자바 프로그래밍 7장

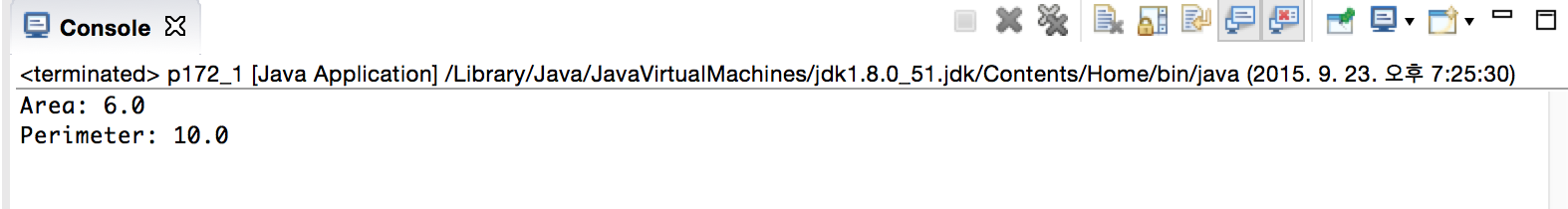
|  |  |
| --- | --- |
| 분반 | 1분반 |
| 학번 | 2012136132 |
| 성명 | 최강석 |

172PAGE 1번문제

1. 문제 : 사각형을 나타내는 클래스 Rectangle을 만들어보자. 사각형은(w)와 세로(h)를 가지며, 사각형의 넓이를 반환하는 area(), 사각형의 둘레를 반환하는 perimeter(), 등의 메소드를 가진다. Rectangle 클래스를 작성하고 객체를 생성하여 테스트하라.
2. 알고리즘 :
   1. Rectangle변수에 가로와 세로의 멤버변수를 선언해준다.
   2. 다음은 면적, 둘레를 반환해줄 함수를 각각 선언해준다.
   3. 메인에 객체를 생성해주고 멤버변수에 가로와 높이를 대입해준다.
   4. 객체의 함수를 호출하여서 면적과 둘레를 출력해준다.
3. 코드 :

|  |
| --- |
| **class** Rectangle{  **double** width;//가로 멤버변수  **double** height;//세로 멤버변수  **public** **double** area() {  **return** width\*height;  //면적을 반환해준다.  }  **public** **double** perimeter(){  **return** 2.0\*(width+height) ;  //둘레를 반환해준다.  }  }  **public** **class** p172\_1 {  **public** **static** **void** main(String[] args) {  // **TODO** Auto-generated method stub  Rectangle rec=**new** Rectangle();  //객체를 생성해준다..  //클래스 멤버변수 가로와 높이를 대입해준다.  rec.width= 2.0;  rec.height = 3.0;  System.***out***.println("Area: " + rec.area());//면적을 출력해준다.  System.***out***.println("Perimeter: " + rec.perimeter());//둘레를 출력해준다.  }  } |

