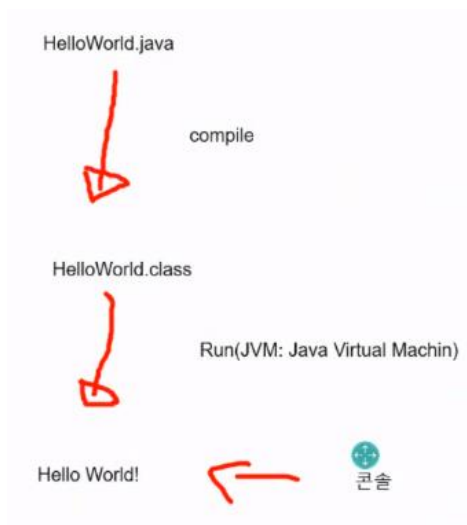


4/14 스터디노트

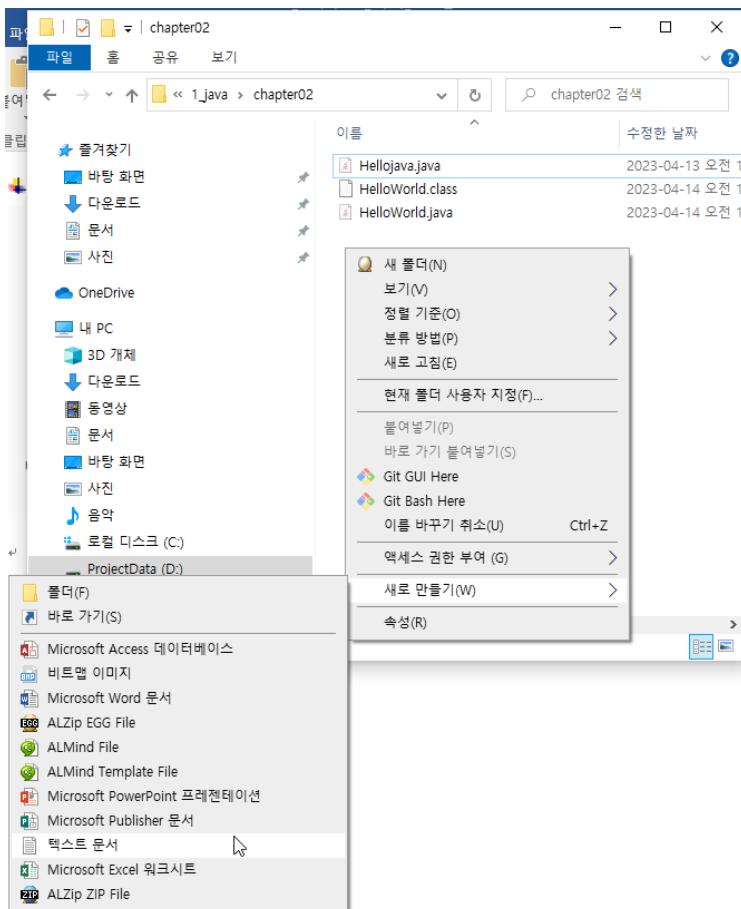
자바의 빌드과정



코드를 컴파일하면 .class 가 되고 실행을 시키면 콘솔에 실행이 됨

자바 수동 빌드 해보기

1. 파일탐색기에서 텍스트 파일 만들기 : 제목은 HelloWorld



```
*Hellojiwon.txt - Windows 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
public class HelloWorld{
    public static void main(String[] args){
        System.out.println("Hello World!");
    }
}
```

텍스트 파일 내용은 오른쪽 사진처럼 java 코드를 타이핑 : 오타 있으면 안됨!!

클래스 명과 파일 명은 동일하게 해야함~!

2. 텍스트 파일 제목 변경하기를 통해서 확장자를 바꿈(.txt -> .java)
3. win+r -> cmd 실행하기
4. 우리는 D 드라이브에 파일이 있으므로 경로를 C에서 D로 바꿔야함

'd:' 입력 후 엔터 `C:\Users\wtj-bu>d:`

5. `D:\>cd D:\choijiwon\Study_Project\1_WebProject\2_BackEnd\1_java\chapter02`

'cd' 입력 후 내가 텍스트 파일 만든 곳 경로를 복붙 후 엔터

6. `D:\choijiwon\Study_Project\1_WebProject\2_BackEnd\1_java\chapter02>javac HelloWorld.java`

'javac HelloWorld.java' 입력 후 엔터

7. `D:\choijiwon\Study_Project\1_WebProject\2_BackEnd\1_java\chapter02>dir`

'dir' 입력 후 엔터

8. 파일 정보 나오는 것 확인

```
D:\choijiwon\Study_Project\1_WebProject\2_BackEnd\1_java\chapter02 디렉터리
2023-04-14 오전 10:08 <DIR> .
2023-04-14 오전 10:08 <DIR> ..
2023-04-13 오전 11:17      142 Hellojava.java
2023-04-14 오전 10:08     426 HelloWorld.class
2023-04-14 오전 10:01     112 HelloWorld.java
                3개 파일             680 바이트
                2개 디렉터리 437,779,243,008 바이트 남음
```

9. 'java HelloWorld' 라고 java + 파일제목 입력하면 아래와 같이 코드 실행됨

```
D:\choijiwon\Study_Project\1_WebProject\2_BackEnd\1_java\chapter02>java HelloWorld
Hello World!
D:\choijiwon\Study_Project\1_WebProject\2_BackEnd\1_java\chapter02>
```

결론 : 옛날에는 이렇게 개발을 했지...

