

학번 : 20193081 이름 : 김민오

[연습 문제]

6장 연습문제 p.277-279에 있는 1, 4, 7, 8, 9를 풀어서 제출하시오. 문제를 풀되 이유나 설명이 필요한 것은 간단히 메모할 것.

1번 : java.lang -> System비롯, 다양한 입출력 등과같은 클래스와 인터페이스를 제공

4번 :

- (1) int i = Integer.parseInt("20");
- (2) double d = Double.parseDouble("35.9");
- (3) String s1 = Boolean.toString(true);
- (4) if(Character.isAlphabetic(c)) System.out.println("eng");
- (5) String s2 = Integer.toBinaryString(n);
- (6) String s3 = Integer.toHexString(50);

7번:

- (1) b
- (2) c,e (equals연산자는 문자열만 비교하므로)

8번:

+, st.hasMoreTokens(), st.nextToken()

9번 : (앞뒤 공백을 2개씩이라고 가정한다면)

a= " HELLO "

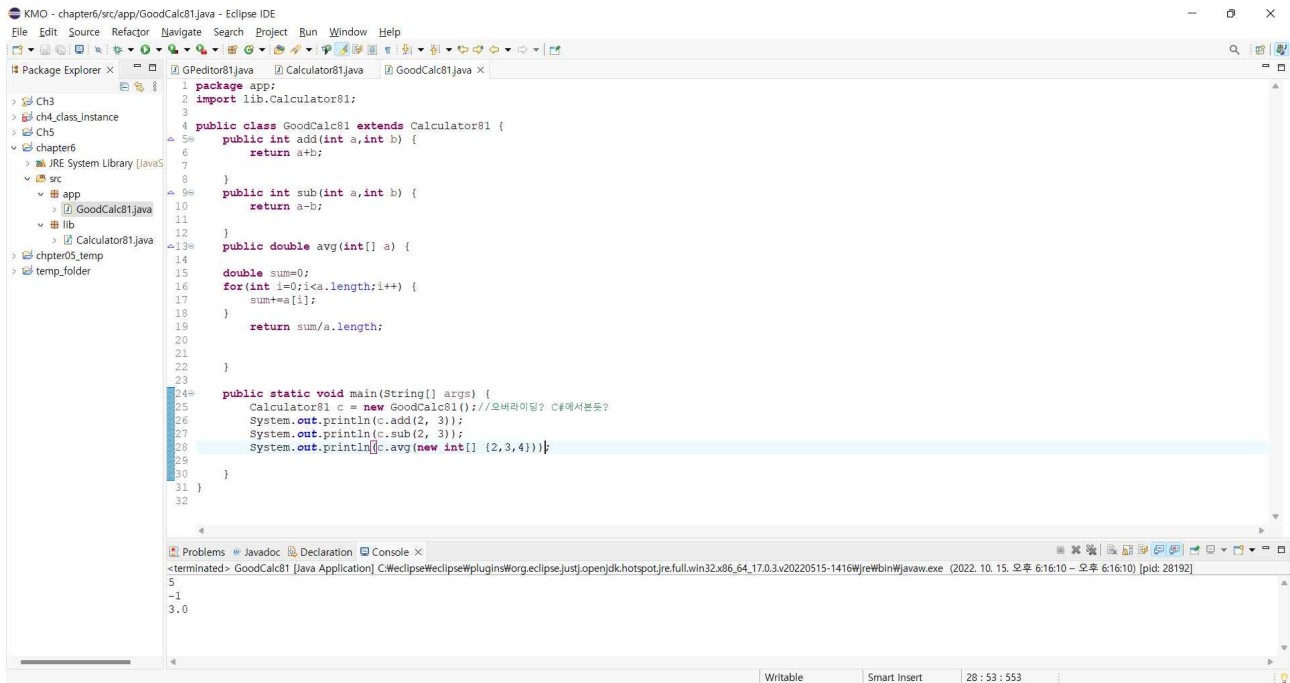
b=" hello "

c="hello"

[실습 문제]

