자바 프로그래밍 4장

|  |  |
| --- | --- |
| 분반 | 1분반 |
| 학번 | 2012136132 |
| 성명 | 최강석 |

221PAGE 1번문제

1. 문제 : 강아지를 나타내는 Dog이라는 이름의 **클래스를 생성한다.**

Dog**클래스는 다음과 같은필드를 가져야한다.**

**name :** 강아지**의** 이름**,전용멤버**

**breed :** 강아지**의 종류, 예를 들면 "푸들",공용멤버**

**age :** 강아지**의 나이,전용멤버**

Dog**클래스는 다음과 같은 생성자와 메소드를 가져야한다.**

**초기화되지않은 필드를 null이나 0으로 초기화한다.**

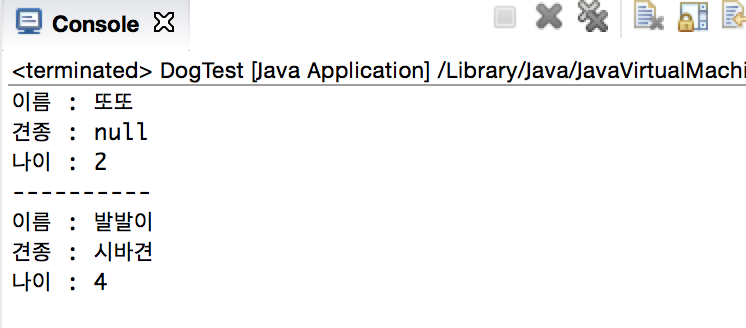
Dog**(String name, in age) :** 강아지**의** 이름**과 나이를 초기화**

Dog**(Strin name, String breed, int age) :** 강아지**의** 이름 **, 종류 , 나이를 초기화**

1. 알고리즘 :
   1. Dog class를 생성해주고 멤버변수로 이름, 견종, 나이를 선언해준다.
   2. 다음은 생성자를 이름과 나이만 받는 생성자, 모두 받는 생성자를 선언해주고 클래스의 멤버변수에 매개변수의 값들을 대입을 시켜준다.
   3. 클래스 멤버변수들을 출력해주는 함수들을 만들어준다.
   4. 메인함수에서 객체생성과 출력들을 해준다.
2. 코드 :

|  |
| --- |
| **class** Dog{//Dog 클래스 생성.  **private** String name;  **private** String breed;  **private** **int** age;  //매개변수로 이름과, 견종, 나이.    **public** Dog(String name, **int** age){  **this**.name = name;  **this**.breed = **null**;  **this**.age = age;  //이름과 나이만 받는 생성자.  }  **public** Dog(String name, String breed, **int** age){  **this**.name = name;  **this**.breed = breed;  **this**.age = age;  //이름, 견종, 나이 전부를 받는 생성자.  }  **public** **void** print(){  //클래스의 멤버변수들을 출력해준다.  System.***out***.println("이름 : " + **this**.name);  System.***out***.println("견종 : " + **this**.breed);  System.***out***.println("나이 : " + **this**.age);  }  }  **public** **class** DogTest {  **public** **static** **void** main(String[] args) {  Dog d1 = **new** Dog("또또", 2);  //이름과 나이만 받는 객체생성.  Dog d2 = **new** Dog("발발이", "시바견", 4);  //이름, 견종, 나이 전부를 받는 객체생성.    //객체들을 출력해준다.  d1.print();  System.***out***.println("----------");  d2.print();  }  } |

1. 결과



221PAGE 2번문제

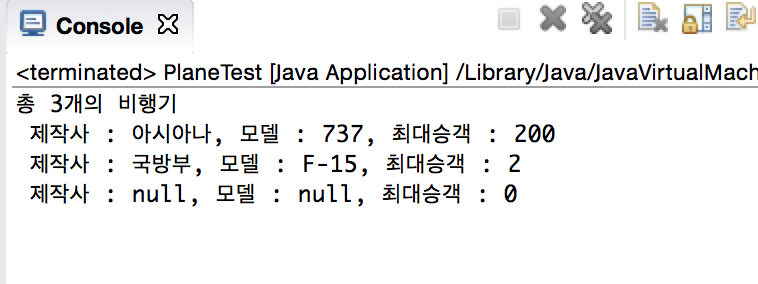
1. 문제 :

비행기를 나타내는 Plane라는 이름의 클래스를 설계하라.Plane 클래스는 제작사(예를 들어서 에어버스),모델(A380),최대 승객수(500)를 필드로 가지고 있다.  
   
(1). 필드를 정의하라.모든 필드는 전용 멤버로하라.  
   