옛날 개발자는 오타 하나만 있어도 실행이 안 됐음.. 실수 소지가 많음

✚ P.61-62 산술연산 ; 정수(int) 끼리의 연산은 정수로 나옴.

- 실수로 연산 결과가 나오게 하고 싶다면 강제 형변환 해야함

```
public static void main(String[] args) {  
  
    int a = 10;  
    int b = 4;  
    System.out.println("10/4="+a/b);  
    System.out.println("10/4="+((double)a/b);  
  
}
```

10/4=2 10/4=2.5

- 부호연산 (p.62)

- ✓ +는 '현재 부호 그대로 유지' 의 의미이고
- ✓ -는 '현재 부호를 반대로 바꿈' 의 의미.
- ✓ 예) `int b = -10;`

```
System.out.println(+b);  
System.out.println(-b);
```

-10 10

✚ 증감연산 (p.63)

- 전위연산 : 변수 **앞**에 연산자 ex) ++a (=a변수에 1더하기)

- ✓ 전위연산은 현재 라인에서 변수 값을 변화시킴 (지금 당장부터)

- 후위연산 : 변수 **뒤**에 연산자 ex) a++ (=a변수에 1더하기)

- ✓ 후위연산은 다음 라인에서 변수 값을 변화시킴 (다음 라인부터 a는 1을 더한 값으로 사용)

```
int a = 10;  
int b = 10;
```

```
System.out.println(++a);  
System.out.println(b++);  
System.out.println(b);
```

11 10 11

a는 전위연산을 시행했으므로 결과값이 바로 전위연산이 시행된 11로 나옴 b는 후위연산을 시행했으므로 연산 결과값이 다음 줄에서 반영됨
--

비교연산 (p.65)

- 좌항 과 우항을 비교하는 연산자. 연산결과는 True 혹은 False로 나옴. (boolean자료형)
- $A \geq B$: A는 B보다 크거나 같다. -> ' $A \geq B$ ' 로 표현함
- $A \leq B$: A는 B보다 크거나 같다. -> ' $A \leq B$ ' 로 표현함
- $A = B$: A는 B와 같다. -> ' $A == B$ ' 로 표현함 (=가 1개 있으면 대입연산자라서 2개 써야함)
- $A \neq B$: A는 B와 같지 않다. -> ' $A != B$ ' 로 표현함

```
int a = 10;  
int b = 5;  
System.out.println(a>b);  
System.out.println(a==b);  
System.out.println(a!=b);
```

```
true  
false  
true
```

```
int i = 4;  
int i2 = 5;  
System.out.println(i=i2); //i에 i2 값을 대입한다는 의미이므로 5가 나옴
```

```
5
```

```
boolean c = a==b; //c는 논리 자료형으로 10=5 연산결과인 False 값이 c에 저장됨  
boolean d = c==false; //d는 논리 자료형으로 c=false 결과인 True 값이 d에 저장됨  
System.out.println(c);  
System.out.println(d);
```

```
false  
true
```

논리연산 (p.66)

● 표현

- ✓ 좌항 [논리연산자] 우항 으로 표현함
- ✓ 좌항이나 우항은 변수, 상수, 표현식, 비교연산이 올 수 있음

● 연산결과표 (진리표)

종류	A항	연산자	B항	결과	설명
AND	true	&& (&)	true	true	양쪽 항 모두 true인 경우만 결과값 true
	true		false	false	
	false		true	false	
	false		false	false	
OR	true	()	true	true	두 항 중 하나 이상의 항이 true인 경우 결과값 true
	true		false	true	
	false		true	true	
	false		false	false	
XOR	true	^	true	false	두 항 중 하나의 항만 true인 경우 결과값 true
	true		false	true	
	false		true	true	
	false		false	false	
NOT		!	true	false	반대의 값
			false	true	

- ✓ AND(그리고) : 엄격한 조건임. 좌항과 우항 양쪽 모두가 True 일때만 True결과가 나옴
- ✓ OR(또는) : 유연한 조건임. 양쪽 중에 하나만 True 있어도 True결과가 나옴
- ✓ XOR : 둘 중 하나 고르기. 양쪽이 달라야(하나는 True, 하나는 False) True결과가 나옴
- ✓ NOT : 우항만 사용함. 우항값의 반대값을 결과로 내보냄.

