




23.03.06 홍윤기

—◇—  
1강 과제



# 목 차

1강 수업내용 정리

1강 과제

## 자바의 특징

- 표준적인 기본 문법
- 객체지향 프로그래밍
- 범용성 : 운영체제에 무관하게 실행

단점 : 느리다

## 프로그래밍의 3가지 스텝

1. 프로그램의 작성
2. 컴파일
3. 실행

소스코드 작성

소스코드 : 사람이 읽을 수 있는 프로그램

## 컴파일

1. 컴퓨터는 CPU가 프로그램을 해석해서 작동<sub>2</sub>.
2. 기계어로 소스코드 변환
3. 컴파일 처리하여 바이트코드로 상태 변환(1,0)
4. 오류 검사
5. 컴파일러



온라인 IDE

[repl.it](https://repl.it)

[inlinegdb.com](https://inlinegdb.com)

## 컨벤션

코드 작성 양식

팀 단위 작업이므로 스타일을 통일해야함

\_ 사용하면 안됨 : c, Python과 다름

링크 :

<https://google.github.io/styleguide/javaguide.html>

번역 : <https://newwisdom.tistory.com/96>



## 컨벤션

- 정확하게 기술
    - 대소문자 구별
  - 밖에서 안으로
  - 가독성 좋게
    - 한줄충 아웃
    - 주석 적극 사용
    - 메소드 내부 :
      - 변수선언 - 계산 - 실행
- 요즘은 선언과 동시에 값을 주는 편

## 변수의 선언과 대입을 동시에 해야하는 이유

```
int a ;
```

```
a = 20;
```

```
int a = 20;
```

- 실수가 있을 수 있다

⇒ 변수의 초기화가 안되어도 디버깅  
과정에서 잡히지 않아 오류를 찾기 어렵다.

# 변수 선언

## 변수

데이터를 저장하기 위해 내부에 준비하는 공간 : 메모리

## 타입

변수의 유형

## 예약어

변수명으로 사용하면 안되는 키워드

ex) int, public, void, static, class 등

# 변수 선언 규칙

알파벳, 숫자 - 코딩 컨벤션 따를 것

변수명 중복 하면 안됨

대소문자 구별

소문자로 시작하는 알기쉬운 이름 :

다음 단어의 첫문자는 대문자, 띄어쓰기, 언더바 사용하지 말 것  
ex) myName, myAge

# 데이터 타입

유형	자료형		크기	범위
정수형	char	"A"	2byte	문자형 (0 ~ 65535)
	byte	100	1byte	(-128 ~ 127)
	short	3515	2byte	(-32768 ~ 32767)
정수형(default)	int	120000	4byte	(-2,147,483,648 ~ 2,147,483,647)
	long	1000000000l	8byte	(-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807)
실수형	float	12.5f	4byte	
실수형(default)	double	12.5	8byte	
논리형	boolean	true		참
		false		거짓
문자열	String	String a = "adadsfadf"	레퍼런스(참조형)	

## 데이터 타입

**실수 : double**

코딩 컨벤션상 데이터타입의 리터럴은 소문자가 표준

**문자 : String**

한글자 변수라도 char 보단 String 사용

**논리 : boolean**

int(0,1) 사용 하지 말것



## 부동소숫점 오류

```
public class Main {  
    public static void main(String[] args) {  
        double myDouble1 = 10.1;  
        double myDouble2 = 10.2;  
  
        System.out.println(myDouble1+myDouble2);  
    }  
}
```

20.29999999999999997

...Program finished with exit code 0  
Press ENTER to exit console.

# 부동소수점 오류



부동소수점 오류(Floating point error)는 컴퓨터에서 실수를 표현하는 방식 중 하나인 부동소수점 방식에서 발생할 수 있는 오류입니다. 부동소수점 방식에서는 실수를 가수(mantissa)와 지수(exponent)로 표현합니다. 이때 가수와 지수는 이진수로 표현되며, 실수의 정확도는 가수의 비트 수와 지수의 범위에 따라 결정됩니다.

하지만 부동소수점 방식에서는 이진법으로 정확히 표현할 수 없는 일부 실수들이 존재합니다. 이는 소수점 이하의 이진수 표현이 무한 소수인 경우에 해당합니다. 이러한 실수들을 컴퓨터에서 근사값으로 표현하게 되는데, 이 때 근사값과 실제 값 사이에 오차가 발생할 수 있습니다. 이러한 오차는 작을 수도 있지만, 때로는 큰 오차가 발생할 수도 있습니다.

