OOPs Advanced Concepts

1) Inheritance

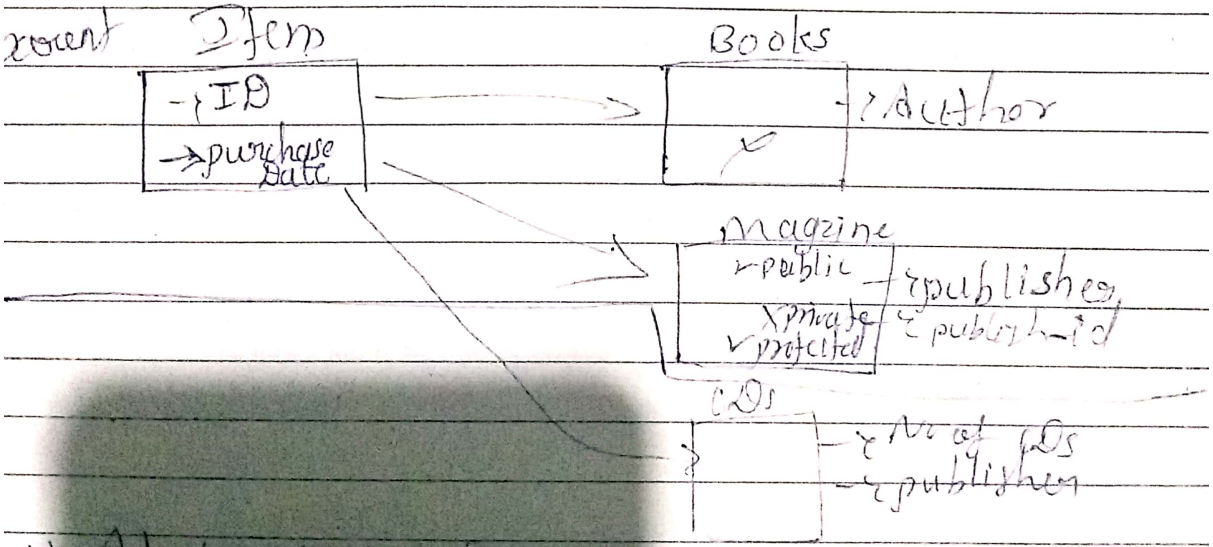
- Parent class, child class
- access modifiers
- Method overloading
- Multilevel inheritance

2) Abstract class

3) Interfaces

1) Inheritance :

- It allows one class (child) to inheritance properties and behaviours of another class (parent)
  - Imagine a Library is a parent class,



2) Abstract class

only Declaration ~~no~~ definition for simple function.

```
abstract class Vehicle {
    abstract void start();   // abstract (must override)

    void fuelType() {   // concrete (already implemented)
        System.out.println("Petrol");
    }
}

class Car extends Vehicle {
    void start() {
        System.out.println("Car started");
    }
}
```

python:

```
from abc import ABC, abstractmethod

class Vehicle(ABC):

    @abstractmethod
    def start(self):   # abstract (must override)
        pass

    def fuel_type(self):   # concrete
        return "Petrol"   # default value

class Car(Vehicle):
    def start(self):
        print("Car started")
```

2) Interfaces

-> only declaration, no definition (implementation) of functions or method

-> variables are final means constant can't change after declare

Ex: Java

```
interface PaymentGateway {
    int Max_Limit = 10000;
    // public static final by default
}

class Paypal implements PaymentGateway {

    void updateLimit() {
        // Max_Limit = 20000;
        // X compile-time error (can't modify)
    }
}
```

we cannot create objects of abstract and interface class directly

-> we use child class of that abstract and interfaces classes to create objects

-> we can call child class objects

11.25

interface DbAccessLayer {

void connect();
void createTable();
void insert();
// only declaration

}

class SdLite extends DbAccessLayer {

void connect() {
// operations
}

void createTable() {
}

void insert() {
}
// must implements all methods
in subclass

}

}