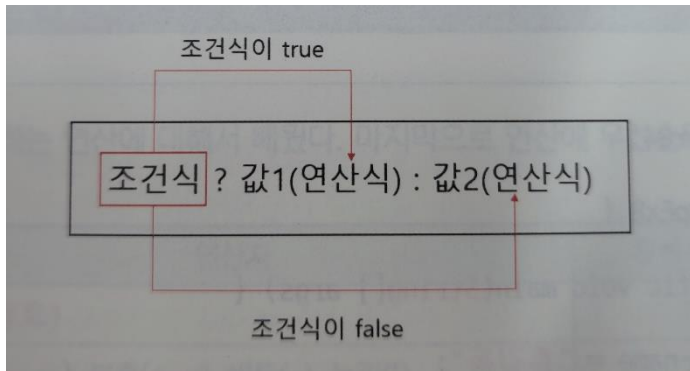
➤ 계산방식 (p.68~p.70 내용 요약)

- ✓ 논리연산은 왼쪽에서 오른쪽으로 계산함 (좌항부터 계산)
- ✓ 모든 우선순위 중에 가장 순위가 높은 연산자는 (). 따라서 ()부터 먼저 계산
- ✓ &&연산의 경우 둘다 True여야 True가 나오기 때문에
첫번째 조건에서 False가 나온 경우 두번째 조건을 계산하지 않고 바로 false 결과를 내보냄
- ✓ 그렇다면 &를 1개만 써서 식을 쓰면?
위처럼 하지 않고 첫번째 결과가 무엇이든 두번째 연산식도 꼭 계산함.
이때 두번째가 [미지의 값]÷[입력값] 같은 모양일 때 [입력값]이 0이 되는 경우
수학적으로 존재할 수 없는 계산식이 되므로 runtime 에러가 발생함.
어쨌든 에러 발생 방지와 시간 단축과 효율을 위해서 && 이렇게 2개씩 써야함~

✚ P.69, p.70-p.74 는 스킵함~!

✚ 삼항연산 (p.75)

- 3개 항으로 이루어진 연산임. 조건의 결과에 따라서 어느 값을 내보낼지 결정함.



- ✓ 예시: `score >= 60 ? "합격" : "불합격"` => score 가 60 이상이면 합격, 60미만이면 불합격

✚ 문자열연산 (p.76)

- + : 앞의 값과 뒤의 값을 붙여줌.

- ✓ 예시: `"나는 " + name + " 입니다"` => `나는 [name변수의값] 입니다.`

- 문자 자료형 변수에는 숫자를 대입할 수 없음

- ✓ 예시: `String weight = 75.7;` => 에러남

`String weight = 75.5 + "";` => 75.5가 숫자가 아니라 문자가 됨

연산자 우선순위 (p.77)

- () 괄호가 가장 우선됨
 - 산술연산의 우선순위
 - 왼쪽에서 오른쪽으로 계산하되, *(곱셈), /(나눗셈) 은 +, - 보다 먼저 계산함.
- ✓ 예시 : $5 + 4 * 3 = 17$ / $(5 + 4) * 3 = 27$

c 언어의 경우에는
컴파일러 종류에 따라 27이 나올수도 있음

- 연산자가 여러 개 있을 때 이미 p.77의 연산자 우선순위가 정해져있지만
그래도 항상 ()로 우선순위를 표현하는 것이 논리적 에러를 방지할 수 있음★

교재 연습문제 답 풀이 (p.79~81)

2. 3

3. 풀지마세요

4. 카톡방에 보낸 템플릿에 작성해보세요 (Excercise4Template.java).

내 답안

```
public static void main(String[] args) {  
  
    //문제에서 요구하는 변수를 선언하고 변수에 적용될 수 있는 값을 초기화해 보세요.  
    int age;  
    float height;  
    boolean parent;  
    boolean heartDease;  
  
    age=5;  
    height=130;  
    parent=true;  
    heartDease=false;  
  
    //boolean take = 탑승 가능 여부에 대한 조건식을 작성하세요.  
    boolean take = ((age>6 || parent==true) && height>120 && heartDease==false);  
  
    //탑승 가능 여부에 대해 true/false로 출력하게 해보세요.  
    System.out.println(take);  
}
```

4번 선생님 답안

```
public static void main(String[] args) {  
  
    int age = 6;  
    int height = 130;  
    boolean parent = false;  
    boolean heartDease = true;  
  
    boolean take = ((age >= 6 && height >= 120) || (height >= 120 && parent == true)) && heartDease == false;  
  
    System.out.println(take);  
}
```

heartDease 여부가 True 이면 무조건 탑승 불가이므로
heartDease 여부를 가장 먼저 계산하도록 코드를 짜면 시간효율적임.
그러나 놀이동산에 오는 인원중 heartDease 여부가 false인 확률이
훨씬 높으므로 오히려 heartDease를 먼저 계산하는게 비효율적일 수도 있음

문제 풀이 팁

- 문장에서 '값'을 표현하는 명사 (키, 부모님여부, 심장질환여부, 나이 등) => 변수, 상수가 되는 대상.