1. 결과

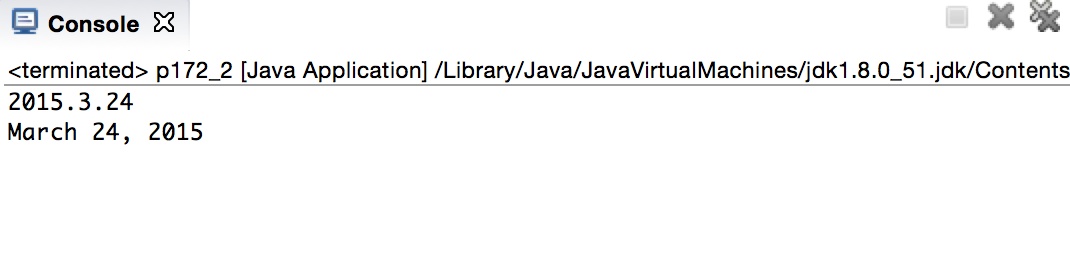


172PAGE 2번문제

1. 문제 : 날짜를 나타내는 클래스 Date를 만들어보자. Date는 연도, 월, 일 등의 속성을 가지며, 날짜를 “2012.7.12.”과 같이 출력하는 메소드 print1(), 날짜를 “July 12, 2012"와 같이 출력하는 print2() 등의 메소드를 가진다. Date 클래스를 작성하고 객체를 생성하여서 테스트하라.
2. 알고리즘 :
   1. 데이트 클래스를 생성해서 멤버변수로 년, 월, 일을 선언해준다.
   2. 숫자로 년, 월, 일을 출력하는 함수와 Switch를 이용한 문자열로 년, 월, 일을 출력하는 함수를 만들어 준다.
   3. 메인에서 객체를 생성해주고 출력함수를 호출해준다.
3. 코드 :

|  |
| --- |
| **class** Date{//데이트 클래스 생성  **int** year = 2015;  **int** month = 3;  **int** day = 24;  //멤버변수로 정수형으로 년, 월, 일을 대입을 해준다.  //자바같은경우는 private에서 초기화가 가능하다.    **public** **void** print1() {  System.***out***.println( year + "."+month + "." + day);  // 숫자로 년, 월, 일을 출력하는 함수  }  **public** **void** print2(){  String m; //월을 입력받을 String m 선언    **switch**(month){  //스위치를 이용하여 month가 각 케이스일 경우에  //해당하는 월을 입력받는다.  **case** 1 : m = "January"; **break**;  **case** 2 : m = "February"; **break**;  **case** 3 : m = "March"; **break**;  **case** 4 : m = "April"; **break**;  **case** 5 : m = "May"; **break**;  **case** 6: m = "june"; **break**;  **case** 7 : m = "july"; **break**;  **case** 8 : m = "August"; **break**;  **case** 9 : m = "September"; **break**;  **case** 10 : m = "October"; **break**;  **case** 11 : m = "November"; **break**;  **case** 12 : m = "December"; **break**;  **default** : m = "Wrong month"; **break**;  }  System.***out***.println(m + " " + day + ", " + year);  //월이 문자열로되는 날짜를 출력해준다.  }    }  **public** **class** p172\_2 {    **public** **static** **void** main(String[] args) {  Date d1 = **new** Date();//Date를 생성해준다.  d1.print1();//숫자 날짜출력  d1.print2();//문자 날짜출력  }  } |

1. 결과 :.

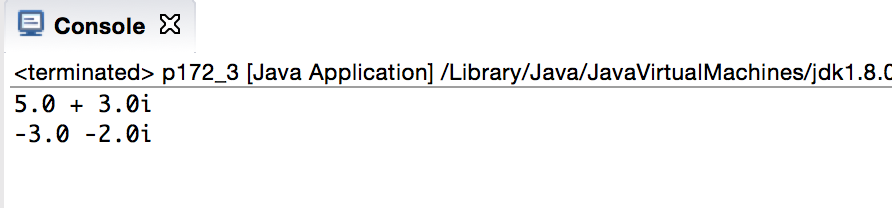


172PAGE 3번문제

1. 문제 : 복소수를 나타내는 클래스를 만들어보자. 복소수는 real + imag \* i 와 같은 형태를 갖는다. 여기서 I = 루트-1이다. 복소수에 필요한 속성들과 가능한 메소드들을 결정한 후에, 클래스를 작성하고 객체를 생성하여서 테스트하라
2. 알고리즘 :
   1. 클래스 Complex를 선언해준다. 매개변수로는 실수, 허수를 나타내줄 더블형 변수를 선언해준다.
   2. Add함수에는 멤버변수에 매개변수를 +=연산해주고 Sub함수에는 멤버변수에 매개변수를 -=연산을 해준다.
   3. Print()함수에는 허수부분이 음수가 나올경우는 –가 자동적으로 출력해주므로 음수가 나올경우는 “+”를 출력해주지 않는다.
   4. 메인함수에 객체 2개를 생성하고 add()와 sub()를 print를 통해 시험해본다.
3. 코드 :

