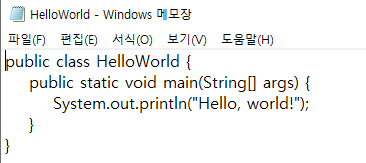
**1주차 JAVA LAB 과제**

20234014 컴퓨터공학과 이은정

1. Hello world 출력해보기

Hello World를 출력하는 데는 2가지 방법이 있다. 첫번째로 메모장으로 코드를 작성하고, CMD에서 자바 가상머신을 사용하여 java 파일을 실행하는 방법이 있다.

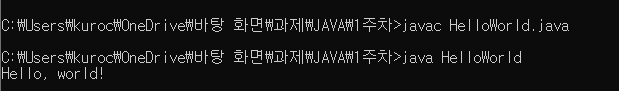
먼저 메모장에

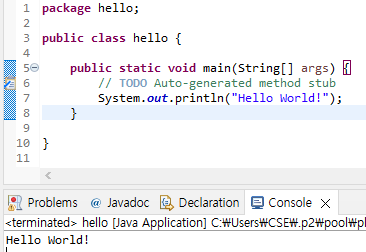


이 처럼 메모장에 일일히 모든 코드를 작성해준다. 이 때 파일명과 클래스명을 동일하게 하지 않을 경우 아래와 같은 오류가 발생할 수 있다.



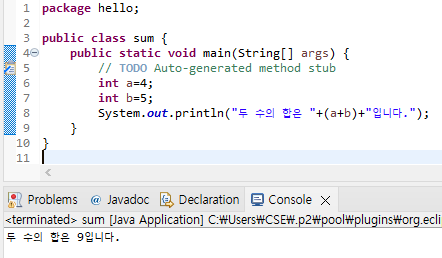
제대로 코드를 작성한 이후에는 해당 파일을 바이트 코드로 컨버팅 해준다. 그리고 새로 생긴 클래스 파일을 java 가상 머신을 통해 실행시켜주면 Hello, World!를 출력할 수 있다.



두 번째는 eclipse에서 코드를 작성하는 방법이 있다. 코드를 작성하고 출력하면 옆의 사진과  
같이 eclipse 프로그램 내에서 결과를 확인할  
수 있다.

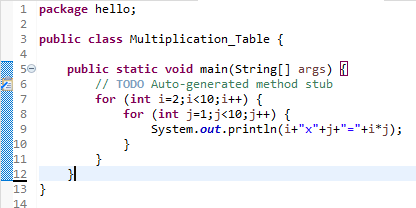
2. 두수의 합 출력하기

c언어와 동일하게 코드를 작성한다. a, b 변수에 각각의 수를 저장할 것이므로 int a, b에 각각의 수를 대입한다. JAVA에서는 println()형으로 코드를 작성하는 것이 아니라, 어디의 패키지에 해당 함수가 있는지를 같이 적어주어야 한다. 따라서 System.out.println()을 통해 해당 값을 출력한다. 문자열과 변수를 연결해서 출력하기 위해서는 +를 통해 서로 값을 연결해준다.



3. 반복문을 이용해서 구구단을 출력하기

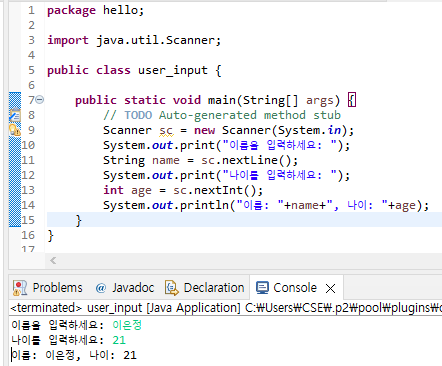
M x N 형태가 반복되는데 N이 1~9까지 반복 이후 M이 1씩 값이 커진다. 따라서 이중 for문을 사용해 코드를 구현한다. J가 한 바퀴 루프를 마치면 i값을 증가시키는 형식으로 코드를 작성한다.



