|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **2019\_1\_java1\_11** | **학번 :** | **20135151** | **이름 :** | **이갑성** |

* **내용 점검**

|  |
| --- |
| 1. 클래스와 객체  * 클래스는 필드와 메소드로 구성 * 클래스는 객체를 생성하는 틀 * 클래스내의 필드와 메소드는 생성된 객체를 통해서 접근. 단, 접근 시 .(dot)사용 * 객체를 생성할 때 new 연산자 사용  1. 생성자  * 객체 초기화 * 클래스 이름과 동일 * 반환 값 없음 |
| //라이브러리(API:Application Program Interface) 용 클래스  **class** Student{  //필드 선언  String n; //이름을 저장하기 위한 필드 선언  **int** j, p; //과목을 저장하기 위한 필드 선언  **int** hap; //과목합을 저장하기 위한 필드 선언  **double** ave;  //생성자 구현  Student(String n, **int** j, **int** p){ //매개변수로 받은 값을 필드 값으로 초기화  **this**.n=n;  **this**.j=j;  **this**.p=p;  }  //메소드 구현  **void** cal(){ //과목 합과 평균을 계산하는 메소드  hap=j+p;  ave = hap/2;  }  }  //실행용 클래스 - main() 메소드를 제공  **public** **class** StudentTest {  **public** **static** **void** main(String[] args) {    //클래스의 필드와 메소드는 객체를 통해서 접근 해야 함  Student obj=**new** Student("김낙준", 90, 85); //Student 클래스 객체 obj 선언 & 생성  obj.cal();  System.***out***.println("이름 : "+ obj.n); //name 필드 초기화  System.***out***.println("자바 성적 : " + obj.j); //java 필드 초기화  System.***out***.println("파이썬 성적 : "+obj.p); //eng 필드 초기화  System.***out***.println("과목합 : " + obj.hap);  System.***out***.println("과목평균 : " + obj.ave);  }  } |
| **[실행 결과]** |

* **프로그램 과제**

1. 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-본인 작성**

**//사각형 넓이를 계산하여 출력하는 메소드 area(), 반환값 없음 –본인 작성**

}

**//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");

**//입력 받은 값을 가로필드로 초기화**

System.*out*.println("세로를 입력하세요");

**//입력 받은 값을 세로필드로 초기화**

**//area() 호출**

}

}

|  |
| --- |
| **[프로그램 소스]**  //사각형 넓이를 계산하기 위한 클래스  **public** **class** Rectangle {  //가로와 세로 값을 저장하기 위한 필드 선언, double형으로 할 것-본인 작성  **double** garo;  **double** sero;  //사각형 넓이를 계산하여 출력하는 메소드 area(), 반환값 없음 –본인 작성  **public** **void** area()  {  **double** sum = garo \* sero;  System.***out***.println("가로: " + garo + " 세로: " + sero + " 넓이: " + sum);  }  }  **import** java.util.\*;  **public** **class** RectangleTest {  **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");  // 입력 받은 값을 가로필드로 초기화  rec.garo = key.nextDouble();  System.***out***.println("세로를 입력하세요");  // 입력 받은 값을 세로필드로 초기화  rec.sero = key.nextDouble();  // area() 호출  rec.area();  }  } |
| **[실행결과]** |

1. 다음과 같은 필드와 메소드로 구성되는 Box 클래스를 작성하고 테스트 하시오

class Box{

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

//생성자 : 매개변수로 가로, 세로, 높이 값을 받아 필드값으로 초기화

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

}

public class BoxTest {

public static void main(String[] args) {

//Box 클래스 객체 선언 – 본인 작성

//객체 생성 - 입력 받은 가로, 세로, 높이 값을 생성자 매개변수로 전달하여 필드 초기화 – 본인 작성

//volume() 메소드 호출 – 본인 작성

}

}

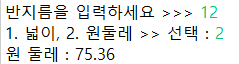
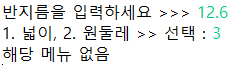
|  |
| --- |
| **[프로그램 소스]**  **class** Box {  // 핃드구성 : 가로(int), 세로(int), 높이(int), 부피(int)  **int** garo, sero, height, vol;  // 생성자 : 매개변수로 가로, 세로, 높이 값을 받아 필드값으로 초기화  **public** Box(**int** g, **int** s, **int** h)  {  garo = g;  sero = s;  height = h;  }  // volume() 메소드 : 부피를 계산하여 필드에 저장, 매개변수와 반환 값 없음  **public** **void** volume()  {  **this**.vol = garo \* sero \* height;  }  }  **public** **class** BoxTest {  **public** **static** **void** main(String[] args) {  // Box 클래스 객체 선언 – 본인 작성  Box b1;  // 객체 생성 - 입력 받은 가로, 세로, 높이 값을 생성자 매개변수로 전달하여 필드 초기화 – 본인 작성  b1 = **new** Box(5, 5, 10);  // volume() 메소드 호출 – 본인 작성  b1.volume();  System.***out***.println("부피: " + b1.vol);  }  } |
| **[실행결과]** |

1. 원 둘레와 원 넓이를 계산하는 Circle 클래스를 작성하시오.