|  |
| --- |
| **class** Complex{  **double** real, img;  //더블형 변수 real은 실수를 의미하고  //img는 허수부분을 의미한다.      **public** **void** add(Complex c){  real+=c.real;//클래스 실수부분에 호출한 객체를 더해준다.  img+=c.img;//클래스 허수부분에 호출한 객체의 허수부분을 더해준다.  }  **public** **void** sub(Complex c){  real-=c.real;//클래스 실수부분에 호출한 객체를 빼준다.  img-=c.img;//클래스 허수부분에 호출한 객체를 빼준다.  }    **public** **void** print(){  **if**(img>= 0){//만약에 허수가 0이상일 경우 +를 붙이고 출력한다.  System.***out***.println(real + " + " + img+"i");  }  **else**{//허수가 음일 경우에는 "+"를 출력하지 않는다.  System.***out***.println(real + " " + img+"i");  }  }  }  **public** **class** p172\_3 {  **public** **static** **void** main(String[] args) {  Complex co1 = **new** Complex();//co1 객체생성.  Complex co2 = **new** Complex();//co2 객체생성.  co1.real=3; //co1의 실수부분에 3을 대입.  co1.img=2; //co1의 허수부분에 2를 대입.    co2.real=2; //co2의 실수부분에 2를 대입.  co2.img=1; //co2의 허수부분에 1를 대입.    co1.add(co2);//co1에 +=co2를 해준 함수 호출.  co1.print();//co1출력.    co2.sub(co1);//co1에 -=co2를 해준 함수 호출.  co2.print();//co2출력.    }  } |

1. 결과

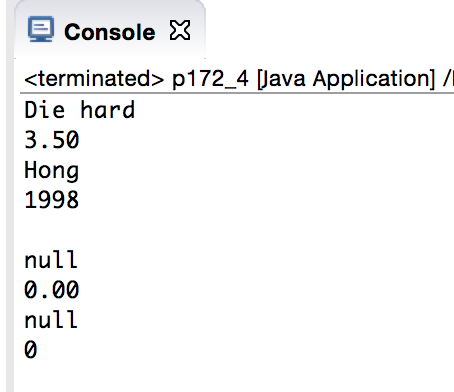


172PAGE 4번문제

1. 문제 : 영화 Movie 클래스를 정의하여 보자. Movie 클래스는 영화 제목, 평점, 감독, 발표된 연도 등의 필드를 가진다. 영화의 모든 정보를 화면에 출력하는 print()라는 메소드를 구현하라. Movie 클래스를 작성하고 객체를 생성하여 테스트하라.
2. 알고리즘 :
   1. Movie 클래스를 생성해주고 맴버변수로 영화제목, 평점, 감독, 개봉일 변수를 선언해준다.
   2. print함수는 멤버변수들의 정보를 출력해줄 코드를 작성.
   3. 메인함수에서 객체를 생성하고 그 객체의 멤버변수에 값을 대입을 해준다.
   4. 그다음 print를 이용하여 영화정보를 출력해준다.
   5. (추가적으로) 자동적으로 클래스 멤버변수들이 초기화 되는지 확인을 해준다.
3. 코드 :

|  |
| --- |
| **class** Movie{  String title;//영화제목.  **double** eval;//영화 평점.  String director;//영화감독.  **int** year;//개봉일.    **public** **void** print() {  //멤버변수들 출력.  System.***out***.printf("%s \n%.2f\n%s\n%d\n\n", title, eval, director, year);  }  }  **public** **class** p172\_4 {  **public** **static** **void** main(String[] args) {  Movie m1 = **new** Movie();//Movie m1 객체생성.  m1.title = "Die hard";  m1.eval = 3.5;  m1.director = "Hong";  m1.year = 1998;  //m1의 매개변수에 대한 내용을 입력을 해줌.  m1.print();//영화정보 출력.    Movie m2 = **new** Movie();  //자바에서 클래스경우에는 초기화를 해주지  //않아도 자동적으로 0, NULL값으로 초기화된다.  m2.print();  //자동으로 0, Null로 초기화된 객체 m2  //에 대한 정보 출력.  }  } |