예를 들어, 0.1을 2진수로 정확히 표현할 수 없기 때문에, 컴퓨터에서는 근사값인 0.100000001490116119384765625로 표현됩니다. 따라서 이 근사값을 계산에 사용하는 경우, 정확한 결과를 얻지 못할 수 있습니다.

부동소수점 오류는 특히 과학, 공학 및 금융 분야에서 실수를 다룰 때 문제가 될 수 있습니다. 이러한 경우에는 부동소수점 방식 대신에 고정소수점 방식 등 다른 방식을 사용하여 정확한 결과를 얻을 수 있습니다. 또는 부동소수점 방식을 사용할 경우에는 근사값 오차를 최소화하는 방법을 사용하여 오차를 줄일 수 있습니다.

## 상수

변경할 필요가 없는 수  
final을 이용해 선언 - 대문자와 언더바(\_)를 사용  
축약어 사용하면 안된다.  
변하면 안되는 변수의 값을 변경했을 때  
(인간의 실수 - 변수를 착각)

## 계산문(계산식)

변수나 값을 이용하여 여러가지 계산을 수행  
대입도 계산의 일종

## 구성요소

연산자  
피연산자  
리터럴

# 리터럴

데이터타입의 상세를 나타냄  
컨벤션 : 소문자로 표기

float	f
long	l
char	'
String	"
16진수	0x
2진수	0b
자릿수 구분자	_



## 이스케이프 시퀀스

\"	이중따옴표(\"")
\'	따옴표 (')
\\	역슬래시/won (\)
\n	개행 문자



# 계산(EVALUATION)

## 계산결과 치환 법칙

계산 결과만 출력

## 우선순위 원칙

정해진 우선 순위가 높은 순서대로 계산

## 결합규칙 원칙

연산자별로 정해진 방향으로 순서대로 계산

# 연 산 자

연산자	설명	사용법
+	덧셈	(문자열은 연결)
-	뺄셈	
*	곱셈	
/	나눗셈	(정수연산은 몫)
%	나머지	
++	1증가	a++(출력후 증가), ++a(증가후 출력)
--	1감소	a--(출력후 감소), --a(감소후 출력)
+=	더해서 대입	a += 2
-=	빼서 대입	a -= 2
*=	곱해서 대입	a *= 2
/=	나눠서 대입	a /= 2
%=	나머지 후 대입	a %= 2

## 대입 연산자

출력과 동시에 연산하지 말 것  
연산이 끝난 결과를 출력|

# 증감 연산자

Java

```
public class Main {  
    public static void main(String[] args) {  
        int a = 10;  
        int b = 10;  
        System.out.println(a++); //10  
        System.out.println(a);    //11  
        System.out.println(++b); //11  
    }  
}
```

# 형 변환(CAST)

## 묵시적 형 변환

```
Main.java ×
1 package com.example;
2
3 public class Main {
4     public static void main(String[] args) {
5         double d = 3;
6         String S = "베스트 " + d;
7         System.out.println(S);
8     }
9 }
10
```

Problems Javadoc Declaration Console ×

<terminated> Main [Java Application] C:\Users\admin\Downloads\

베스트 3.0

1. 실수(double)에 정수(int) 넣어도 자동 형변환
2. 실수 값 d가 자동으로 String화 됨.