실습 6-1 p.245의 예제 응용프로그램 (Calculator, GoodCalc)을 두 개의 패키지(lib, app)로 나누어 작성하여 수행할 것. 클래스의 이름은 예제에 주어진 클래스 이름을 사용하되, 끝에 자기학번 끝자리 2자리 번호를 붙일 것. 프로젝트 이름은 chapter6으로 하고, 패키지는 lib, app의 2개로 함

[결과 화면]



[프로그램 소스]

[lib >> Calculator]

package lib; //자동으로 해당 문구를 생성 시킴

public abstract class Calculator81 { //자동으로 추상클래스! (앞 설정에서 추상클래스 속성을 체크했으므로!)

```
    public abstract int add(int a,int b);
    public abstract int sub(int a, int b);
    public abstract double avg(int[] a);
}
```

[app >> GoodCalc81]

package app;
import lib.Calculator81;

```
public class GoodCalc81 extends Calculator81 {
    public int add(int a,int b) {
        return a+b;
    }

    public int sub(int a,int b) {
        return a-b;
    }

    public double avg(int[] a) {

        double sum=0;
        for(int i=0;i<a.length;i++) {
            sum+=a[i];
        }
        return sum/a.length;
    }
}
```

```

    }

    return sum/a.length;

}

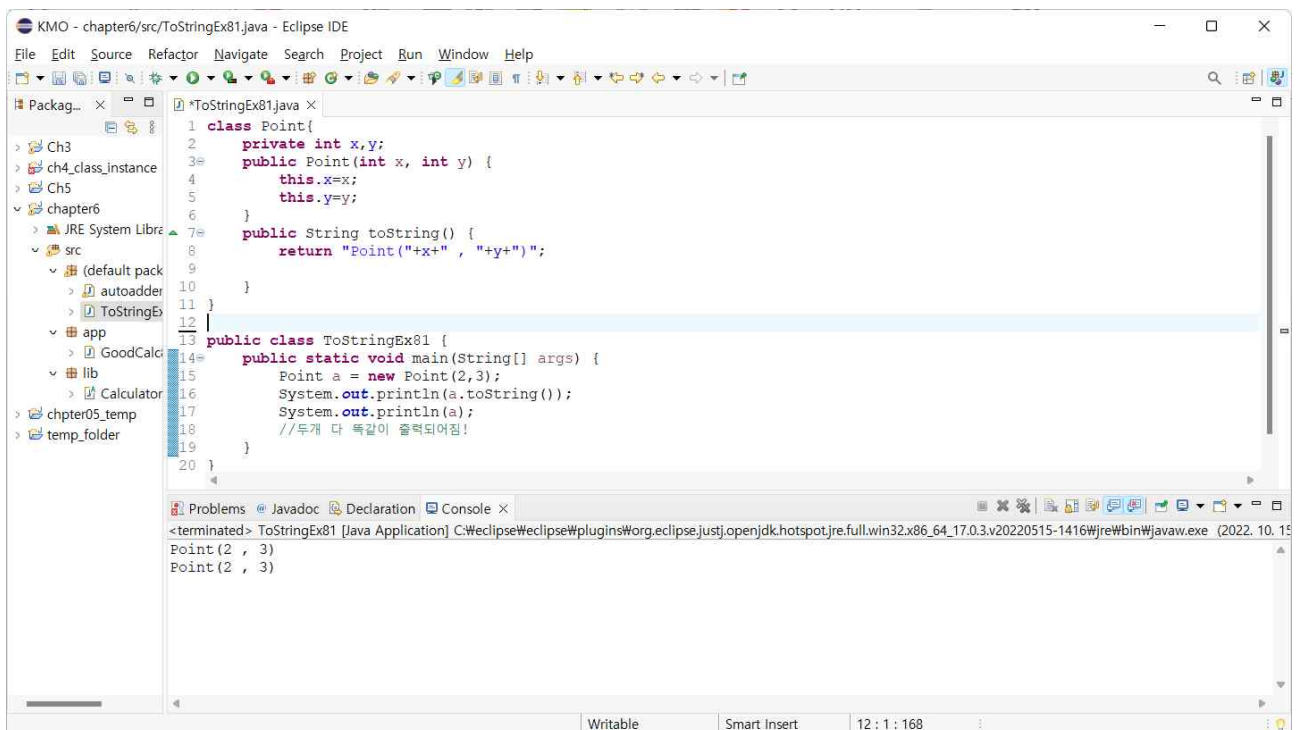
public static void main(String[] args) {
    Calculator81 c = new GoodCalc81();//오버라이딩? C#에서본듯?
    System.out.println(c.add(2, 3));
    System.out.println(c.sub(2, 3));
    System.out.println(c.avg(new int[] {2,3,4}));
}
}