(2). 모든 필드에 대한 접근자와 설정자 메소드를 작성한다.  
   
(3). Plane 클래스의 생성자를 몇 개를 중복 정의하라. 생성자는 모든 데이터를 받을 수도 있고 아니면 하나도 받지 않을 수 있다.  
   
(4).PlaneTest라는 이름의 테스트 클래스를 만드는데 main()에서 Plane 객체 여러 개를 생성하고 접근자와 설정자를 호출하여 보라.  
   
(5).Plane 클래스에 지금까지 생성도니 비행기의 개수를 나타내는 정적 변수인 planes를 추가하고 생성자에서 증가시켜 보자.  
   
(6).Plane 클래스에 정적 변수 planes의 값을 반환하는 정적 메소드인 getPlanes()를 추가하고 main()에서 호출하여 보라.

1. 알고리즘 :
   1. 멤버 변수로 제작사, 모델, 최대승객수, 정적변수인 planes를 선언해준다.
   2. 생성자를 여러 개 만들어 주고 각 생성자에는 정적변수인 planes을 ++연산을 통해 호출할 때마다 증가시켜 줄 수 있도록 해준다.
   3. 각 멤버변수에 대한 설정자와 접근자를 선언해준다.
   4. 객체를 여러 개 만들어 객체를 출력시켜준다.
2. 코드 :

|  |
| --- |
| **class** Plane{  **private** String production;  **private** String model;  **private** **int** maxPassenger;  **private** **static** **int** *planes* = 0;  //멤버 변수로 제작사, 모델, 최대승객수  //그리고 마지막으로 정적 변수 planes 선언  **public** Plane(){  //매개변수가 없을시  //모든 값을 빈 값으로 대입을 해주고  //비행기 수를 증가시켜야하므로  //정적변수인 planes++; 연산  production = **null**;  model = **null**;  maxPassenger =0;  *planes*++;  }  **public** Plane(String p, String m){  //매개변수로 제작사, 모델만 받을시에  //해당하는 값들을 대입을 해주고  //비행기수 증가를 시켜준다.(planes)  production = p;  model = m;  *planes*++;  }  **public** Plane(String p, String m, **int** n){  //매개변수로 제작사, 모델, 최대승객수를 받을 시에  //해당하는 값들을 대입을 해주고  //비행기 수 증가.  production=p;  model=m;  maxPassenger=n;  *planes*++;  }  **public** Plane(**int** n){  maxPassenger=n;  *planes*++;  //최대 승객수만 받을 경우  //그 값을 대입해주고  //비행기 수 증가.  }  //각 멤버변수에 대한 접근자와 설정자에 대한  //함수를 만들어 준다.  **public** String getproduction(){  **return** production;  }  **public** String getmodel(){  **return** model;  }  **public** **int** getmaxPassenger(){  **return** maxPassenger;  }  **public** **static** **int** getPlanes(){  **return** *planes*;  //정적 변수의 접근자도 설정해준다.  }  **public** **void** setproduction(String p){  production = p;  }  **public** **void** setmodel(String m){  model = m;  }  **public** **void** setmaxPassenger(**int** n){  maxPassenger = n;  }    }  **public** **class** PlaneTest {  **public** **static** **void** main(String[] args) {  //객체를 생성해준다.  Plane p1 = **new** Plane("아시아나", "737", 200);  Plane p2 = **new** Plane("국방부", "F-15", 2);  Plane p3 = **new** Plane();    //객체에 대한 출력들을 해준다.  System.***out***.println("총 " +Plane.*getPlanes*() + "개의 비행기");  System.***out***.println(" 제작사 : " + p1.getproduction()  + ", 모델 : " + p1.getmodel() +  ", 최대승객 : " + p1.getmaxPassenger());    System.***out***.println(" 제작사 : " + p2.getproduction()  + ", 모델 : " + p2.getmodel() +  ", 최대승객 : " + p2.getmaxPassenger());    System.***out***.println(" 제작사 : " + p3.getproduction()  + ", 모델 : " + p3.getmodel() +  ", 최대승객 : " + p3.getmaxPassenger());  }  } |

1. 결과



221PAGE 3번문제

1. 문제 : 상자를 나타내는 Box라는 이름의 클래스를 설**계하라.** Box 클래스**는** 상자**의 높이,너비,높이를 필드로 가지고 있다.**