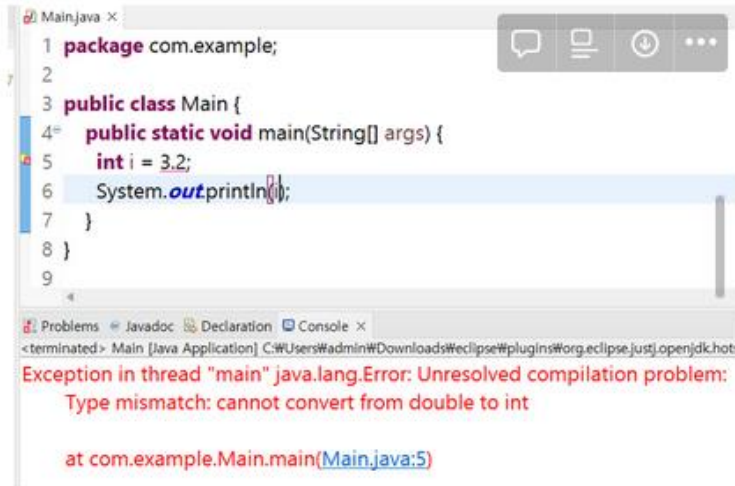
상위 변수 타입에 하위 변수 타입을 넣으면 자동으로 변환

# 형 변환(CAST)

## 명시적 형 변환

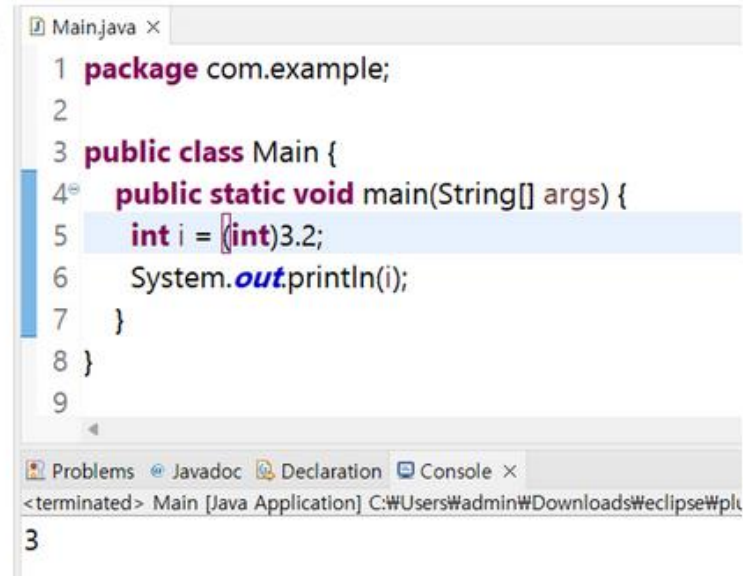
강제적인 형 변환

하위 변수 타입에 상위 변수 타입을 넣는 경우



```
1 package com.example;
2
3 public class Main {
4     public static void main(String[] args) {
5         int i = 3.2;
6         System.out.println(i);
7     }
8 }
9
```

Exception in thread "main" java.lang.Error: Unresolved compilation problem:  
Type mismatch: cannot convert from double to int  
at com.example.Main.main(Main.java:5)



```
1 package com.example;
2
3 public class Main {
4     public static void main(String[] args) {
5         int i = (int)3.2;
6         System.out.println(i);
7     }
8 }
9
```

좌우 피연산자는 동일한 타입이어야함

용량이 큰 타입으로 자동 형 변환 됨.



# 형 변환(CAST)

용량이 큰 타입으로 자동 형 변환 됨.

```
Main.java ×  
1 package com.example;  
2  
3 public class Main {  
4     public static void main(String[] args) {  
5         double d = 8.5 / 2;  
6         long l = 5 + 2l;  
7         System.out.println(d);  
8         System.out.println(l);  
9     }  
    
```

Problems Javadoc Declaration Console ×  
<terminated> Main [Java Application] C:\Users\admin\DownloadsWe  
4.25  
7

# 난수 생성

```
Main.java ×
1 package com.example;
2
3 public class Main {
4     public static void main(String[] args) {
5         int r = new java.util.Random().nextInt(90);
6         System.out.println(r);
7     }
8 }
9
```

Problems @ Javadoc Declaration Console ×

<terminated> Main [Java Application] C:\Users\admin\Downloads\weclipse

82

h 미만 범위로 선언 가능

# 난수 생성

범위를 지정하는 방법 (a이상 b 미만)

```
Main.java ×
1 package com.example;
2
3 public class Main {
4     public static void main(String[] args) {
5         // 5~ 10 까지
6         int rand = new java.util.Random().nextInt(6);
7         rand += 5;
8         // -5~ 5 까지
9         int rd = new java.util.Random().nextInt(11) - 5;
10        System.out.println(rd);
11    }
12 }
13
```

Problems Javadoc Declaration Console ×

<terminated> Main [Java Application] C:\Users\admin\Downloads\ eclipse\plug

-1

# 키보드 입력받기

```
1 package com.example;
2
3 import java.util.Scanner;
4
5 public class Main {
6     public static void main(String[] args) {
7         System.out.println("이름 입력 : ");
8         String name = new Scanner(System.in).nextLine();
9         System.out.println("나이 입력 : ");
10        int age = new Scanner(System.in).nextInt();
11        System.out.println(name + ", " + age);
12    }
13 }
```

Problems @ Javadoc Declaration Console ×

<terminated> Main [Java Application] C:\Users\admin\Downloads\ eclipse\plugins\org.