범위 표현

- A가 B를 초과할 경우, A가 B보다 큰 경우 => $A > B$
- A가 B 미만일 경우, A가 B보다 작은 경우 => $A < B$
- A가 B 이상일 경우, A가 B보다 같거나 큰 경우 => $A \geq B$
- A가 B 이하일 경우, A가 B보다 같거나 작은 경우 => $A \leq B$

5. 내 답안

```
public class P80_Excercise5 {  
    new *  
    public static void main(String[] args) {  
        int year = 2024;  
        boolean leapYear = year%100!=0 && ( year%4==0 || year%400==0 ) ;  
  
        System.out.println(leapYear);  
    }  
}
```

선생님 답안

```
public static void main(String[] args) {  
  
    int year = 2020;  
  
    boolean leapYear = (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0);  
  
    // 윤년이면 true, 윤년이 아니면 false  
    System.out.println(leapYear);  
  
}
```

6. 내 답안

```
int price = 187000;  
int oman = price/50000 ;  
int ilman = (price%50000)/10000;  
int ochun = (price%10000)/5000;  
int ilchun = (price%5000)/1000;  
  
System.out.println("5만원권 : "+oman+"장");  
System.out.println("1만원권 : "+ilman+"장");  
System.out.println("5천원권 : "+ochun+"장");  
System.out.println("1천원권 : "+ilchun+"장");
```

선생님 답안

```
int price = 187000;  
int oman = price / 50000;  
int ilman = price % 50000 / 10000;  
int ochun = price % 10000 / 5000;  
int ilchun = price % 5000 / 1000;  
  
System.out.println("5만원권 : "+oman+"장");  
System.out.println("1만원권 : "+ilman+"장");  
System.out.println("5천원권 : "+ochun+"장");  
System.out.println("1천원권 : "+ilchun+"장");
```

7. 내 답안

```
int number = 1234;  
int result = number / 100 * 100 ;  
System.out.println(result);
```

선생님 답안

```
int number = 1234;  
int result = number / 100 * 100;  
System.out.println(result);  
  
int result2 = number - number % 100;  
System.out.println(result2);
```

실습

- 인텔리제이 단축키 Ctrl + Shift + enter = 현재 행 끝에 ; 붙이기
- 콘솔에서 숫자값을 입력받는 함수
 - ✓ import java.util.Scanner;
 - ✓ int num1 = scanner.nextInt(); // 숫자 입력 받는 함수(프로그램, 기능)

```
import java.util.Scanner;
new *
public class ScannerTest {
    new *
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("첫 번째 정수를 입력하세요: ");
        int num1 = scanner.nextInt();
        System.out.println("입력하신 숫자는 "+num1+"입니다.");
    }
}
```

첫 번째 정수를 입력하세요:

→ 결과창에 숫자를 입력해보자

첫 번째 정수를 입력하세요:
50

→ 엔터

첫 번째 정수를 입력하세요:
50
입력하신 숫자는 50입니다.

: 콘솔창에 내가 입력값을 넣을 수 있음

- println : 메시지를 출력하고 줄바꿈
- print : 메시지만 출력

- 입력값 2개 해보기

```
import java.util.Scanner;
new *
public class ScannerTest {
    new *
}    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("첫 번째 정수를 입력하세요: ");
        int num1 = scanner.nextInt();
        System.out.print("두 번째 정수를 입력하세요: ");
        int num2 = scanner.nextInt();

        int result = num1 * num2;
        System.out.println(num1+" * "+num2+" = "+result);
    }
}
```

➔ 콘솔창에 차례대로 아무 숫자를 2개 넣어보자

```
첫 번째 정수를 입력하세요: 10
두 번째 정수를 입력하세요: 7
10 * 7 = 70
```

🌈 선생님 연습문제

- 문제1) 두 개의 정수를 입력받아 합과 곱을 출력하는 프로그램을 작성하세요.

* 화면 출력 예시

첫 번째 정수를 입력하세요: 10

두 번째 정수를 입력하세요: 5

두 정수의 합: 15

두 정수의 곱: 50

```
package chapter04;
import java.util.Scanner;
new *
public class Q1 {
    new *
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("첫 번째 정수를 입력하세요: ");
        int num1 = scanner.nextInt();
        System.out.print("두 번째 정수를 입력하세요: ");
        int num2 = scanner.nextInt();

        int result1 = num1+num2;
        System.out.println("두 정수의 합:"+result1);

        int result2 = num1*num2;
        System.out.println("두 정수의 곱:"+result2);
    }
}
```

첫 번째 정수를 입력하세요: 20
두 번째 정수를 입력하세요: 2
두 정수의 합:22
두 정수의 곱:40

➤ 문제2) 두 개의 정수를 입력받아 사칙연산 결과를 출력하는 프로그램을 작성하세요.

* 화면 출력 예시

첫 번째 정수를 입력하세요: 10

두 번째 정수를 입력하세요: 5

10 + 5 = 15

10 - 5 = 5

10 * 5 = 50

10 / 5 = 2

10 % 5 = 0

```
첫 번째 정수를 입력하세요: 5
두 번째 정수를 입력하세요: 5
5 + 5 = 10
5 - 5 = 0
5 * 5 = 25
5 / 5 = 1
5 % 5 = 0
```

```
package chapter04;
import java.util.Scanner;

new *
public class Q2 {
    new *
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("첫 번째 정수를 입력하세요: ");
        int num1 = scanner.nextInt();
        System.out.print("두 번째 정수를 입력하세요: ");
        int num2 = scanner.nextInt();

        int result1 = num1+num2;
        System.out.println(num1+" + "+num2+" = "+result1);

        int result2 = num1-num2;
        System.out.println(num1+" - "+num2+" = "+result2);

        int result3 = num1*num2;
        System.out.println(num1+" * "+num2+" = "+result3);

        int result4 = num1/num2;
        System.out.println(num1+" / "+num2+" = "+result4);

        int result5 = num1%num2;
        System.out.println(num1+" % "+num2+" = "+result5);
    }
}
```

➤ 문제3) 두 개의 정수를 입력받아 큰 수를 출력하는 프로그램을 작성하세요.