결과는 왼쪽과 같이 제대로 출력됨을 알 수 있다.

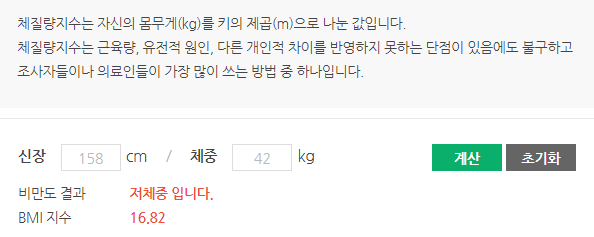
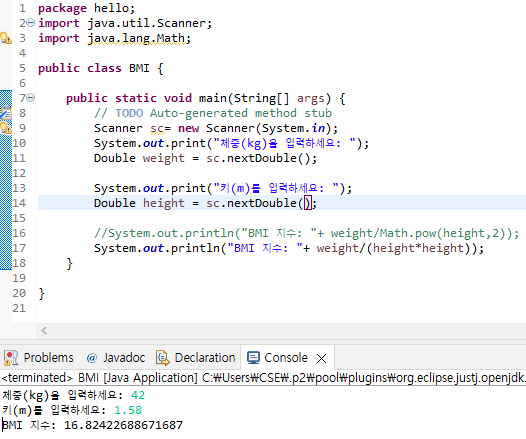
5. 사용자로부터 값을 입력 받아 출력하기

New Scanner(System.in)을 통해 값을 입력 받는다. 이 때 주의할 점은 변수형과 함수에서 next변수형이 동일해야 한다. 출력할 때 일반 문자열과 변수는 +를 통해 연결한다.



5. BMI 계산기 만들기

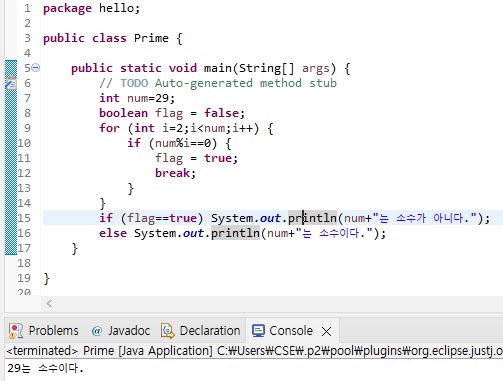
값을 입력 받는 부분이 나온다. 값을 입력 받는 함수는 new Scanner(System.in)이다. c언어와 마찬가지로 변수형 + 변수명 형태로 변수를 만들고 next변수형 함수를 통해 값을 입력 받는다. 변수의 변수형과 next 뒤에 나오는 변수형이 다르면 오류가 발생한다. 키/(몸무게 제곱) 이므로 java.lang.Math 패키지의 함수를 사용해서 제곱 값을 구할 수도 있고, 간단하게 height\*height를 통해 값을 구할 수도 있다.

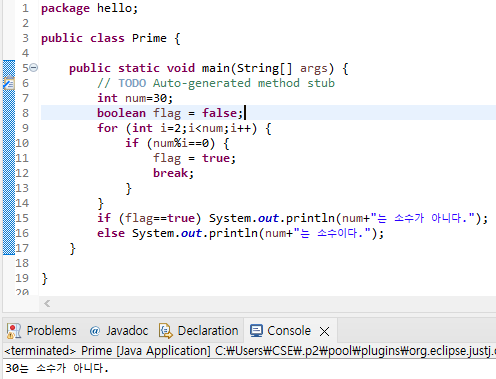


네이버 계산기와 비교했을 때 값이 동일하므로 코드가 제대로 구현되었음을 알 수 있다.