이름 입력 :

홍윤기

나이 입력 :

27

홍윤기, 27

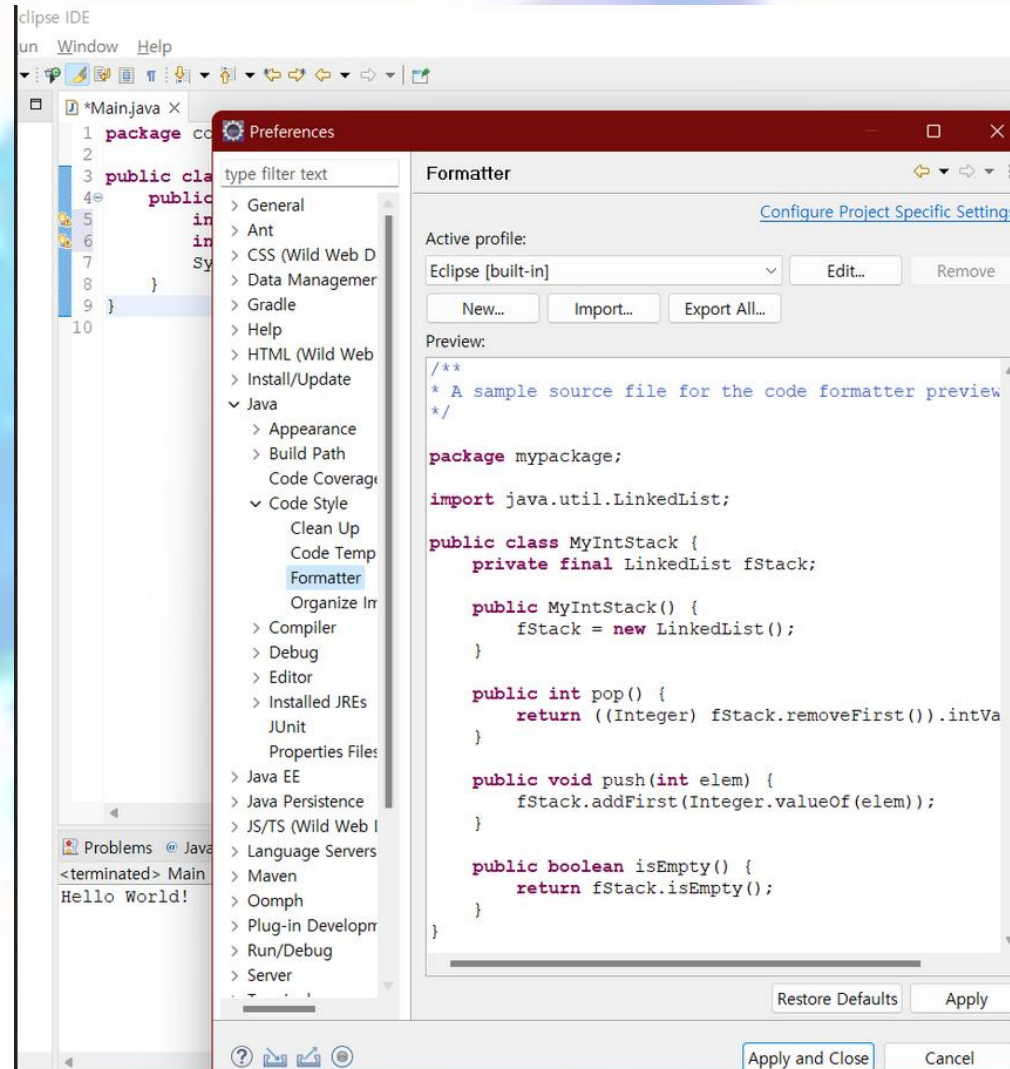
BufferedReader 사용하는 방법도 있음



이클립스 실습

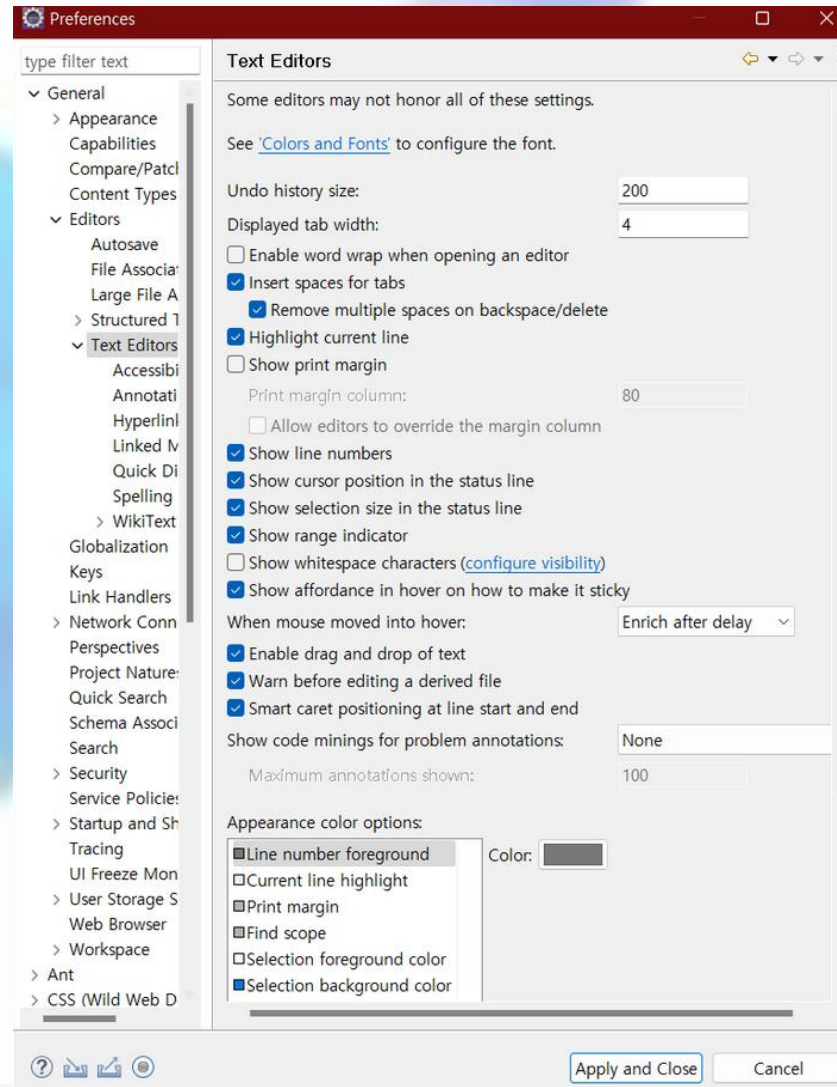


# FORMATTER 적용

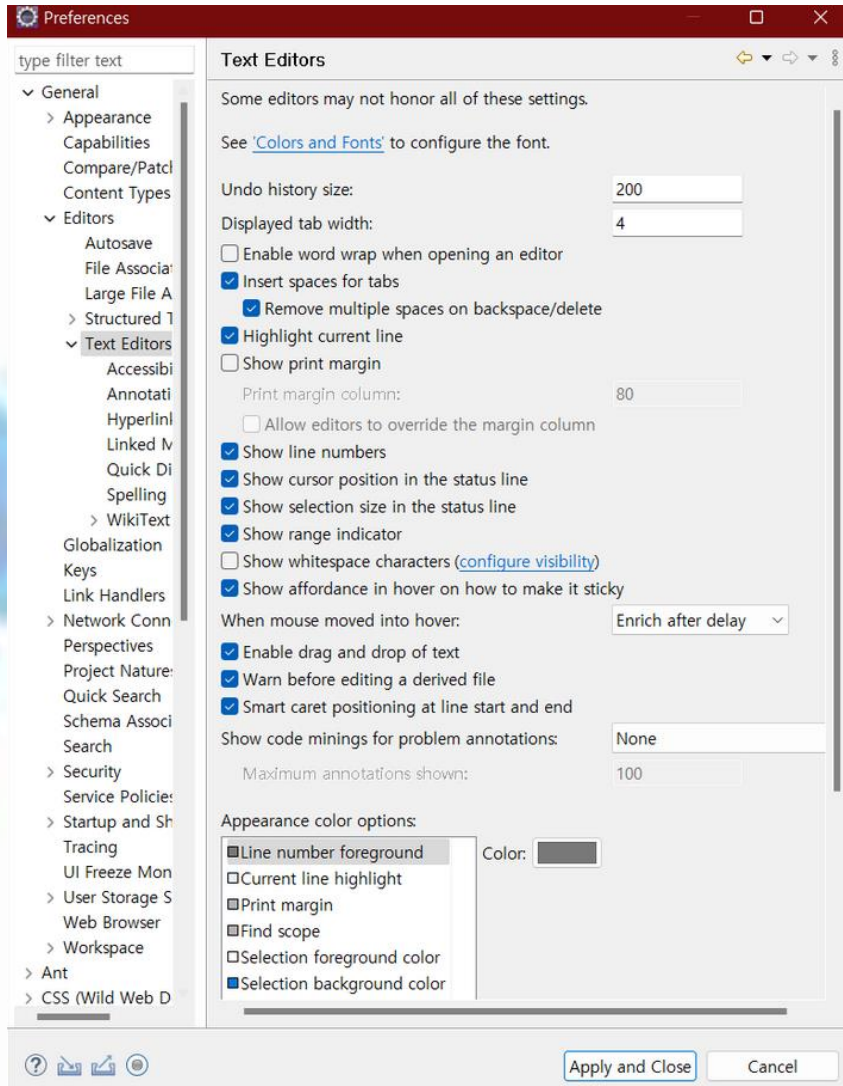




# TAB -> SPACE \*4 적용



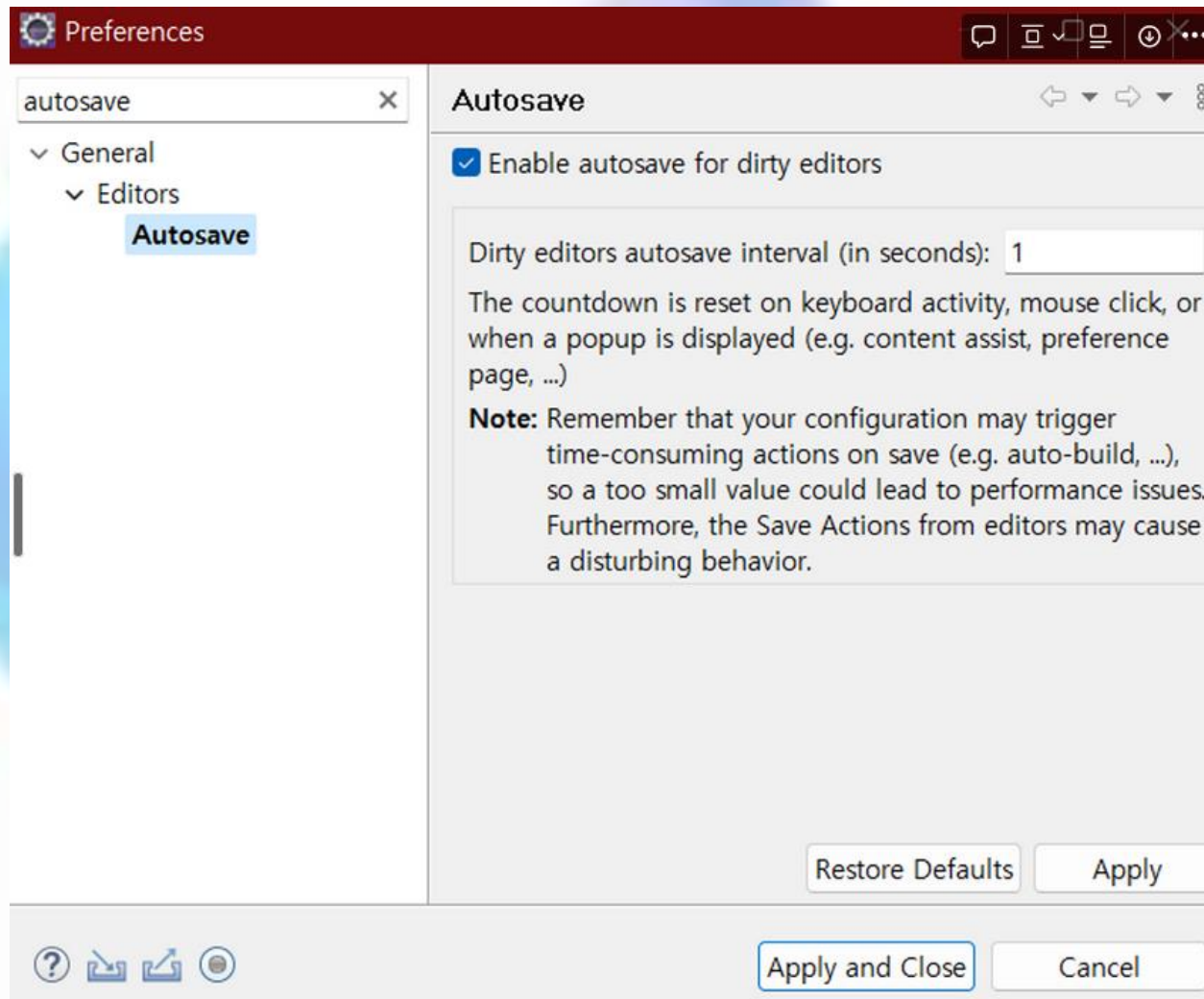