* 화면 출력 예시

첫 번째 정수를 입력하세요: 7

두 번째 정수를 입력하세요: 8

입력받은 두 정수 중에서 큰 수는 8입니다.

```
package chapter04;
import java.util.Scanner;
new *
public class Q3 {
    new *
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("첫 번째 정수를 입력하세요: ");
        int num1 = scanner.nextInt();
        System.out.print("두 번째 정수를 입력하세요: ");
        int num2 = scanner.nextInt();

        int result1 = num1>num2 ? num1:num2;
        System.out.println("입력받은 두 정수 중에서 큰 수는 "+result1+"입니다.");
    }
}
```

첫 번째 정수를 입력하세요: 5 두 번째 정수를 입력하세요: 1 입력받은 두 정수 중에서 큰 수는 5입니다.
--

➤ 문제4) 하나의 정수를 입력받아 짝수/홀수 여부를 판단하는 프로그램을 작성해 보세요.

* 화면 출력 예시

정수를 입력하세요: 678

678은(는) 짝수입니다.

```
package chapter04;
import java.util.Scanner;
new *
public class Q4 {
    new *
}    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

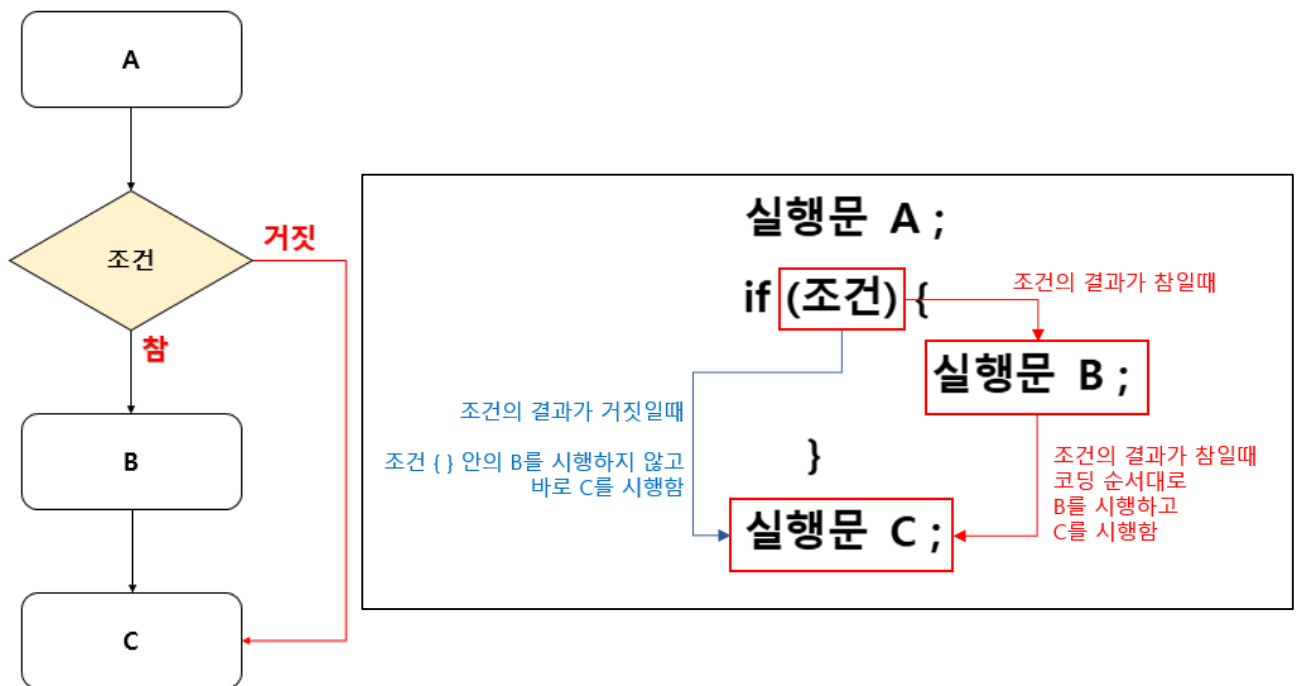
        System.out.print("정수를 입력하세요: ");
        int num1 = scanner.nextInt();

        String result1 = num1%2==0 ? "짝수":"홀수";
        System.out.println(num1+"은(는) "+result1+"입니다.");
    }
}
```

정수를 입력하세요: 15
15은(는) 홀수입니다.

Chapter 05 제어문

프로그램의 실행제어 (p.84)



- ➔ 만약 조건이 참이라면 B를 수행하고 C를 수행함.
하지만 조건이 거짓이라면 B를 건너뛰고 C를 수행함.

