

## [마인크래프트 소개]



- 마르쿠스 페르손이 만든 후, Mojang AB사에 의해 개발, 판매되고 있는 샌드박스 게임 (현재는 MS에 인수)
- 모든 곳이 네모난 블록으로 만들어진 세계에서 몬스터들을 피해 집을 짓고, 도구를 만들고, 채광, 농사를 하여 생존하는 간단한 구성
- 건축물을 짓거나, 빌딩을 짓거나, 블록을 쌓아 그림을 그리거나, 게임 모드를 만드는 등 높은 자유도로 플레이어들이 콘텐츠를 스스로 창조할 수 있음, 또한 멀티플레이도 가능하며, 서버라는 하나의 공간 안에서 여러 플레이어가 게임을 즐길 수 있음.
- 플러그인: 마인크래프트 서버에서 원하는 기능(게임, 편의기능) 추가

## [텔레포트 기능 만들어 보기: 특정한 지점(0,64,5)로 이동]

마인크래프트의 Steve를 텔레포트 시키기 전에, 현실세계의 자신을 도봉아이나라도서관으로 이동시키는 명령부터 컴퓨터에게 내려 보자.

(브레인스토밍)

- 서울특별시로 자신을 텔레포트 시킬 때 컴퓨터에게 알려줘야 할 정보는 어떤 게 있을까요?
- 컴퓨터에게 내려야 할 명령을 구체화해 보자, 구체화라는 것은 아무것도 모르는 사람에게 무엇을 어떻게 해야하는지 세부적으로 설명하는 설명서와 같다.
- \*참고: 서울특별시의 위경도는 (37, 127) 이다.

- 위에서 답한 내용을 바탕으로 더 구체적으로 한 줄짜리 명령을 만들어보자.

단, 이번에는 '만약( ~인 경우 ) ☞ ~~~을 한다.' 식으로 문장을 만들 것.

ex) 만약(동생이 휴대폰으로 게임을 한다면) ☞ 엄마가 휴대폰을 뺏는다

만약( 서울특별시로 텔레포트하라는 명령이 들어오면 ), then(그 다음에는)

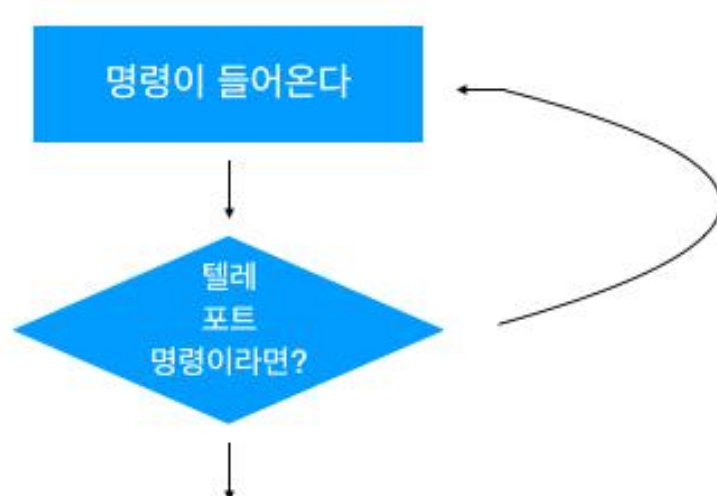


- 이번에는 내린 명령을 바탕으로 순서도를 만들어 보자. 왼쪽에 보이는 <사용할 수 있는 블록>을 이용하여 순서도를 완성하시오.

### 순서도를 만들어 보자



<사용할 수 있는 블록>

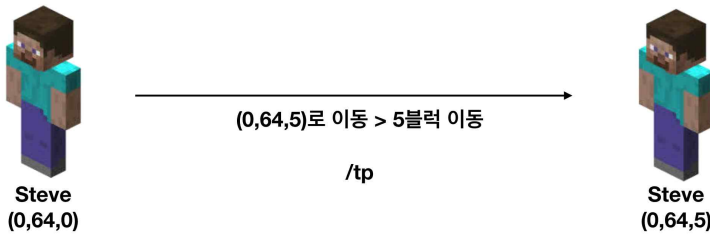


(                    )을 텔레포트

 행성: (                    )위치 (                    )(                    )

### [텔레포트-특정한 지점(0,64,5)로 이동]

- 아래 그림과 같이 특정한 지점(0,64,5) 좌표로 이동하는 텔레포트 순서도를 만들어 보자. 순서도 그림의 왼쪽에 보이는 <사용할 수 있는 블록>을 이용하여 순서도를 완성하시오.