# TAB -> SPACE \*4 적용



☒ Remove multiple spaces on backspace/delete

설정시 탭을 한번에 지울수 있음

# AUTOSAVE



클래스 생성

프로젝트 생성

패키지 생성


com.도메인명

소문자로만 작성

클래스 생성




# 클래스 생성

 New Java Class

Java Class

Create a new Java class.



Source folder:

JavaExam/src

Browse...

Package:

com.example

Browse...

☐ Enclosing type:

Browse...

Name:

Main

Modifiers:

☒ public

☐ package

☐ private

☐ protected

☐ abstract

☐ final

☐ static

☒ none

☐ sealed

☐ non-sealed

☐ final

Superclass:

java.lang.Object

Browse...

Interfaces:

Add...

Remove

Which method stubs would you like to create?

☒ public static void main(String[] args)

☐ Constructors from superclass

☒ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))

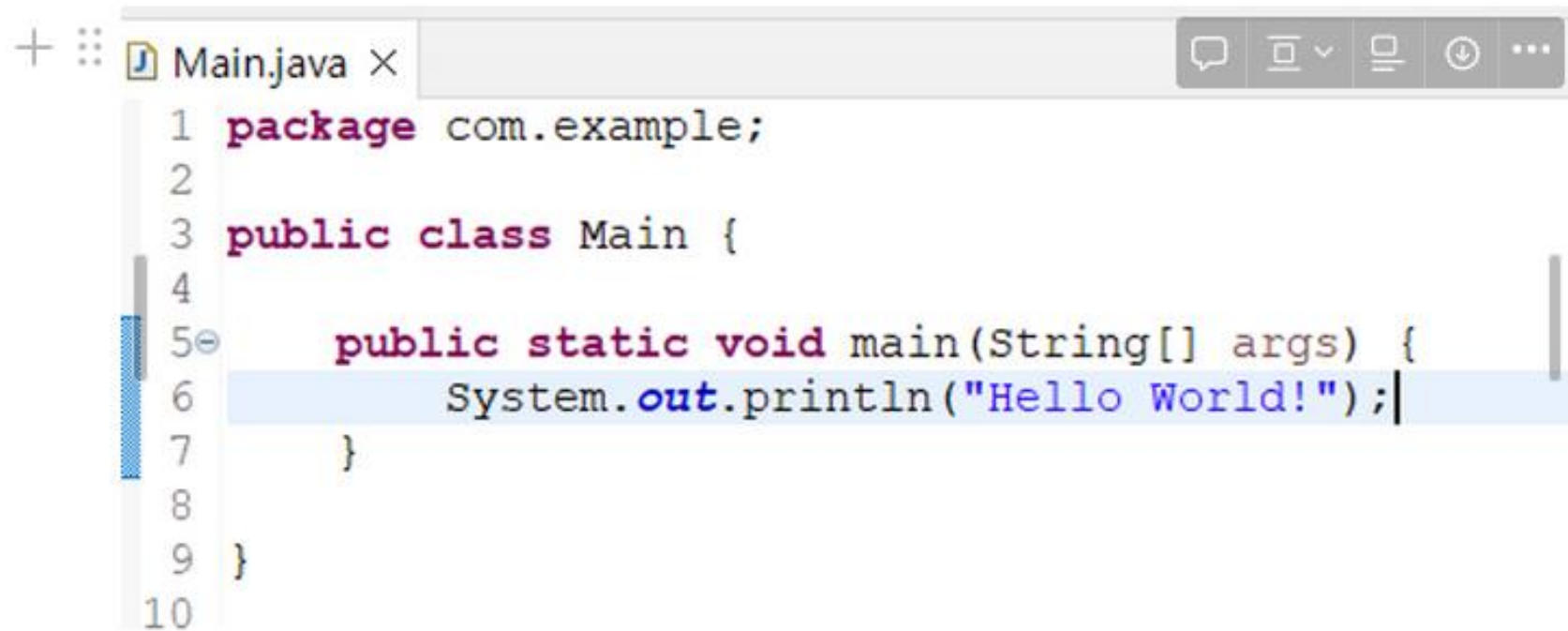
☐ Generate comments

?