- 참 : 'True' 혹은 '1' 혹은 5, 10, 100 등 거짓이 아닌 모든 값
- 거짓 : 'False' 혹은 '0'
- 조건 : 값, 변수, 조건식 등이 들어갈 수 있음.
이 조건의 결과가 true일때 if문 안의 실행문 B를 실행함.
- 값 = 리터럴. (예시: true, false, 1, 0, 1000... 등)

- if 문 연습해보기

```
public class ex {  
    new *  
    public static void main(String[] args) {  
        //조건이 그 자체값이 True 인 경우  
        if(true){  
            System.out.println("참입니다.");  
        }  
    }  
}
```

참입니다.

```
public class ex {  
    new *  
    public static void main(String[] args) {  
        //조건이 변수인 경우  
        boolean condition1 = false;  
        if(condition1) {  
            System.out.println("거짓입니다.");  
        }  
    }  
}
```

조건이 false 이므로
조건문 안의 실행문을 실행시키지 않고
아무 실행 없이 코드 종료되어
아무결과도 나오지 않음

```
public class ex {  
    new *  
    public static void main(String[] args) {  
        //조건이 연산식인 경우  
        int num1 = 100;  
        if (num1 == 100) {  
            System.out.println("참입니다.");  
        }  
    }  
}
```

참입니다.

```

public class ex {
    new *
    public static void main(String[] args) {
        //조건이 논리 연산식인 경우
        int num2 = 10;
        int num3 = 5;
        if((num2==10)&&(num3==5)){
            System.out.println("참입니다.");
        }
    }
}

```

참입니다.

- ➔ if (조건식1 [논리연산자] 조건식2 [논리연산자] ...)
- ➔ 개별 조건식은 요구사항으로 해석할 수 있다.

● if 예제 (p.86)

```

public class P86_IfEx {
    new *
    public static void main(String[] args) {
        int score=80;
        System.out.println("시험 시작");
        if (score >= 60) {
            System.out.println("합격입니다.");
        }
        System.out.println("시험 종료");
    }
}

```

시험 시작
합격입니다.
시험 종료

- if else 예제 (P.87)

```
public class P87_IfEx1 {  
    new *  
    public static void main(String[] args) {  
        int score=50;  
        System.out.println("시험 시작");  
        if (score >= 60) {  
            System.out.println("합격입니다.");  
        } else {  
            System.out.println("불합격입니다.");  
        }  
        System.out.println("시험 종료");  
    }  
}
```

시험 시작 불합격입니다. 시험 종료

➔ if 조건인 `score >= 60` 의 결과가 `false` 라면
 `else { }` 안에 있는 실행문을 실행한다.

- else if 예제 (p.88)

```
public static void main(String[] args) {
    int score = 80;
    String grade = "";

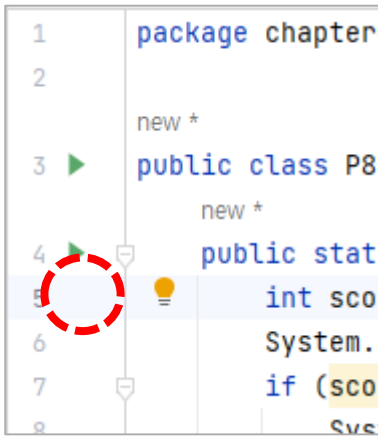
    System.out.println("학점부여 시작");
    if (score >= 95) {
        grade = "A+";
    }
    else if (score >=90) {
        grade = "A";
    }
    else if (score >=85) {
        grade = "B+";
    }
    else if (score >=80) {
        grade = "B";
    }
    else if (score >=70) {
        grade = "C";
    }
    else if (score >=60) {
        grade = "D";
    }
    else {
        grade = "F";
    }

    System.out.println("당신의 학점은 "+grade+" 입니다.");
    System.out.println("학점부여 끝");
}
```

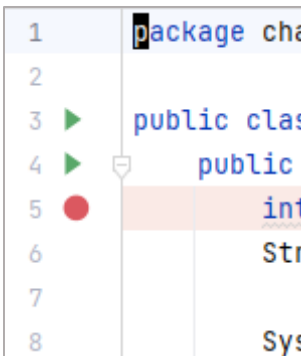
선행 if 와 else if 가 모두 false 일때 시행

- ➔ 첫번째 if 부터 시행함. 첫번째 if 조건은 false 이므로 첫번째 if { } 무시
- ➔ 두번째 if 를 시행함. 두번째 if 조건도 false 이므로 두번째 if { }도 무시
- ➔
- ➔ 네번째 if 를 시행함. 네번째 if 조건이 true 이므로 네번째 if { } 시행하여 grade 변수에는 'B' 가 대입됨.
- ➔ 네번째 if 조건이 true 였으므로 아래의 if들은 모두 무시함
- ➔ if 문 바깥의 실행문 (빨간점선네모) 부분만 시행하고 코드 종료됨.