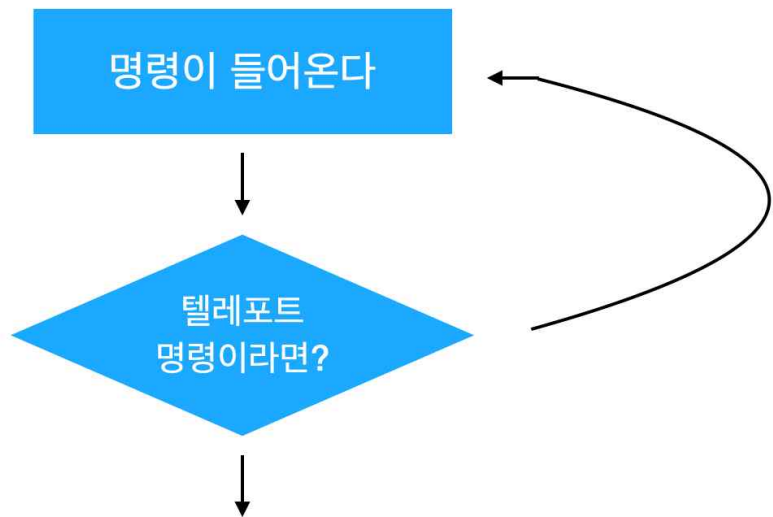
특정한 지점으로

마인크래프트의 스티브를 이동시켜 봅시다.

### 순서도를 만들어 보자



<사용할 수 있는 블록>



( )을 텔레포트 위치: ( ) ( ) ( ) ( )

### [텔레포트-특정한 지점(0,64,5)로 이동: 프로그래밍 해보기]

- \* 실제로 마인크래프트 플러그인을 만들어 보고 싶으신 분들만 보시기 바랍니다. 생략해도 괜찮습니다.
- \* 기본적인 Java 언어에 대한 이해가 있으면 Code를 이해하기에 좋습니다.

```

1: package javaplugin;
2: import org.bukkit.Location;
3: import org.bukkit.World;
4: import org.bukkit.command.Command;
5: import org.bukkit.command.CommandExecutor;
6: import org.bukkit.command.CommandSender;
7: import org.bukkit.entity.Player;
8: import org.bukkit.plugin.java.JavaPlugin;
9:
10: public class Main extends JavaPlugin implements CommandExecutor{
11:     @Override
12:     public void onEnable() {
13:         System.out.println("플러그인이 활성화되었습니다!");
14:         getCommand("tp").setExecutor(this); //tp 명령어를 쳤을 때, onCommand에서 받아들이도록 합니다.
15:     }
16:
17:     @Override
18:     public boolean onCommand(CommandSender sender, Command cmd, String string, String[] args) {
19:         if(cmd.getName().equalsIgnoreCase("tp")) { //만약, 입력받은 명령어가 tp인 경우
20:             Player player = (Player)sender; //플레이어의 정보 받기
21:             World world = player.getWorld(); //World의 정보 받기
22:             player.teleport(new Location(world, 0, 64, 5)); //World의 0,64,5 지점으로 이동
23:         }
24:         return true; //boolean 타입이므로, True/False를 return해야 함
25:     }
26: }
    
```

plugin.yml에 아래 항목을 추가한다.

```

name: Main
version: 1.0
main: javaplugin.Main
....
commands:
  tp:
    description: Teleport Command
    
```

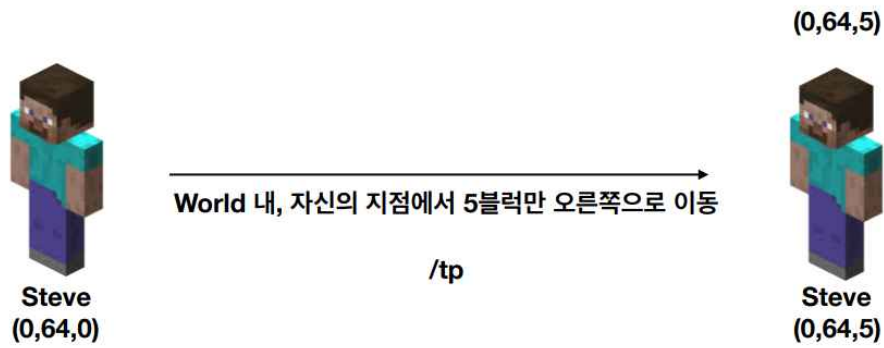
※ commands: 내 'tp'는 /tp 명령어를 의미하며, 바꾸고 싶다면 'teleport' 등으로 바꾸고, Code를 수정하면 됨.