Finish

Cancel

## 예제 출력



```
+ :: Main.java ×
1 package com.example;
2
3 public class Main {
4
5     public static void main(String[] args) {
6         System.out.println("Hello World!");
7     }
8
9 }
10
```

## 단축키

sysout + ctrl space  $\Rightarrow$  system.out.println()



A horizontal splash of water with a gradient of colors from blue on the left to pink on the right, set against a light beige background.

# 연습문제

## 1-1

**Java**로 프로그램을 개발하기 위해, ( 컴파일러 ) 와 ( 인터프리터 ) 라는 소프트웨어가 필요하다. ( 컴파일러 ) 는 **Java** 문법으로 작성한 ( 소스코드 ) 를 ( 바이트코드 ) 로 변환 해 줍니다. ( 인터프리터 ) 는 내부에 가지고 있는 ( **JVM** ) 을 사용하여 이것을 해석하고 기계어로 변환하여 **CPU** 가 실행합니다.

# 1-2

Java ▾

```
package com.example;

import java.util.*;

public class Main {
    public static void main(String[] args) {
        int a = 3;
        int b = 5;
        int c = a * b;

        System.out.println("가로 " + a + ", 세로 " + b + "의 직사각형의 면적은 " + c);
    }
}
```

Problems @ Javadoc Declaration Console ×  
<terminated> Main [Java Application] C:\Users\admin\W  
가로 3, 세로 5의 직사각형의 면적은 15

# 1-3

Java ▾

복사 캡션 ...

```
package com.example;

import java.util.*;

public class Main {
    public static void main(String[] args) {
        boolean a = true;
        String b = "남";
        double c = 3.14;
        Long d = 3141592658539791;
        String e = "항구를 공격! 적에게 15 포인트의 데미지를 주었다";
    }
}
```

# 2-1

Java ▾

복사 캡션 ...

```
package com.example;

import java.util.*;

public class Main {
    public static void main(String[] args) {
        int x = 5;
        int y = 10;
        int answer = x + y;
        System.out.println("x+y는 " + answer);
    }
}
```

Problems @ Javadoc Declaration Console ×

<terminated> Main [Java Application] C:\Users\admin\

x+y는 15

2-2

2, 4, 5, 7

1. 반환 결과가 정수형인 정수형과 실수형 데이터의 연산  $\Rightarrow$  하위 형에 상위 형 계산됨
2. 옳음 (묵시적 형 변환, `double`  $\rightarrow$  `float`)
3. 하위 형인 정수타입 선언 되었으나, 상위 형인 문자열 5가 대입
4. 옳음
5. 옳음
6. 논리 데이터는 `boolean`으로 선언함
7. 옳음 (명시적 데이터타입 선언)



## 2-3

Java ▾

복사 캡션 ...

```
package com.example;

import java.util.*;

public class Main {
    public static void main(String[] args) {
        System.out.println("점을 보세요");
        System.out.println("이름을 입력해 주세요");
        String name = new java.util.Scanner(System.in).nextLine();
        System.out.println("나이를 입력 해 주세요");
        String ageString = new java.util.Scanner(System.in).nextLine();
        int age = Integer.parseInt(ageString);
        int fortune = new java.util.Random().nextInt(4);
        fortune++;
        System.out.println("점괘가 나왔습니다!");
        System.out.println(age + "살의 " + name + "씨, 당신의 운세번호는 " + fortune + " 입니다");
        System.out.println("1:대박 2:중박 3:보통 4:망");
    }
}
```

Problems @ Javadoc Declaration Console ×

<terminated> Main [Java Application] C:\Users\wadmin\Des

점을 보세요

이름을 입력해 주세요

홍윤기

나이를 입력 해 주세요

27

점괘가 나왔습니다!

27살의 홍윤기씨, 당신의 운세번호는 4 입니다

1:대박 2:중박 3:보통 4:망