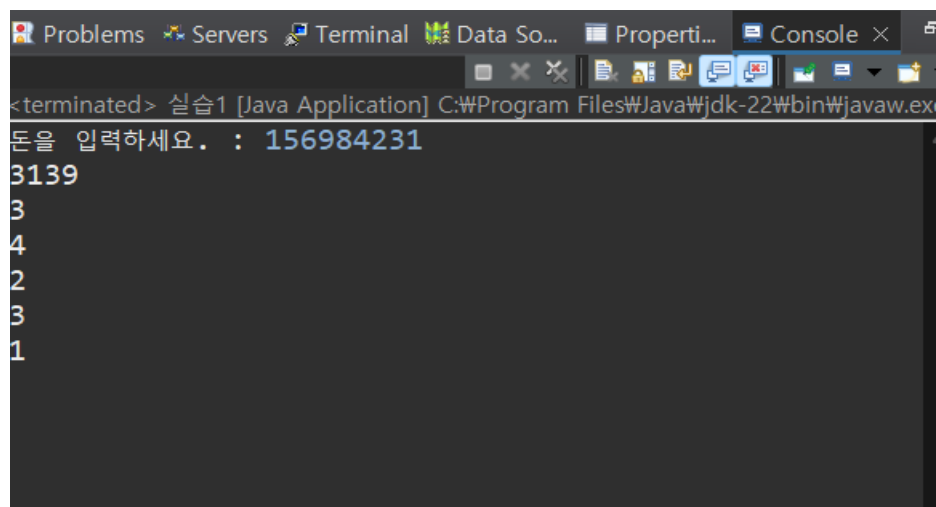


실습 1:

-실습코드

```
1 package 실습과제3주차;
2 import java.util.Scanner;
3
4 public class 실습1 {
5
6     public static void main(String[] args) {
7         // TODO Auto-generated method stub
8         int [] unit = { 50000,10000,1000,100,10,1};
9
10        Scanner scanner = new Scanner(System.in);
11        System.out.print("돈을 입력하세요. : ");
12        int money = scanner.nextInt();
13
14        for (int i=0;i<6;i++) {
15            System.out.println(money/unit[i]);
16            money=money%unit[i];
17        //        System.out.println();
18        }
19        scanner.close();
20
21    }
22
23 }
24
```

-결과출력



```
<terminated> 실습1 [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe
돈을 입력하세요. : 156984231
3139
3
4
2
3
1
```

-코드관련 설명

처음에 배열 unit을 {50000,10000,1000,100,10,1}으로 초기화합니다. "돈을 입력하세요 :"
라는 문구와 함께 스캐너로 돈에 대한 입력을 받은 뒤, 그 입력을 money라는 정수 변수
에 저장합니다.

이후 for 문을 사용해 오만 원권, 만 원권, 천 원권, 100원짜리 동전, 10원짜리 동전, 1원
짜리 동전이 각 몇 개로 반환되는지 출력하는 코드를 작성해 봅니다.

money 변수를 unit[0]로 나누고, 나눈 몫을 출력함과 동시에 줄을 띄우고, 나눈 나머지는
다시 money 변수가 됩니다. 이를 unit[1], unit[2], unit[3], ... unit[5]까지 반복하면 됩니다.

따라서 for문의 범위는 (int i=0;i<6;i++)이 됩니다.

For문 속의 코드는

```
System.out.println(money/unit[i]);
```

```
money=money%unit[i];
```

이렇게 됩니다.

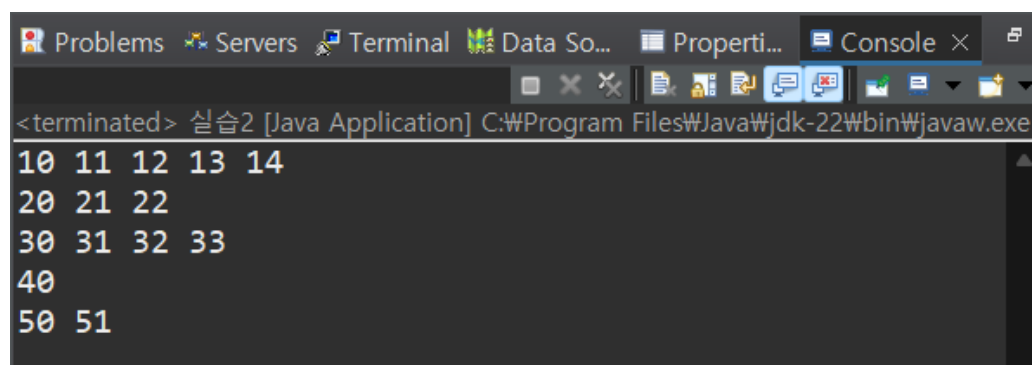
이후 Scanner 를 닫고, 클래스를 종료합니다.