```

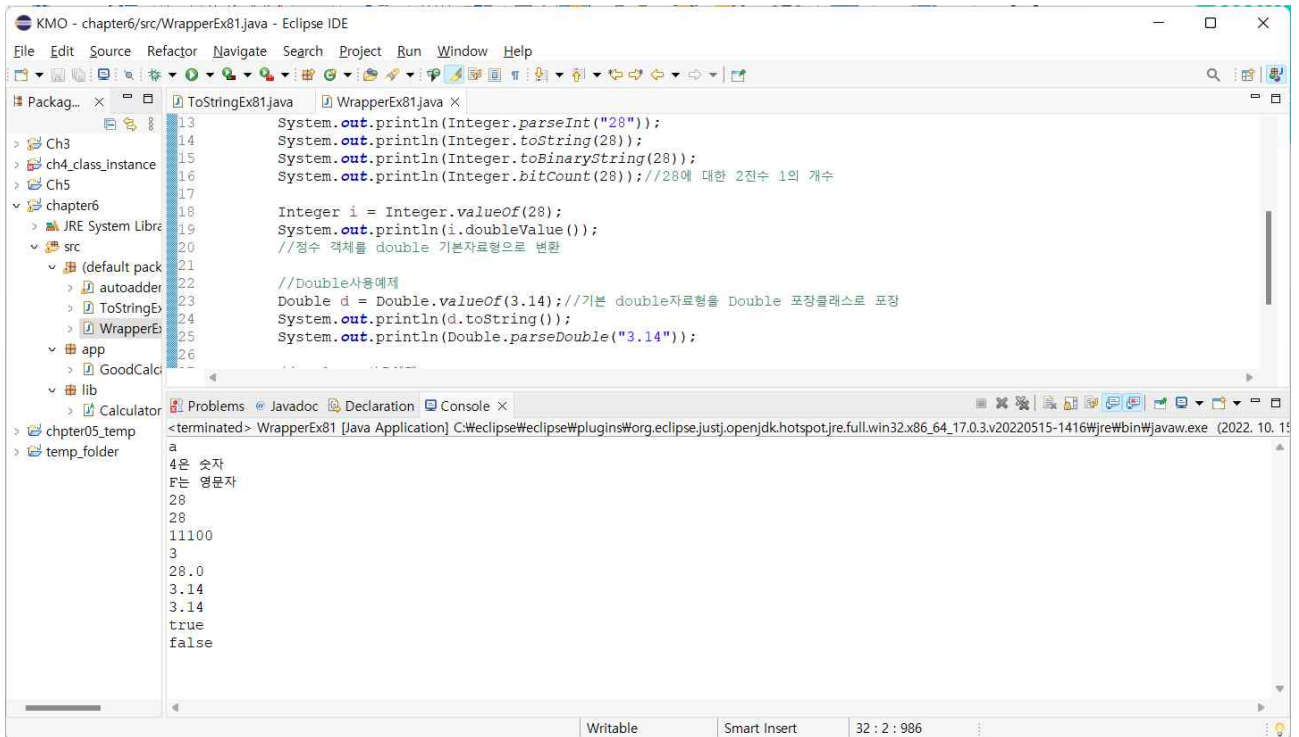
실습 6-2 예제 6-2, 6-5, 6-6, 6-7 각각을 이해한 후 실행하여 결과를 확인할 것. 클래스의 이름은 예제에 주어진 클래스 이름을 사용하되, 끝에 자기학번 끝자리 2자리 번호를 붙일 것.