**박스가 비어있는지 그렇지 않은지를** 나타내는 **empty라고 하는 필드도 추가한다.**

Box 클래스**의 생성자를 중복 정의하라.**

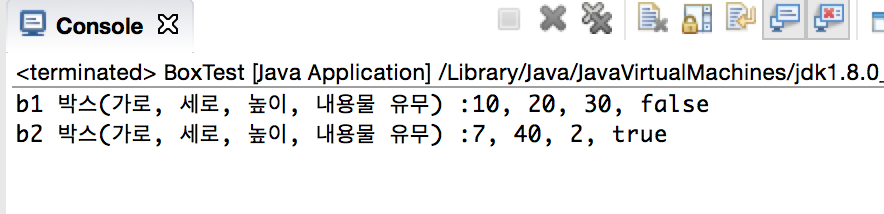
**생성자는 모든 데이터를 받을 수도 있고 아니면 하나도 받지 않을 수 있다.**

**새로 생성된** Box**는 비어있다고 가정한다.**

1. 알고리즘 :
   1. 상자의 멤버변수로 세로, 가로, 높이, empty를 선언해준다.
   2. 박스에 대한 생성자에 각 멤버변수에 매개변수를 대입을 해주고 생성자에서 empty값을 false로 대입을 해준다.
   3. 그리고 각 세로, 가로, 높이에 대한 설정자와 접근자에 대한 함수를 정의 해준다.
   4. 메인 함수에 여러 개의 박스 객체를 생성하고 그에 대한 정보들을 출력을 해준다.
2. 코드 :

|  |
| --- |
| **class** Box{  **private** **int** width;  **private** **int** length;  **private** **int** height;  **private** **boolean** empty;  //세로, 가로, 높이, 상자의 내용물의 유/무  //에 대한 멤버 변수들을 선언해준다.    **public** Box(){  //박스를 그냥 만들어줄 경우에는  //내용물은 무.  **this**.empty = **false**;  **this**.width=0;  **this**.length=0;  **this**.height=0;  }  **public** Box(**int** width, **int** length, **int** height){  //박스의 가로,세로,높이의 매개변수를 받을 경우에  //그 값을 대입을 해주고 상자는 빈 값으로 넣어준다.  **this**.width=width;  **this**.length=length;  **this**.height=height;  **this**.empty=**false**;  }  //각 세로, 가로, 높이에 대한 설정자와 접근자에 대한  //함수를 정의해준다.  **public** **int** getWidth(){  **return** width;  }  **public** **void** setWidth(**int** width){  **this**.width=width;  }  **public** **int** getLength(){  **return** length;  }  **public** **void** setLength(**int** length){  **this**.length=length;  }  **public** **int** getHeight(){  **return** height;  }  **public** **void** setHeight(**int** height){  **this**.height=height;  }  **public** **boolean** getEmpty(){  **return** empty;  //상자의 내용물의 유/무의 반환.  }  **public** **void** setEmpty(**boolean** empty){  **this**.empty=empty;  //상자의 내용물의 설정.  }    }  **public** **class** BoxTest {  **public** **static** **void** main(String[] args) {  Box b1=**new** Box(10, 20, 30);//멤버변수 가로, 세로, 높이를 받는 박스 객  Box b2=**new** Box(); //멤버변수가 없는 박스객체.  //b2의 객체에 가로, 세로, 높이, 상자의 내용물의 여부를  //설정자를 이용해주어 대입을 해준다.  b2.setHeight(2);  b2.setLength(40);  b2.setWidth(7);  b2.setEmpty(**true**);  //b1과 b2에 대한 정보들을 출력해준다.  System.***out***.println("b1 박스(가로, 세로, 높이, 내용물 유무) :"  + b1.getWidth() +", "+ b1.getLength()+", " +  b1.getHeight() + ", " + b1.getEmpty());    System.***out***.println("b2 박스(가로, 세로, 높이, 내용물 유무) :"  + b2.getWidth() +", "+ b2.getLength()+", " +  b2.getHeight() + ", " + b2.getEmpty());    }  } |

1. 결과



1221PAGE 4번문제

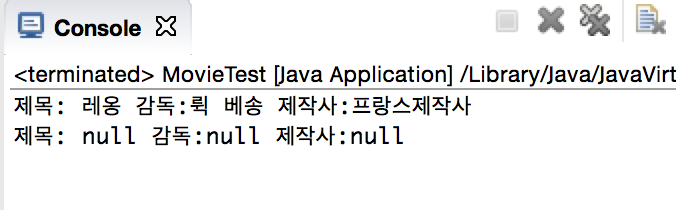
1. 문제 : 영화를 나타내는 Movie라는 이름의 클래스를 설계하라.  
   제목, 감독, 제작사를 나타내는 필드를 가진다. Movie 클래스의   
   생성자를 중복 정의하라. 생성자는 모든 데이터를 받을 수도 있고   
   아니면 하나도 받지 않을 수 있다.
2. 알고리즘 :
   1. Movie의 멤버변수로 제목, 영화감독, 제작사를 선언해준다.
   2. 생성자를 두개 만들어서 하나는 아무것도 안받을 경우 하나는 전부 받을 경우로 해주어서 전부받을경우 멤버변수에 매개변수의 값을 대입을 해주고 아무 것도 안받을 경우는 빈 값을 넣어준다.
   3. print함수를 만들어 Movie에 대한 멤버변수들을 출력해준다.
   4. 메인함수에 모든 데이터를 받는 객체와 하나도 받지 않는 객체를 생성해주어서

그에대한 정보들을 출력을 해준다.