실습 2:

-실습 코드

```
1 package 실습과제3주차;
2
3 public class 실습2 {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         int c[][];
8         c=new int[5][];
9         c[0]=new int[5];
10        c[1]=new int[3];
11        c[2]=new int[4];
12        c[3]=new int[1];
13        c[4]=new int[2];
14
15        for (int i=0 ; i< c.length ; i++) {
16            for (int j=0 ; j<c[i].length ;j++) {
17                c[i][j]=10*(i+1)+j;
18            }
19        }
20        for (int i=0 ; i<c.length ; i++) {
21            for (int j=0; j<c[i].length ;j++) {
22                System.out.print(c[i][j]+" ");
23            }
24            System.out.println();
25        }
26
27    }
28
29 }
```

-결과 출력



```
<terminated> 실습2 [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe
10 11 12 13 14
20 21 22
30 31 32 33
40
50 51
```

-코드관련 설명

비정방형 배열을 만들기 위해 우선 배열 `c[][]`를 선언합니다. `c=new int[5][]`로 배열의 행의 개수를 5개로 만들어 줍니다. 실습예제 그림을 살펴보면 0번째 행의 열의 개수는 5, 1번째 행의 열의 개수는 3, 2번째 행의 열의 개수는 4, 3번째 행의 열의 개수는 1, 4번째 행의 열의 개수는 2입니다. `c[0]=new int[5]`와 같은 방식으로 각 행의 열의 개수를 만들어 줍니다.

이후 for 문으로 배열 속 숫자를 채워 넣습니다. 먼저 행의 순서를 `i`라 할 때, `i`가 0부터 행의 마지막까지 반복되도록 만듭니다. 이를 for문으로 구현하면

```
for (int i=0 ; i< c.length ; i++).
```

이후 각 행마다 열의 숫자를 순서대로 채워 넣습니다.

이중 for 문을 이용하여 구현할 수 있고, 실습예제 그림의 숫자의 규칙을 살펴보면 $i*10$ 을 기준으로 열의 숫자가 각각 +0, +1, +2... 임을 알 수 있습니다. 각 행의 열의 개수가 정해졌기에 for문으로 구현하면

```
for (int j=0 ; j < c[i].length ;j++)
```

```
    c[i][j]=10*(i+1)+j;
```

그럼 배열에 숫자를 다 채웠습니다. 이제 해당 배열을 출력만 하면 됩니다.

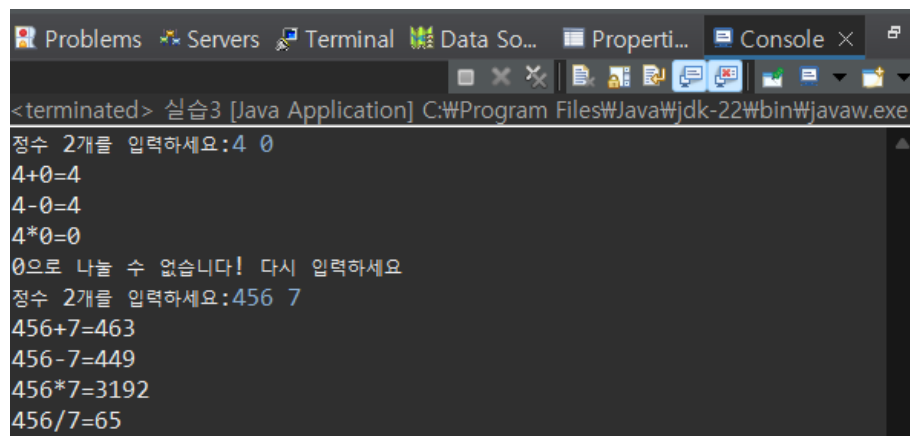
위에서 했던 이중 for문을 사용하여 각각 해당 행렬을 출력하고, 열과 열 사이에는 간격 " "를 더해주고, 행과 행 사이에는 `println`을 통해 줄 띄어쓰기를 해주면 됩니다.

```
for (int i=0 ; i<c.length ; i++) {  
    for (int j=0; j<c[i].length ;j++) {  
        System.out.print(c[i][j]+" ");  
    }  
    System.out.println();  
}
```

