interface 인터페이스이름{

public static final 타입 상수이름 = 값;

public abstract 메서드이름(매개변수목록);

}

인터페이스에서 포함될 수 있는 것들 :: 상수/메서드원형

두가지 멤버 변수 :: public static final/public abstact

부모클래스 :: 하나만 상속

extends >하나만 상속

interface >다중 상속

인터페이스의 확장

상위개념의 인터페이스를 하위 인터페이스에서 상속 받음

상속을 위한 키워드 :: extends

**interface** Sports {

**public** **void** setHomeTeam(String name);

**public** **void** setVisitingTeam(String name);

}

**interface** Football **extends** Sports {

**public** **void** homeTeamScored(**int** points);

**public** **void** visitingTeamScored(**int** points);

**public** **void** endOfQuarter(**int** quarter);

}

**interface** Hockey **extends** Sports {

**public** **void** homeGoalScored();

**public** **void** visitingGoalScored();

**public** **void** endOfPeriod(**int** period);

**public** **void** overtimePeriod(**int** ot);

}

인터페이스 다중상속

문제1>

인터페이스를 사용하여 계산기 프로그램을 만드시오

답)

**package** calculator;

@Remotable

**public** **interface** AddService {

**double** add(**double** n1, **double** n2);

}

**package** calculator;

**public** **class** AddServiceImpl **implements** AddService {

**public** **double** add(**double** n1, **double** n2) {

System.***out***.println("AddService - add " + n1 + " and " + n2);

**return** n1 + n2;

}

}

**public** **interface** CalculatorService {

**double** add(**double** n1, **double** n2);

**double** subtract(**double** n1, **double** n2);

**double** multiply(**double** n1, **double** n2);

**double** divide(**double** n1, **double** n2);

}

**package** calculator;

**public** **class** CalculatorServiceImpl **implements** CalculatorService {

@Reference

**private** AddService addService;

@Reference

**private** SubtractService subtractService;

@Reference

**private** MultiplyService multiplyService;

@Reference

**private** DivideService divideService;

**public** **double** add(**double** n1, **double** n2) {

**return** addService.add(n1, n2);

}

**public** **double** subtract(**double** n1, **double** n2) {

**return** subtractService.subtract(n1, n2);

}

**public** **double** multiply(**double** n1, **double** n2) {

**return** multiplyService.multiply(n1, n2);

}

**public** **double** divide(**double** n1, **double** n2) {

**return** divideService.divide(n1, n2);

}

}

문제2>

원의 넓이와 둘레를 구하는 프로그램이다

interface Shape을 작성하라

**public** **class** Circle **implements** Shape {

**static** **double** *radius*;

**public** Circle(**double** radius) {

**this**.*radius* = radius;

}

**public** **double** getRadius(**double** radius) {

**return** radius;

}

**public** **void** setRadius(**double** radius) {

**this**.*radius* = radius;

}

**public** **double** Area() {

**return** ***PI*** \* *radius* \* *radius*; // 원의 넓이 구하는 공식

}

**public** **double** Perimeter() {

**return** 2 \* ***PI*** \* *radius*; // 원의 들레 구하는 공식

}

}

**import** java.util.Scanner;

**public** **class** Main {

**public** **static** **void** main(String[] args) {

Circle circle= **new** Circle(3.1);

Scanner scan = **new** Scanner(System.***in***);

System.***out***.println("원의 반지름을 입력");

Circle.*radius*=scan.nextInt();

System.***out***.println("원의 넓이 : "+circle.Area()+" 둘레 : "+ circle.Perimeter());

}

}

답>

**public** **interface** Shape {

**final** **double** ***PI***=3.14159;

**public** **double** Area(**double** area);

**public** **double** Perimeter(**double** perimeter);

}