1. 코드 :

|  |
| --- |
| **class** Movie{  **private** String title;  **private** String director;  **private** String production;  //제목, 영화감독, 제작사에 대한 멤버변수 선언.    **public** Movie(String title, String director, String production){  //매개변수를 모두 받을 경우에  //멤버변수에 매개변수의 값들을 대입을 해준다.  **this**.title=title;  **this**.director=director;  **this**.production=production;  }  **public** Movie(){  //매개변수가 없을 경우 빈 값을 넣어준다.  **this**.title=**null**;  **this**.director=**null**;  **this**.production=**null**;    }  **public** **void** print(){  //Movie에 대한 멤버변수들을 출력해준다.  System.***out***.println("제목: " + title + " 감독:" + director +" 제작사:" +production);  }    }  **public** **class** MovieTest {  **public** **static** **void** main(String[] args) {  Movie m1=**new** Movie("레옹", "뤽 베송", "프랑스제작사");  //모든 데이터를 받을 경우.  Movie m2=**new** Movie();  //생성자가 데이터를 하나도 받지 않을 경우    m1.print();  m2.print();  //객체 m1과 m2에 대한 정보 출력  }  } |

1. 결과



221PAGE 5번문제

1. 문제 : 은행 계좌를 나타내는 BankAccount 라는 이름의 클래스를 설계하라.  
   BankAccount 클래스는 이름, 계좌번호, 잔액, 이자율을 나타내는 필드를 가진다.  
   BankAccount 클래스의 생성자를 중복 정의하라.  
   생성자는 모든 데이터를 받을 수도 있고 아니면 하나도 받지 않을 수 있다.
2. 알고리즘 :
   1. Class BankAccount를 선언해주고 그에대한 멤버변수로 이름, 계좌번호, 잔액, 이율을 선언해준다.
   2. 매개변수를 모두 받는 경우 멤버 변수에 매개 변수의 값들을 넣어준다.
   3. 매개변수를 받지 않는 생성자일 경우에는 빈값을 넣어준다.
   4. Print()함수에는 BankAccount에 대한 멤버변수를 출력을 해주는 내용을 넣어준다.
   5. 메인함수에서 객체를 생성해주고 print()함수 호출을 통해 각 객체에 대한 멤버변수들을 출력해준다.
3. 코드 :

|  |
| --- |
| **class** BankAccount{  **private** String name;  **private** String AccountNum;  **private** **int** money;  **private** **double** rates;  //멤버 변수로 이름과, 계좌번호, 잔액, 이율을 선언해준다.    **public** BankAccount(String name, String AccountNum, **int** money, **double** rates){  //매개 변수를 모두 받는 경우  //멤버 변수에 매개 변수의 값들을 넣어준다.  **this**.name=name;  **this**.AccountNum=AccountNum;  **this**.money=money;  **this**.rates=rates;  }  **public** BankAccount(){  //매개 변수를 받지 않는 경우  //멤버 변수에 빈 값들을 넣어준다.  **this**.name=**null**;  **this**.AccountNum=**null**;  **this**.money=0;  **this**.rates=0;  }  **public** **void** print(){  //클래스의 멤버변수에 대한 정보들을 출력해준다.  System.***out***.println("name : " + **this**.name);  System.***out***.println("AccountNum : " + **this**.AccountNum);  System.***out***.println("money : " + **this**.money);  System.***out***.println("rates : " + **this**.rates);  }  }  **public** **class** BankAccountTest {  **public** **static** **void** main(String[] args) {  //매개변수를 모두 갖는 객체 b1을 생성해주고  //하나도 받지 않는 객체 b2를 생성해준다.  BankAccount b1=**new** BankAccount  ("최강석", "110-360-123456", 10000, 2.4);  BankAccount b2=**new** BankAccount();    //b1과 b2에 대한 정보들을 출력해준다.  b1.print();  System.***out***.println("-------------------");  b2.print();    }  } |

1. 결과