1. 결과

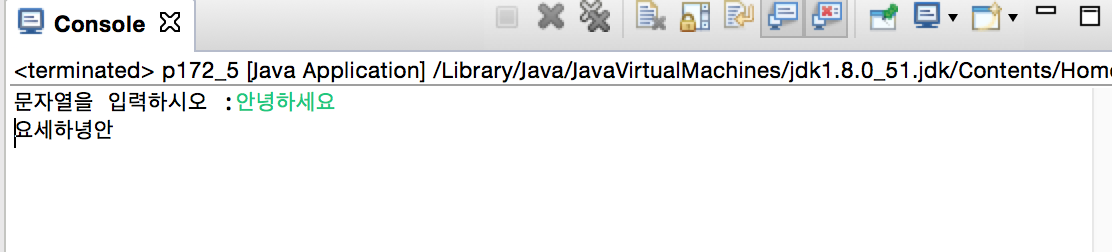


172PAGE 5번문제

1. 문제 : 사용자에게서 받은 문자열을 역순으로 화면에 출력하는 프로그램을 작성하여 보자. 예를 들어서 사용자가 "secret"을 입력하면 "terces"를 출력한다.
2. 알고리즘 :
   1. ReverseString 클래스를 선언해준다.
   2. getString()에는 스캐너를 통해 문자열을 str에 입력받는다.
   3. Reverse()는 입력받은 문자열을통해 charAt을 통해 문자열의 끝에서부터 처음순으로 출력을 시켜준다.
   4. 메인에서는 객체생성을 먼저해준다음에 getString(), Reverse()순으로 함수를 출력해준다.
3. 코드 :

|  |
| --- |
| **import** java.util.Scanner;  //입력을 받기위한 Scanner  **class** ReverseString {  String str;  //문자열을 입력받으려고 String 선언.  **public** **void** getString() {  Scanner input = **new** Scanner(System.***in***);  System.***out***.print("문자열을 입력하시오 :");  str = input.next();  input.close();  //문자열을 입력받고 닫아준다.  }  **public** **void** reverse() {  **for** (**int** i = (str.length() - 1); i >= 0; i--) {  //str의 길이에서 -1해준값은 str의 맨 뒤에 입력받았던 문자열의 위치이다.  //그러므로 그 값부더 --연산을 통해 하나씩 출력해주면  //문자열이 거꾸로 출력하게 된다.  System.***out***.printf("%c", str.charAt(i));  }  }  }  **public** **class** p172\_5 {  **public** **static** **void** main(String[] args) {  ReverseString test = **new** ReverseString();  //객체생성  test.getString();//문자열을 입력받을 함수.  test.reverse();//문자열을 거꾸로 출력하는 함수.  }  } |

1. 결과

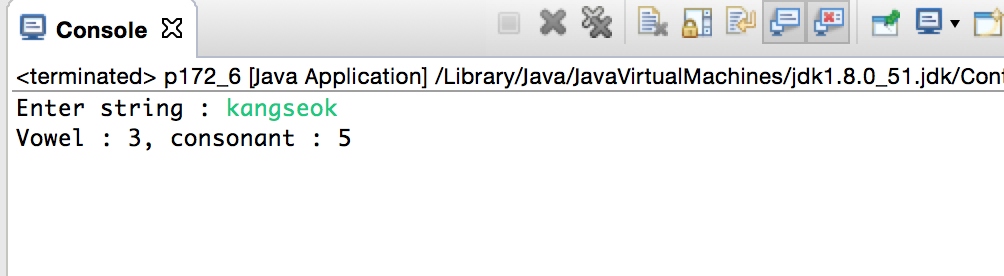


172PAGE 6번문제

1. 문제 : **사용자로부터 받은 문자열에서 자음과 모음의 개수를 계산하여서 출력하는 프로그램을 작성하라.**
2. 알고리즘 :
   1. Class CountM을 선언해주고 문자열을 입력받을 String str을 선언해준다.
   2. getString에서 문자열을 입력받는다.
   3. Count()함수에서는 모음 횟수, 자음 횟수, 모음인지 자음인지 확인하기 위한 char형 변수를 선언해준다. for문에서는 문자열의 길이만큼 루프를 돌리고 문자가 모음이면 모음변수에 ++연산을 해주고 자음이면 자음변수에 –연산을 해준다. 그다음 루프가 끝나고 나면 모음과 자음을 횟수를 출력해준다.
   4. 메인에서는 객체를 생성해주고 getString(), Count순으로 호출을 해준다.
3. 코드 :