[결과 화면]

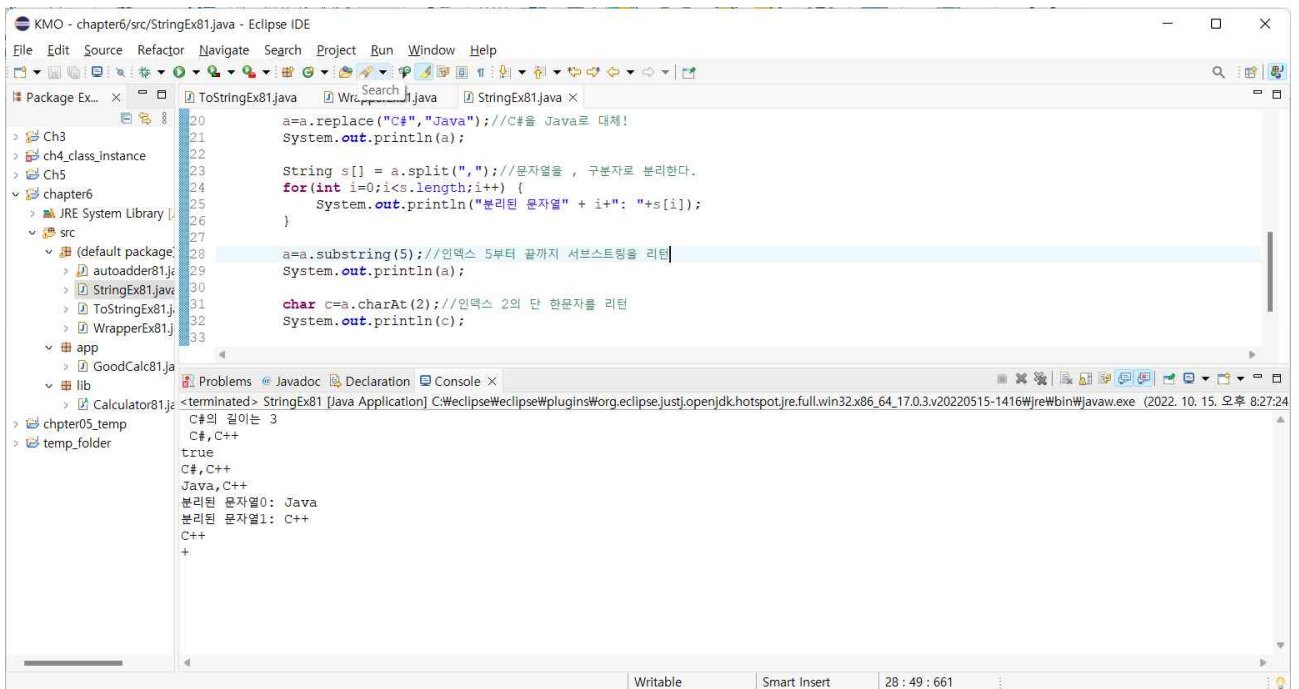
<예제 6-2>



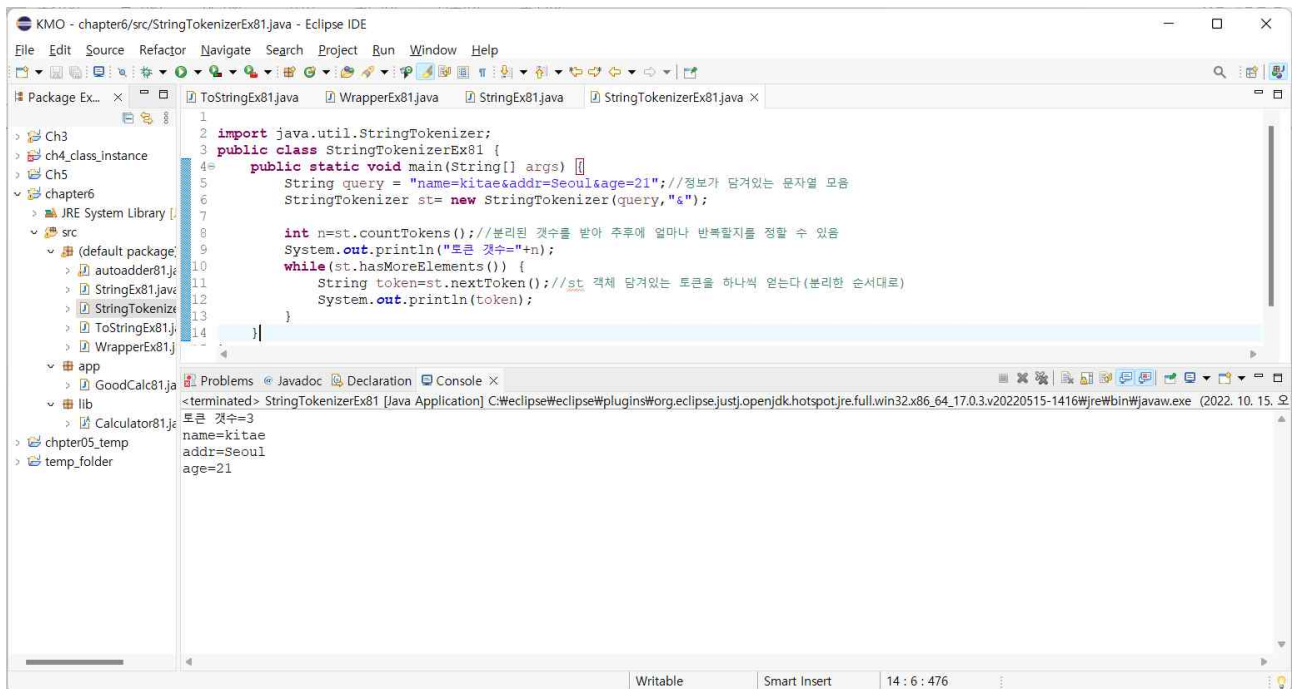
<예제 6-5>



<예제 6-6>



<예제 6-7>



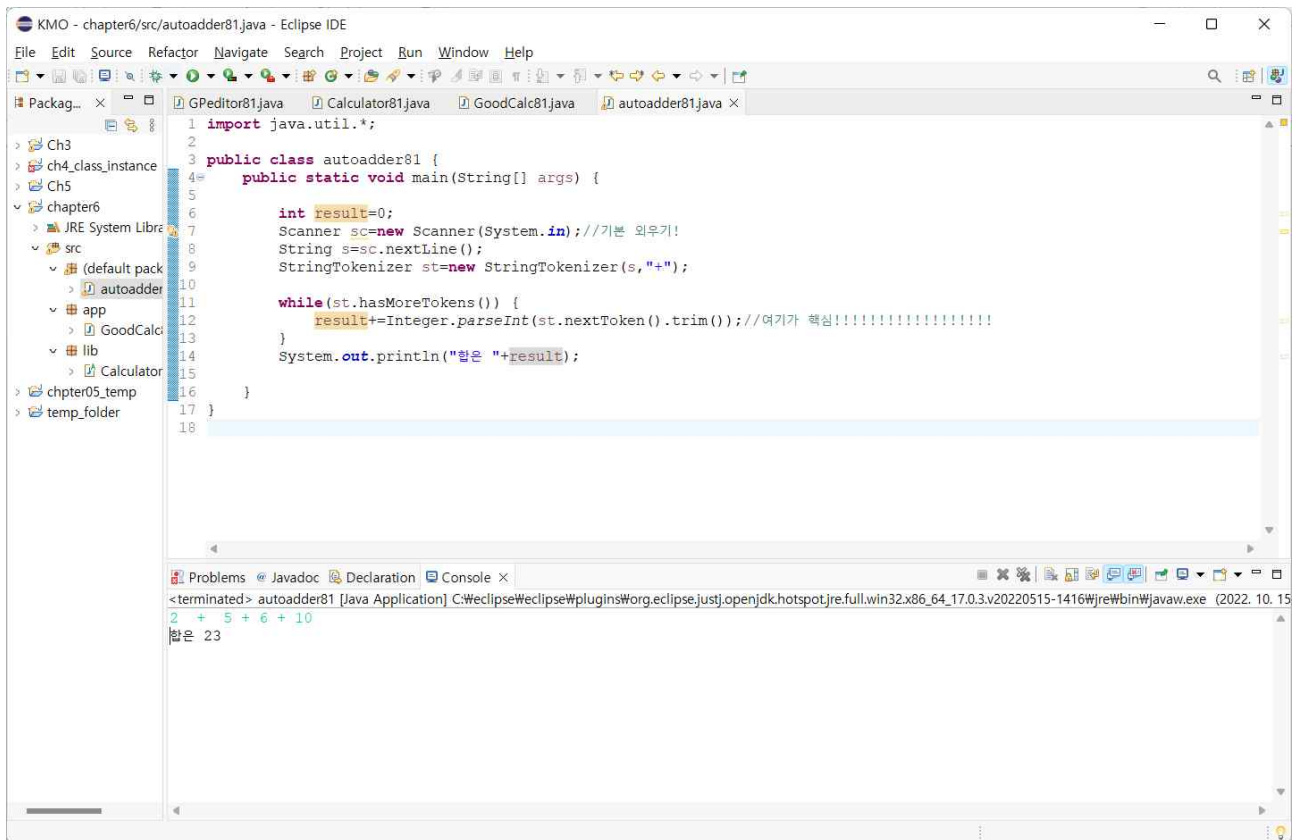
```
1 import java.util.StringTokenizer;
2 public class StringTokenizerEx81 {
3     public static void main(String[] args) {
4         String query = "name=kitae&addr=Seoul&age=21"; // 정보가 담겨있는 문자열 모음
5         StringTokenizer st= new StringTokenizer(query, "&");
6
7         int n=st.countTokens(); // 분리된 갯수를 받아 주후에 얼마나 반복할지를 정할 수 있음
8         System.out.println("토큰 갯수="+n);
9         while(st.hasMoreElements()) {
10             String token=st.nextToken(); // st 객체 담겨있는 토큰을 하나씩 얻는다 (분리한 순서대로)
11             System.out.println(token);
12         }
13     }
14 }
```