* 필드 : double형의 반지름
* 생성자 : 매개변수로 받은 값을 필드로 초기화
* getArea() 메소드- 원 넓이를 계산하여 출력, 반환값 없음
* getCircum() 메소드 - 원 둘레를 계산하여 출력, 반환값 없음

Circle 객체를 생성하고 테스트하는 CircleTest 클래스를 작성 하시오.

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

|  |
| --- |
| **[프로그램 소스]**    **public** **class** Circle {  //필드 : double형의 반지름  **double** r;  //생성자 : 매개변수로 받은 값을 필드로 초기화  **public** Circle(**double** r)  {  **this**.r = r;  }  //getArea() 메소드- 원 넓이를 계산하여 출력, 반환값 없음  **public** **void** getArea()  {  **double** result;  result = r \* r \* 3.14;  System.***out***.println("반지름 " + r + "인 원의 넓이: " + result);  }  //getCircum() 메소드 - 원 둘레를 계산하여 출력, 반환값 없음  **public** **void** getCircum()  {  **double** result;  result = r \* 2 \* 3.14;  System.***out***.println("반지름 " + r + "인 원의 둘레: " + result);  }  }  **import** java.util.Scanner;  **public** **class** CircleTest {  **public** **static** **void** main(String[] args) {  Scanner sc = **new** Scanner(System.***in***);  **int** menu;  **double** r;    System.***out***.print("반지름을 입력하세요 >>> ");  r = sc.nextDouble();  Circle c = **new** Circle(r);  System.***out***.print("1.넓이, 2.원둘레 >> 선택: ");  menu = sc.nextInt();  **switch**(menu)  {  **case** 1 :  c.getArea();  **break**;  **case** 2 :  c.getCircum();  **break**;  **default** :  System.***out***.println("해당 메뉴 없음");  **break**;  }  }  } |
| **[실행결과]** |

1. 아이디는 키보드로 입력 받으며, 비밀번호는 4자리수의 정수형 난수로 초기화하는 Info 클래스를 제시된 조건대로 작성하시오.

* 필드 구성

- id : String, 아이디 저장

- pass : int, 비밀번호 저장

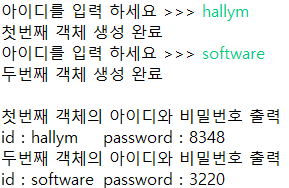
* 생성자 : 아이디 필드는 매개변수로 받은 값으로 초기화하고, 난수를 생성하여 비밀번호를 초기화
* disPlay() 메소드 : 아이디와 비밀번호 출력, 반환값 없음

Info 객체를 생성하고 테스트하는 InfoTest 클래스를 작성 하시오.

* main() 메소드

- Info 객체를 두개를 선언하고 아이디는 입력 받아서 생성자 매개변수로 전달

- 두 개의 객체 내용 출력(disPlay()메소드 호출)



|  |
| --- |
| **[프로그램 소스]**  **public** **class** Info {  String id;  **int** pass;    **public** Info(String id)  {  **this**.id = id;  pass = (**int**)(Math.*random*()\*9000) + 1000;  }    **public** **void** disPlay() {  System.***out***.println("id : " + id + "\tpassword: " + pass);  }  }  **import** java.util.Scanner;  **public** **class** InfoTest {  **public** **static** **void** main(String[] args) {  Scanner sc = **new** Scanner(System.***in***);  String id;  System.***out***.print("아이디를 입력 하세요: ");  id = sc.next();  Info i1 = **new** Info(id);  System.***out***.println("첫번째 객체 생성 완료");  System.***out***.print("아이디를 입력 하세요: ");  id = sc.next();  Info i2 = **new** Info(id);  System.***out***.println("두번째 객체 생성 완료");    System.***out***.println("\n\n첫번째 객체의 아이디와 비밀번호 출력");  i1.disPlay();  System.***out***.println("두번째 객체의 아이디와 비밀번호 출력");  i2.disPlay();  }  } |
| **[실행결과]** |