**12주차\_실습과제(학번: 20157135이름:이상우)**

* 클래스 – 필드와 메소드로 구성

public class Student{

String name; /**/이름을 저장하기 위한 필드 선언**

int java, eng; **//과목을 저장하기 위한 필드 선언**

public int hap(){ **//과목 합을 계산하여 반환하는 메소드**

return java+eng;

}

public String toString(){ **//객체 내용(이름, 과목성적, 합)을 문자열로 반환하는 메소드**

String str="이름 : “ + name;

str += "\njava : " + java + "\teng : " + eng;

str += "\n과목합 : " + hap();

}

}

/**/Student 클래스에 대한 객체를 생성하고 테스트하는 클래스**

public class Exam01 {

public static void main(String[] args) {

**//클래스의 필드와 메소드는 객체를 통해서 접근 해야 함**

Student obj=new Student(); **//Student 클래스 객체 obj 선언 & 생성**

obj.name="김낙준"; **//name 필드 초기화**

obj.java=90; **//java 필드 초기화**

obj.eng=98; **//eng 필드 초기화**

System.*out*.println(obj); **//객체 내용 출력**

}

}

|  |
| --- |
| **[실행결과]** |

* 프로그램과제 1 : 은행 계좌를 처리하기 위한 클래스를 작성하고 테스트하는 프로그램이다. 밑줄 친 부분을 채워 넣은 후 실행결과를 제시하시오

**//은행 계좌 처리를 위한 클래스**

**public** **class** Account {

**//이름을 저장하기 위한 필드 name 선언**

**//잔액을 저장하기 위한 필드 balance 선언(4byte 정수형)**

**//형식 매개변수로 받은 값만큼 잔액을 증가하는 메소드, 반환값 없음** - 본인작성

**void** dePosit(**int** money){

}

**//형식 매개변수로 받은 값만큼 잔액을 감소하는 메소드, 반환값 없음 – 본인작성**

**void** withDraw(**int** money){

}

**public** int getBalance(){ **//필드 balance 값을 반환하는 메소드**

**return** balance;

}

**public** String toString(){ **//객체 내용(이름과 잔액)을 문자열로 반환**

**return** "이름 : " + name + "\t잔액 : " + balance;

}

}

**//Account 클래스에 대한 객체를 생성하고 테스트 하는 클래스**

**public** **class** AccountTest {

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

**//Account 클래스 객체 obj 선언 & 생성 – 본인작성**

obj.name = "길동이"; **// 이름 필드 초기화**

**//잔액 필드를 20000으로 초기화 – 본인작성**

**//dePosit()메소드 호출, 10000을 매개변수로 전달- 본인작성**

System.*out*.println("입금 후 잔액 : " + obj.getBalance());

/**/withDraw() 메소드 호출, 5000을 매개변수로 전달- 본인작성**

System.*out*.println("출금 후 잔액 : " + obj.getBalance());

System.*out*.println(obj); **//객체 내용 출력**

}

}

|  |
| --- |
| **[프로그램 소스]**  **class** Account {  String name; //이름을 저장하기 위한 필드 name 선언  **int** balance; //잔액을 저장하기 위한 필드 balance 선언(4byte 정수형)    //형식 매개변수로 받은 값만큼 잔액을 증가하는 메소드, 반환값 없음 - 본인작성  **void** dePosit(**int** money){  balance+=money;  }  //형식 매개변수로 받은 값만큼 잔액을 감소하는 메소드, 반환값 없음 – 본인작성  **void** withDraw(**int** money){  balance-=money;  }  **public** **int** getBalance(){ //필드 balance 값을 반환하는 메소드  **return** balance;  }  **public** String toString(){ //객체 내용(이름과 잔액)을 문자열로 반환  **return** "이름 : " + name + "\t잔액 : " + balance;  }  }  **public** **class** test02 {  **public** **static** **void** main(String[] args) {  Account obj = **new** Account(); //Account 클래스 객체 obj 선언 & 생성 – 본인작성  obj.name = "길동이"; // 이름 필드 초기화  obj.balance=20000; //잔액 필드를 20000으로 초기화 – 본인작성  obj.dePosit(10000); //dePosit()메소드 호출, 10000을 매개변수로 전달- 본인작성    System.***out***.println("입금 후 잔액 : " + obj.getBalance());  obj.withDraw(5000); //withDraw() 메소드 호출, 5000을 매개변수로 전달- 본인작성    System.***out***.println("출금 후 잔액 : " + obj.getBalance());  System.***out***.println(obj); //객체 내용 출력  }  } |
| **[실행결과]** |