|  |
| --- |
| **import** java.util.Scanner;  //입력을 받기위한 Scanner  **class** CountM {  String str;  //문자열을 입력받기위해 String str 선언.  **public** **void** getString() {  Scanner input = **new** Scanner(System.***in***);  //입력을 받기위한 input 선언.  System.***out***.print("Enter string : ");  str = input.nextLine();  input.close();  //문자열을 입력받고 닫아줌.    }  **public** **void** count() {//문자열에서 모음과 자음이 몇개인지 확인해주는 함수.  **int** vowel = 0;//모음 횟수.  **int** consonant = 0;//자음 횟수.  **char** ch;//문자열에서 해당하는 문자열을 검출하기위해 char형 변수 선언.  **for** (**int** i = 0; i < str.length(); i++) {  //문자열의 길이만큼 검사를해줌.  ch = str.charAt(i);  **if** ((ch >= 'A' && ch <= 'Z') || (ch >= 'a' && ch <= 'z'))  **if** (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u')  vowel++;  //모음이 올 경우에 모음변수에 ++를 해준다.  **else**  consonant++;  //자음이 올 경우에 자음변수에 ++를 해준다.  }  System.***out***.println("Vowel : " + vowel + ", consonant : " + consonant);  //자음과 모음의 횟수를 출력해준다.  }  }  **public** **class** p172\_6 {  **public** **static** **void** main(String[] args) {  CountM cnt = **new** CountM();//객체 생성  cnt.getString();//문자열을 입력받는다.  cnt.count(); //문자열의 자음 모음의 수를 출력해주는 함수.  }  } |
|  |

1. 결과



172PAGE 7번문제

1. 문제 : 사용자로부터 아이디를 받아서 미리 저장된 아이디와 일치하는지를 검사하는 프로그램을 작성하여 보자.
2. 알고리즘 :
   1. Class Identity를 생성해준다. 멤버변수로는 문자열 id를 선언해준다.
   2. check함수에는 매개변수로 문자열을 받는다 그리고나서 객체의 문자열과 매개변수의 문자열을 compareTo를 이용하여 아이디가 동일여부를 출력을 해준다.
   3. 메인에는 확인할 id를 입력받기위한 문자열 idcheck를 선언해준다. 그다음 객체를 선언해주고 그 객체의 멤버변수에 문자열(id)를 넣어준다.
   4. 다음은 문자열(idcheck)를 입력받는다.
   5. Check()함수에 idecheck을 매개변수로 넣어주어 호출을 해준다.

1. 코드 :

|  |
| --- |
| **import** java.util.Scanner;  **class** Identity{  String id;//id 문자열 생성.    **public** **void** check(String str){  **if**(str.compareTo(id) == 0){  //문자열.compareTo(비교문자열)같은  //경우에는 문자열이 동일할 경우 0의 값이 나온다.  System.***out***.println("동일합니다.");  }**else**{  System.***out***.println("아이디가 일치하지 않습니다");  //문자열이 0이외의 값이 나올경우 일치하지 않음.  }  }    }  **public** **class** p172\_7 {  **public** **static** **void** main(String[] args) {  String idcheck; //확인할 id를 입력받기위한 문자열 선언.  Scanner input= **new** Scanner(System.***in***);  Identity cks=**new** Identity(); //객체생성.  cks.id="choikang"; //생성한 객체에 아이디를 넣어준다.    System.***out***.print("확인할 id를 입력하시오 : ");  idcheck =input.next();  cks.check(idcheck);  //아이디를 입력하고 그입력한 변수를 check의 매개변수로 넘겨준다.  }  } |

1. 결과

