**Inheritance:**

class Animal {

String name;

public void eat() {

System.out.println("I can eat");

}

}

class Dog extends Animal {

public void display() {

System.out.println("My name is " + name);

}

}

class Main {

public static void main(String[] args) {

Dog labrador = new Dog();

labrador.name = "Rohu";

labrador.display();

labrador.eat();

}

}

**Abstraction:**

abstract class Car{

abstract void accelerate();

}

class Suzuki extends Car{

void accelerate(){

System.out.println("Suzuki::accelerate");

}

}

class Main{

public static void main(String args[]){

Car obj = new Suzuki(); //Car object =>contents of Suzuki

obj.accelerate(); //call the method

}

}

**Interface:**

interface Polygon\_Shape {

void calculateArea(int length, int breadth);

}

class Rectangle implements Polygon\_Shape {

public void calculateArea(int length, int breadth) {

System.out.println("The area of the rectangle is " + (length \* breadth));

}

}

class Main {

public static void main(String[] args) {

Rectangle rect = new Rectangle();

rect.calculateArea(10, 20);

}

}

**Association:**

class Account {

private String bank\_name;

private long Account\_number;

Account(String bank\_name, long Account\_number) {

this.bank\_name = bank\_name;

this.Account\_number = Account\_number;

}

public String getBankName() {

returnthis.bank\_name;

}

public long getAccountNumber() {

returnthis.Account\_number;

}

}

class Employee

{

private String emp\_name;

Employee(String emp\_name)

{

this.emp\_name = emp\_name;

}

public String getEmployeeName()

{

return this.emp\_name;

}

}

class Main

{

public static void main (String[] args)

{

Employee emp = new Employee("Andrew");

Account acc = new Account("Citi Bank", 13319);

System.out.println(emp.getEmployeeName() +

" has an account with " + acc.getBankName() + " with Account Number:"

+ acc.getAccountNumber());

}

}

**Composition:**

class CarEngine {

public void startEngine(){

System.out.println("Car Engine Started.");

}

public void stopEngine(){

System.out.println("Car Engine Stopped.");

}

}

class Car {

private String color;

private int max\_Speed;

public void carDetails(){

System.out.println("Car Color= "+color + "; Max Speed= " + max\_Speed);

}

public void setColor(String color) {

this.color = color;

}

public void setMaxSpeed(int max\_Speed) {

this.max\_Speed = max\_Speed;

}

}

class Honda extends Car{

public void HondaStart(){

CarEngine Honda\_Engine = new CarEngine();

Honda\_Engine.startEngine();

}

}

public class Main {

public static void main(String[] args) {

Honda HondaCity = new Honda();

HondaCity.setColor("Silver");

HondaCity.setMaxSpeed(180);

HondaCity.carDetails();

HondaCity.HondaStart();

}

}

**Agregation:**

class Author

{

String authorName;

int age;

String place;

// Author class constructor

Author(String name, int age, String place)

{

this.authorName = name;

this.age = age;

this.place = place;

}

}

class Book

{

String name;

int price;

// author details

Author auther;

Book(String n, int p, Author auther)

{

this.name = n;

this.price = p;

this.auther = auther;

}

public static void main(String[] args) {

Author auther = new Author("John", 42, "USA");

Book b = new Book("Java for Begginer", 800, auther);

System.out.println("Book Name: "+b.name);

System.out.println("Book Price: "+b.price);

System.out.println("------------Auther Details----------");

System.out.println("Auther Name: "+b.auther.authorName);

System.out.println("Auther Age: "+b.auther.age);

System.out.println("Auther place: "+b.auther.place);

}

}