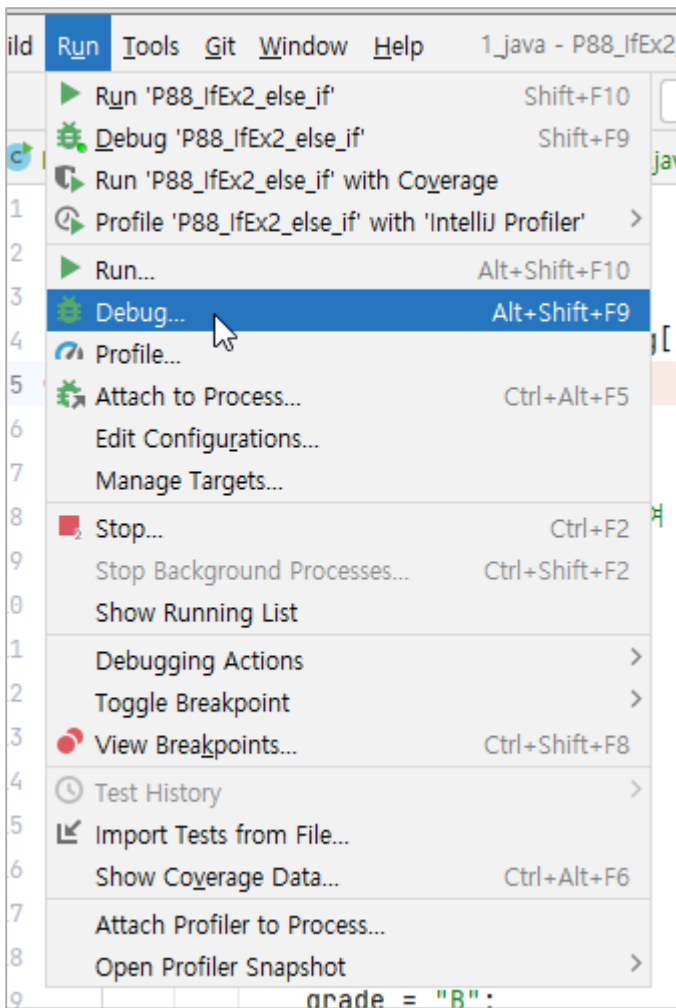
디버깅 해보기



-> 행 번호 숫자 옆의 빈공간을 클릭(브레이크포인트 걸기)