* 프로그래밍 과제 2 : 사각형 넓이를 계산하는 Rectangle 클래스를 작성하고 테스트하시오

**//사각형 넓이를 계산하기 위한 클래스**

**public** **class** Rectangle {

**//가로와 세로 값을 저장하기 위한 필드 선언, double형으로 할 것-본인작성**

**//사각형 넓이를 계산하여 반환하는 메소드 getArea()-본인작성**

**public** String toString(){ **//객체 내용(가로, 세로, 넓이)을 문자열로 반환-본인작성**

}

}

**//Rectangle 클래스의 객체를 생성하고 테스트하는 클래스**

**import** java.util.\*;

**public** **class** RectangleTest {

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

**//Rectangle 클래스 rec 선언 & 생성**

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

System.*out*.println("가로를 입력하세요");

**//입력받은 값을 가로필드로 초기화**

System.*out*.println("세로를 입력하세요");

**//입력받은 값을 세로필드로 초기화**

**//객체 내용 출력**

}

}

|  |
| --- |
| **[프로그램 소스]**  **import** java.util.\*;  **class** Rectangle {  //가로와 세로 값을 저장하기 위한 필드 선언, double형으로 할 것-본인작성  **double** wa,ha;  //사각형 넓이를 계산하여 반환하는 메소드 getArea()-본인작성  **public** **double** getArea(){  **return** wa \* ha ;  }    **public** String toString(){ //객체 내용(가로, 세로, 넓이)을 문자열로 반환-본인작성  String str;  **return** "가로 :"+wa + "세로 :"+ha + "넓이";  }  }  **public** **class** test03 {  **public** **static** **void** main(String[] args) {  //Rectangle 클래스 rec 선언 & 생성  Rectangle rec = **new** Rectangle();  Scanner key=**new** Scanner(System.***in***);  System.***out***.println("가로를 입력하세요");  rec.wa=key.nextDouble();  System.***out***.println("세로를 입력하세요");  rec.ha=key.nextDouble();  //입력받은 값을 세로필드로 초기화  //객체 내용 출력  System.***out***.print(rec.toString()+rec.getArea());  }  } |
| **[실행결과]** |

* 프로그램 과제 3 : 다음과 같은 필드와 메소드로 구성되는 Box 클래스를 작성하고 테스트 하시오

class Box{

//핃드구성 : 가로(int), 세로(int), 높이(int), 부피(int)

//volume() 메소드 : 부피를 계산하여 필드에 저장, 매개변수와 반환값 없음

//toString() 메소드 : 객체 내용(가로, 세로, 높이, 부피)을 문자열로 반환, 매개변수 없음

}

public class BoxTest {

public static void main(String[] args) {

//Box 클래스 객체 선언 & 생성 – 본인 작성

//입력 받은 값으로 가로, 세로, 높이 필드 초기화 – 본인작성

//volume() 메소드 호출 – 본인작성

//객체 내용 출력 – 본인작성

}

}

|  |
| --- |
| **[프로그램 소스]**  **import** java.util.\*;  **class** Box{  **int** ga , sa , ha , vo;  //핃드구성 : 가로(int), 세로(int), 높이(int), 부피(int)  **public** **void** volume(){  vo=ga\*sa\*ha;  }  //volume() 메소드 : 부피를 계산하여 필드에 저장, 매개변수와 반환값 없음  //toString() 메소드 : 객체 내용(가로, 세로, 높이, 부피)을 문자열로 반환, 매개변수 없음  **public** String toString(){  String str;  str="가로"+ga +"세로"+sa +"높이"+ ha+"부피";  **return** str;  }  }  **public** **class** test04 {  **public** **static** **void** main(String[] args) {  Scanner key=**new** Scanner(System.***in***);  Box obj =**new** Box();  //Box 클래스 객체 선언 & 생성 – 본인 작성  System.***out***.println("가로 입력");  obj.ga=key.nextInt();  System.***out***.println("세로 입력");  obj.sa=key.nextInt();  System.***out***.println("높이 입력");  obj.ha=key.nextInt();  //입력 받은 값으로 가로, 세로, 높이 필드 초기화 – 본인작성  obj.volume();  //volume() 메소드 호출 – 본인작성  //객체 내용 출력 – 본인작성  System.***out***.print(obj.toString()+obj.vo);  }  } |
| **[실행결과]** |

* 프로그램 과제 4 : 원 둘레와 원 넓이를 계산하는 Circle 클래스를 작성하시오.
* 필드 : double형의 반지름
* getArea() 메소드- 원 넓이를 계산하여 반환
* getCircum() 메소드 - 원 둘레를 계산하여 반환
* toString() 메소드 – 객체 내용(반지름, 넓이, 둘레)을 문자열로 반환

Circle 객체를 생성하고 테스트하는 CircleTest 클래스를 작성 하시오.