6. 29가 소수인지 아닌지 판별하기

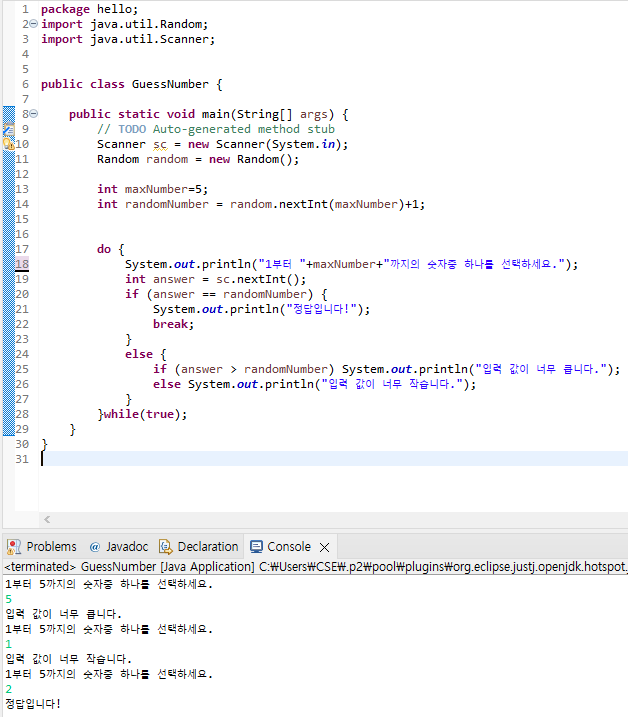
소수는 1과 본인 자신의 수로만 나누어 떨어지는 수이다. 즉슨 소수를 판별하기 위해서는 2에서 n-1까지의 수로 나누었을 때 나머지가 0이 아니면 된다. scanner함수를 통해 값을 입력 받은 이후에 이 수를 2~n-1까지의 수로 각각 나눈다. 이 때 한 번이라도 나머지의 값이 0이 나올 경우 해당 수는 소수가 아니게 된다. 이는 Boolean값으로 판별한다. 만약 한 번이라도 나머지 값이 한 번이라도 0이 아닐 경우 Boolean을 false에서 true로 바꾼다. N-1까지의 수로 나누고 난 이후에 Boolean 값을 검사하여 Boolean이 true라면 해당 수는 소수가 아니라고 판별한다. 따라서 코드는 아래와 같다.



6-2 수를 직접 입력 받는 경우도 위의 알고리즘에 따라 코드를 작성한다. 29가 소수인지를 검사할 때에는 n-1이 28로 고정되어 있었으나 값을 입력 받는 경우 n-1은 고정값이 아니게 된다. 따라서 입력 받는 값을 n으로 하고, n-1까지 for문을 돌려 소수인지 검사한다. 

6. 숫자 맞추기

랜덤한 숫자를 하나 변수에 저장하고, 답변을 하나 입력 받는다. 랜덤한 숫자와 비교하여 답변의 크기가 숫자보다 크면 “값이 크다”를, 작으면 “작다”를 입력받는다. 해당 숫자 맞추기 게임은 맞출 때 까지 반복한다. 이 알고리즘을 만족시키기 위해서 new Random()을 통해 랜덤한 값을 g나 넣는다. Random(maxNumber)+1을 통해서 0~maxNumber-1까지의 범위를 1~ maxNumber까지로 변경해준다. nextInt();를 통해 예상 숫자를 입력받는다. 그리고 randomNumber과 입력 값을 비교한다. If == 를 통해 비교해주고, 만약 if문을 만족한다면 break하여 루프를 벗어나고, 게임을 끝낸다. 그것이 아니라면 각각의 값을 비교한다. 이 과정은 정답을 맞출 때 까지 반복해야 하므로 무한반복으로 처리한다.



위 코드처럼 while문을 통해 무한반복 처리를 한다. 만약 정답일 경우에만 빠져나오도록 break문을 실행하고, 빠져나오면 게임은 종료된다. 위 코드는 실험을 하기 위해 maxNumber을 작게 만들었지만 원코드는 100으로 maxNumber을 수정하면 된다.