terminated> StringTokenizerEx81 [Java Application] C:\eclipse\workspace\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.3.v20220515-1416\jre\bin\javaw.exe (2022. 10. 15. 오후 14:06)

토큰 갯수=3
name=kitae
addr=Seoul
age=21

실습 6-3 p.280의 실습 문제 6-4를 수행할 것. 클래스 이름 끝에 자기학번 끝자리 2자리 번호를 붙일 것.

[결과 화면]



The screenshot shows the Eclipse IDE with the file 'autoadder81.java' open. The code is as follows:

```
1 import java.util.*;
2
3 public class autoadder81 {
4     public static void main(String[] args) {
5
6         int result=0;
7         Scanner sc=new Scanner(System.in); //기본 외우기!
8         String s=sc.nextLine();
9         StringTokenizer st=new StringTokenizer(s,"+");
10
11         while(st.hasMoreTokens()) {
12             result+=Integer.parseInt(st.nextToken().trim()); //여기가 핵심!!!!!!!!!!!!!!!!!!!!!!
13         }
14         System.out.println("합은 "+result);
15     }
16 }
17
18
```

The console output shows the result of the program execution:

```
<terminated> autoadder81 [Java Application] C:\eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_17.0.3.v20220515-1416\jre\bin\javaw.exe (2022. 10. 15)
2 + 5 + 6 + 10
합은 23
```

[프로그램 소스]

```
import java.util.*;

public class autoadder81 {
    public static void main(String[] args) {

        int result=0;
        Scanner sc=new Scanner(System.in); //기본 외우기!
        String s=sc.nextLine();
        StringTokenizer st=new StringTokenizer(s,"+");

        while(st.hasMoreTokens()) {
            result+=Integer.parseInt(st.nextToken().trim()); //여기가 핵심!
        }
        System.out.println("합은 "+result);

    }
}
```