* main() 메소드 – Circle 객체 생성, 입력받은 값으로 반지름초기화, 객체 내용 출력

|  |
| --- |
| **[프로그램 소스]**  **import** java.util.\*;  **class** Circle{  **double** ban;  **public** **double** getArea(){  **return** (Math.***PI***)\*2\*ban;  }  **public** **double** getCircum() {  **return** 2\*(Math.***PI***)\*ban;  }  **public** String toString() {  **return** "반지름 :"+ban +"넓이 :"+ getArea() +"둘레 : " +getCircum() ;  }  }  **public** **class** test05 {  **public** **static** **void** main(String [] args){  Scanner key = **new** Scanner(System.***in***);  Circle obj = **new** Circle();  System.***out***.print("반지름 입력");  obj.ban=key.nextDouble();  System.***out***.println( obj.toString());    }  } |
| **[실행결과]** |

* 메소드 오버로딩

**public** **class** Dim {

//메소드 오버로딩 : 매개변수 개수와 타입, 메소드 반환타입을 다르게 정의

//단, 메소드 이름은 동일하다

**int** sum(**int**[] arr){

**int** hap=0;

**for**(**int** i=0; i<arr.length;i++)

hap += arr[i];

**return** hap;

}

**double** sum(**double**[] arr){

**double** hap=0;

**for**(**double** d : arr)

hap += d;

**return** hap;

}

}

**public** **class** DimTest {

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

Dim obj=**new** Dim(); //객체 생성

//정수형 무명 배열을 매개변수로 전달

**int** ihap=obj.sum(**new** **int**[] {4,2,6,5,8,9});

//실수형 무명 배열을 매개변수로 전달

**double** dhap = obj.sum(**new** **double**[] {12.3, 4.2, 5.7, 2.4, 4.1});

System.***out***.println("정수 배열 합 " + ihap);

System.***out***.println("실수 배열 합 " + dhap);

}

}

|  |
| --- |
| **[실행결과]** |

* 프로그램 과제 1: 무명 배열을 매개변수로 받아 가장 큰 값을 반환하는 메소드 max()로 구성되는 클래스 MaxDim을 작성하시오. 단, 무명배열은 정수형, 실수형, 문자형이며 max()메소드는 오버로딩한다.

class MaxDim{

//정수형 배열을 매개변수로 받아 가장 큰 값을 반환하는 max() - 본인 작성

//실수형 배열을 매개변수로 받아 가장 큰 값을 반환하는 max() - 본인 작성

//문자형 배열을 매개변수로 받아 가장 큰 값을 반환하는 max() - 본인 작성

}

* 프로그램 과제 2: MaxDim 클래스 객체를 생성하고 테스트하는 MaxDimTest 클래스를 작성하시오. 주석에 해당하는 문장을 작성하시오

**public** **class** MaxDimTest {

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

//MaxDim 클래스 객체 선언 & 생성

//max() 메소드 호출, 정수형 무명 배열을 매개변수로 전달, 반환값 변수에 저장

//max() 메소드 호출, 실수형 무명 배열을 매개변수로 전달, 반환값 변수에 저장

//max() 메소드 호출, 문자형 무명 배열을 매개변수로 전달, 반환값 변수에 저장

// 반환받은 결과출력

}

}

|  |
| --- |
| **[프로그램 소스]**  **class** MaxDim{  **public** **int** max(**int** [] a){  **int** mas =a[0];  **for**(**int** i=0; i<a.length;i++){  **if**( a[i] > mas){  mas=a[i];  }  }  **return** mas;  }  //정수형 배열을 매개변수로 받아 가장 큰 값을 반환하는 max() - 본인 작성  //실수형 배열을 매개변수로 받아 가장 큰 값을 반환하는 max() - 본인 작성  **public** **double** max(**double** [] a){  **double** mas =a[0];  **for**(**int** i=0; i<a.length;i++){  **if**( a[i] > mas){  mas=a[i];  }  }  **return** mas;  }  //문자형 배열을 매개변수로 받아 가장 큰 값을 반환하는 max() - 본인 작성  **public** **char** max(**char** [] a){  **char** mas =a[0];  **for**(**int** i=0; i<a.length;i++){  **if**( a[i] > mas){  mas=a[i];  }  }  **return** mas;  }  }  **public** **class** text07 {  **public** **static** **void** main(String[] args) {  MaxDim obj = **new** MaxDim();  //MaxDim 클래스 객체 선언 & 생성  //max() 메소드 호출, 정수형 무명 배열을 매개변수로 전달, 반환값 변수에 저장  **int** a=obj.max( **new** **int**[]{1,2,3,4,5,6,7,});  //max() 메소드 호출, 실수형 무명 배열을 매개변수로 전달, 반환값 변수에 저장  **double** b= obj.max( **new** **double**[]{1.1,2.2,3.2,4.8,5.2,6.9,7.2,});  //max() 메소드 호출, 문자형 무명 배열을 매개변수로 전달, 반환값 변수에 저장  **char** c= obj.max( **new** **char**[]{'a','b','c','d','e','f'});  // 반환받은 결과출력  System.***out***.print( "정수"+a +" "+"실수"+b+" "+"문자 "+c+" " );  }  } |
| **[실행결과]** |