실습 3:

```
1 package 실습과제3주차;
2 import java.util.Scanner;
3 public class 실습3 {
4     public static int sum(int n, int m) {
5         return n + m;
6     }
7     public static int sub(int n, int m) {
8         return n - m;
9     }
10    public static int mul(int n, int m) {
11        return n * m;
12    }
13    public static int div(int n, int m) {
14        return n / m;
15    }
16
17    public static void main(String[] args) {
18        // TODO Auto-generated method stub
19        Scanner scanner = new Scanner(System.in);
20        while(true) {
21            System.out.print("정수 2개를 입력하세요:");
22            int a=scanner.nextInt();
23            int b=scanner.nextInt();
24            System.out.println(a+"+"+b+"="+ sum(a,b));
25            System.out.println(a+"-"+b+"="+ sub(a,b));
26            System.out.println(a+"*"+b+"="+ mul(a,b));
27            try {
28
29                System.out.println(a+"/"+b+"="+ div(a,b));
30
31                break;
32            }
33            catch(ArithmeticException e){
34                System.out.println("0으로 나눌 수 없습니다! 다시 입력하세요");
35            }
36
37        }
38
39        scanner.close();
40    }
41 }
42
43 }
```

-결과 출력



```
<terminated> 실습3 [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe
정수 2개를 입력하세요:4 0
4+0=4
4-0=4
4*0=0
0으로 나눌 수 없습니다! 다시 입력하세요
정수 2개를 입력하세요:456 7
456+7=463
456-7=449
456*7=3192
456/7=65
```

-코드 관련 설명

이전 과제에서 했던 sum,sub,mul,div 메소드를 만듭니다. 이후 public static void main(String[] args)로 try-catch문을 사용해 봅시다.

스캐너로 입력을 받습니다. "정수 2개를 입력하세요:"라는 문구와 함께 두 정수 int a, int b를 입력을 받습니다. 이후 두 a,b에 대한 sum,sub,mul,div 메소드를 출력하면 되는데, 문제는 b가 0일 때 발생합니다. 정수를 0으로 나눌 수 없기 때문입니다. 따라서 div 메소드를 출력하는 것을 try블록으로, try블록에서 0으로 나누는 예외가 발생한다면 그걸 catch블록으로 넘겨서 catch블록 안의 "0으로 나눌 수 없습니다! 다시 입력하세요"를 출력하게 만듭니다.

중요한 점은, catch 블록이 실행되면 다시 처음으로 돌아가서 int a, int a를 다시 입력받아야 한다는 점입니다.

따라서 입력 받는 부분부터 try, catch까지 전부 while(true)로 묶습니다. a,b입력을 받은 뒤, 먼저 sum,sub,mul 메소드에 대해 결과를 출력하고, div 메소드에 대해서는 try와 catch를 사용합니다. try 에서는, div(a,b)에서 b가 0이 아니라면 그대로 출력하고 while 문을 break합니다. 만약 b가 0이라면 catch(ArithmeticException e)를 사용해 "0으로 나눌 수 없습니다! 다시 입력하세요"를 출력하고 다시 while 문 속의 내용들을 처음부터 반복하면 됩니다.

Scanner 를 닫고, 클래스를 종료합니다.

+)여기서 입력 a,b가 정수가 아닌 float라면, div(a,b)에서 에러가 나지 않는 걸 발견할 수 있습니다. 출력시 infinity라고 출력되는 것을 볼 수 있습니다.