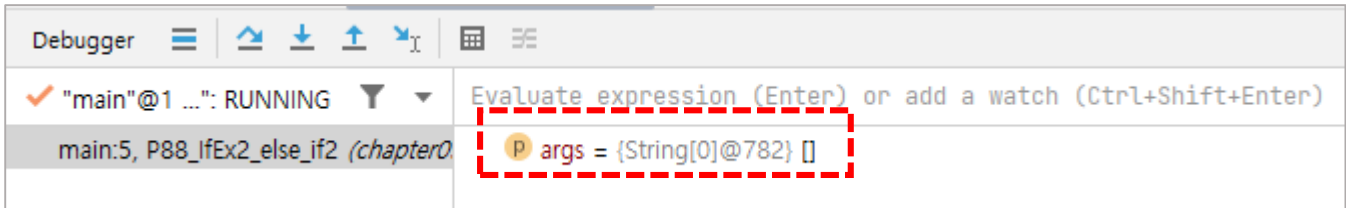


-> 클릭한 곳에 빨간 점이 생김

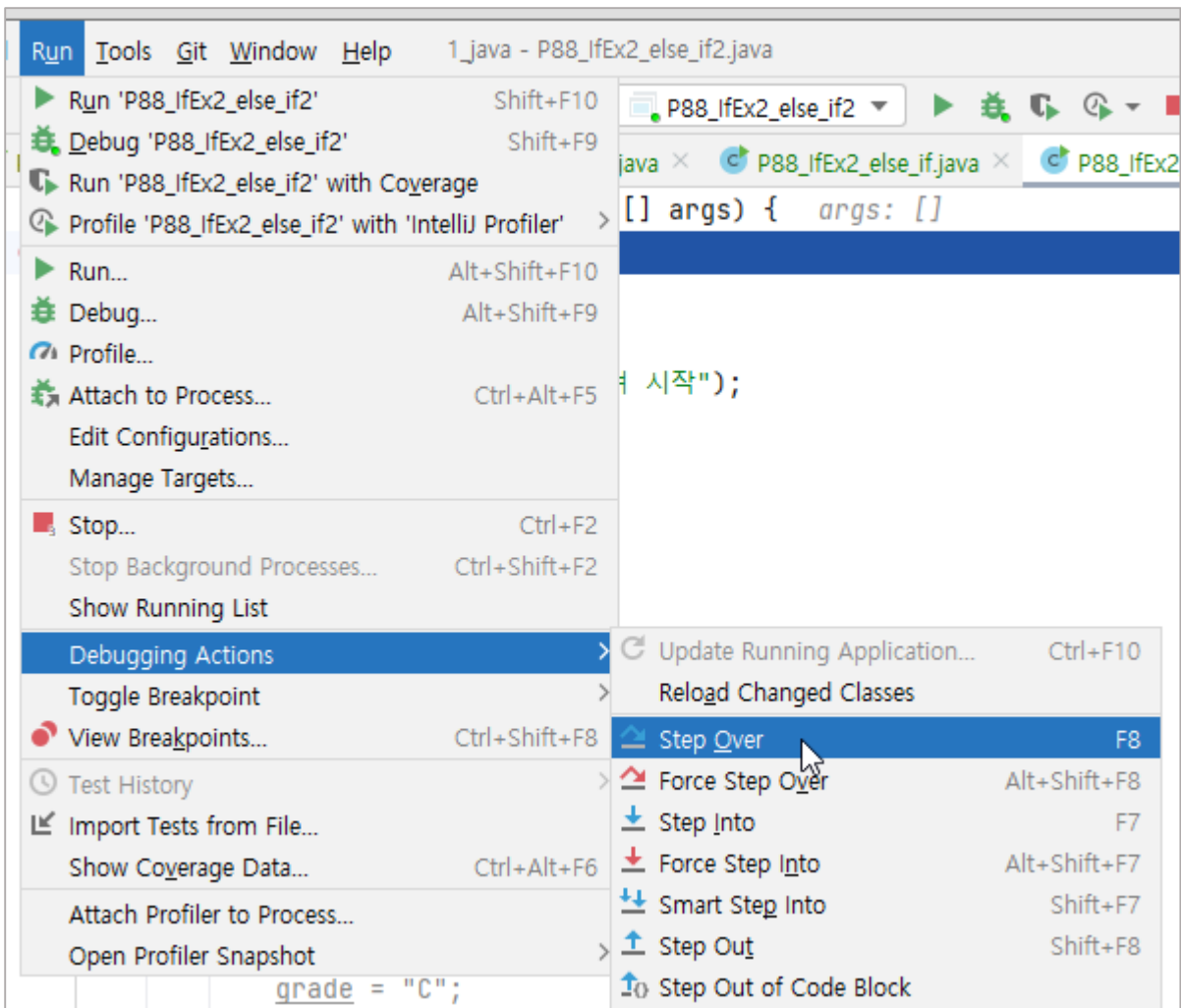


-> 메뉴에서 디버그선택(단축키 Alt+Shift+F9)

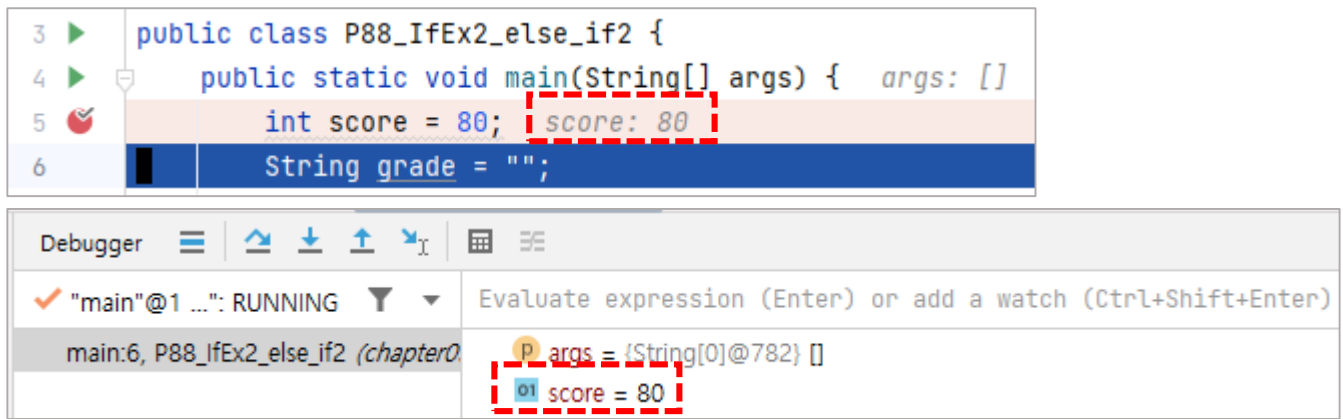
```
1 package chapter05;
2
3 public class P88_IfEx2_else_if2 {
4     public static void main(String[] args) { args: []
5         int score = 80;
6         String grade = "";
```



➔ 해당 행 바로 위까지 선언된 변수의 값이 설명됨



➔ 메뉴에서 다음줄 디버깅 하기를 선택(단축키 F8)



➔ 그 다음줄 디버깅 결과가 나타남 (변수 값이 설명됨)

score변수의 값이 80으로 초기화 되어있으므로

if (score >= 80) 부분에서 if { } 안의 실행문 grade="B" 가 시행되면서
grade변수의 값이 B가 되고

아래의 else if 문은 건너뛰면서 종료됨.

F8를 계속 눌러서 한줄 한줄 디버깅 결과를 보고,
콘솔에 어떻게 코드결과가 수행되는지 확인해보기

한 줄 씩 디버깅 => 코드의 시행 결과를 중간중간에 알 수 있음!

<단축키 정리>

Run 의 단축키가 처음 시행하는 경우 Alt+Shift+F10 인 것처럼

디버그도 처음 시행하는 경우 Alt+Shift+F9 로 시작하며,

처음 디버그를 시행하고 나서 다음줄 디버깅은 F8 을 눌러서 진행함