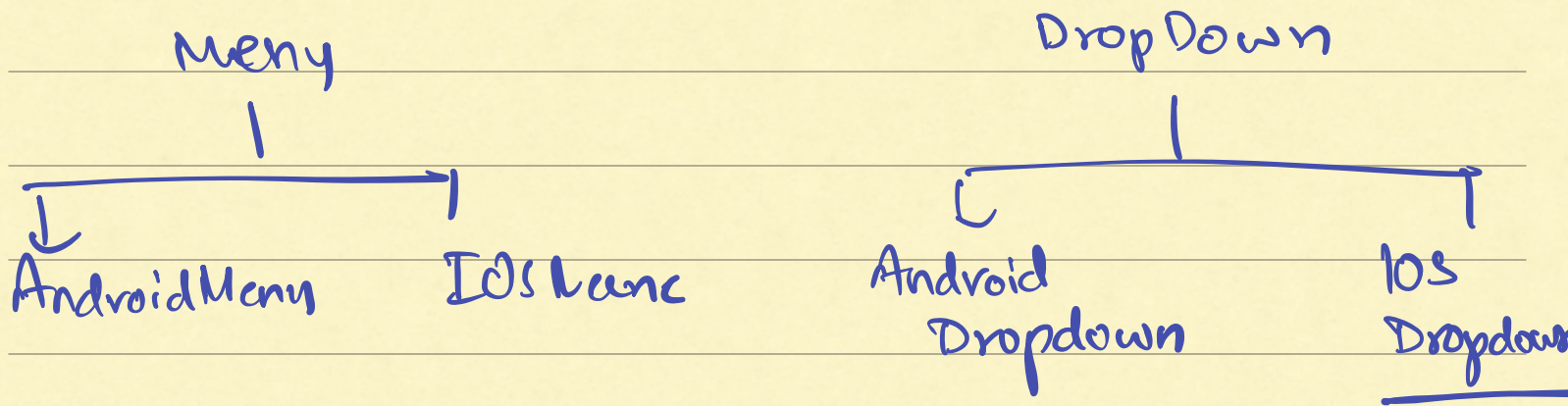
iOS

- createButton() {

return new

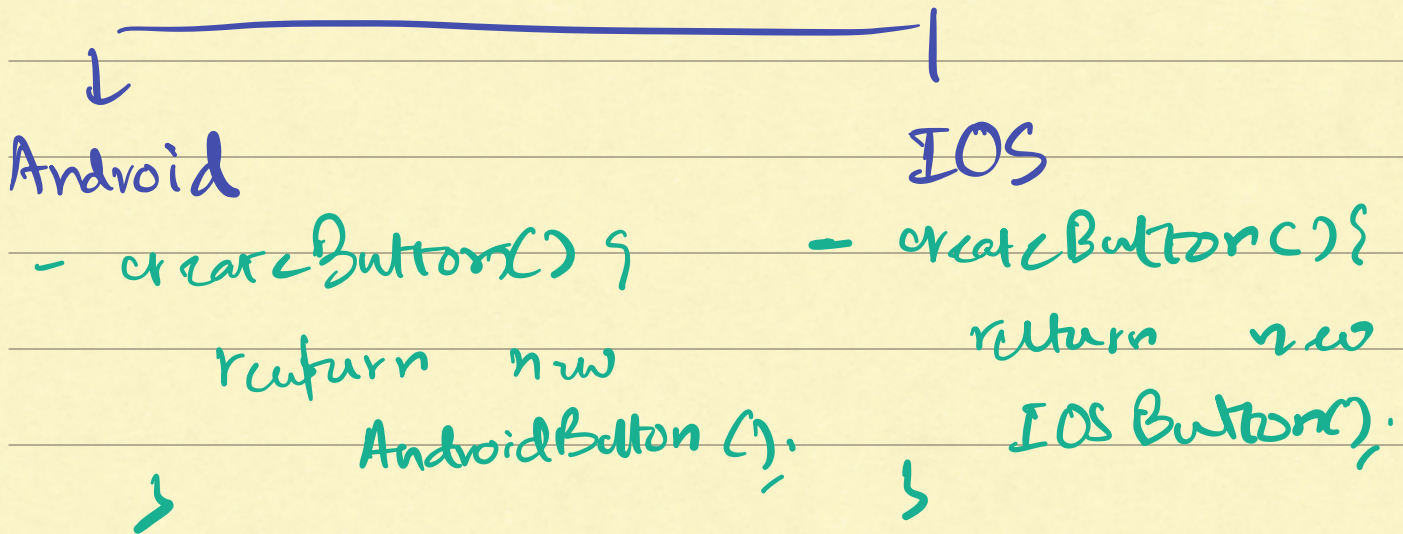
iOSButton();

}



abstract Platform

```
create Button;  
create Menu;  
create Dropdown;
```



Platform

<<UI Component Factory>>

abstract createComponent
Factory();

- createButton();

- createMenu();

- createDropdown();

Android

iOS

Android Comp Factory

iOS Comp Factory

createComponentFactory();

return new

AndroidCompFactory();

}

- createButton();

return new

AndroidButton();

- createMenu();

- createDropdown();

}