14번: `getCommand("바꾸고 싶은 명령어").setExecutor(this);`

27번: `if(cmd.getName().equalsIgnoreCase("바꾸고 싶은 명령어"))`

※ description: 은 명령어의 설명입니다.

[텔레포트-플레이어의 지점에서 5칸 옆으로 이동하기]



5블럭만 오른쪽으로

마인크래프트의 스티브를 이동시켜 봅시다.

(브레인스토밍)

- 자신의 지점에서, 플레이어를 5블럭만 오른쪽으로 이동시키기 위해 컴퓨터에게 어떻게 구체적으로 명령을 내릴 수 있을까요?

- 위에서 답한 내용을 바탕으로 한 줄짜리 명령을 만들어보자.

단, 이번에는 '만약( ~인 경우 ) ➡ ~~~을 한다.' 식으로 문장을 만들 것.

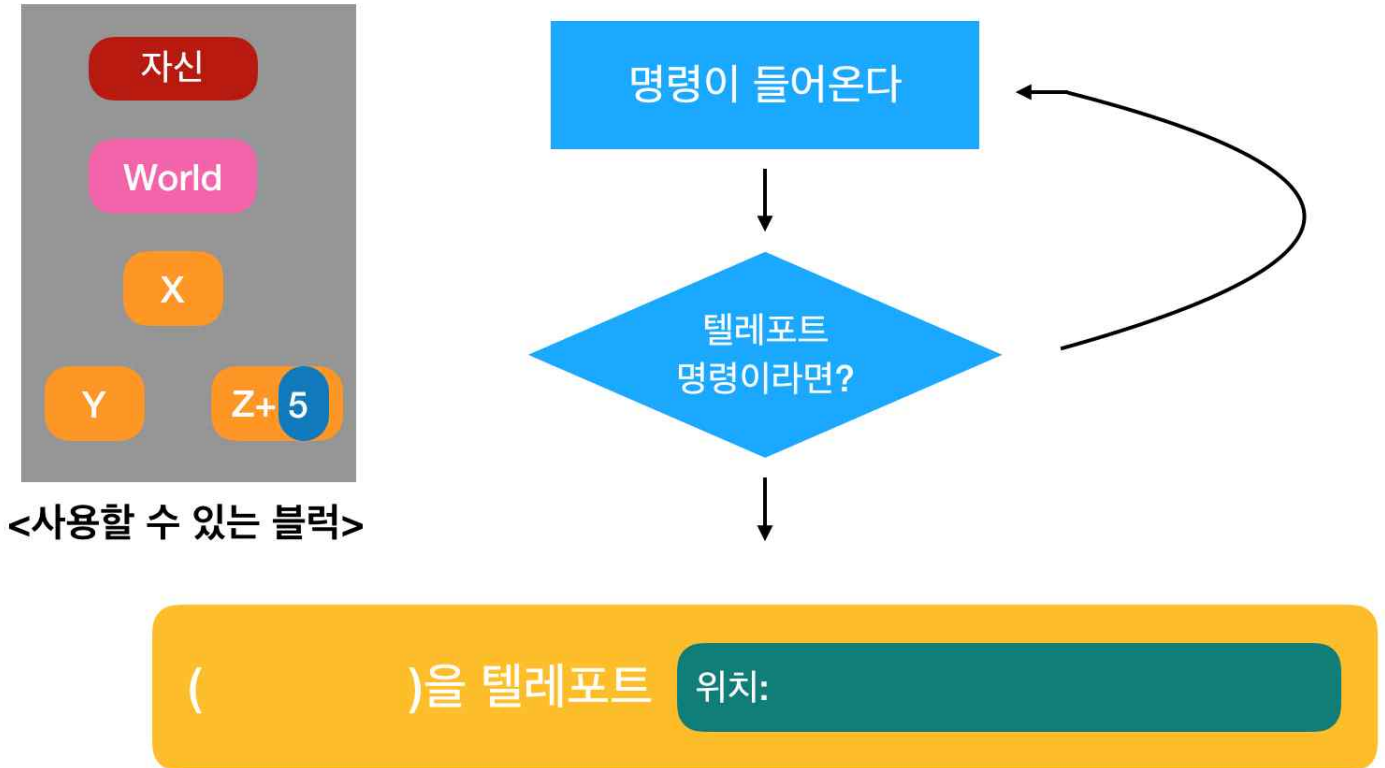
ex) 만약(동생이 휴대폰으로 게임을 한다면) ➡ 엄마가 휴대폰을 뺏는다

만약(  ), then(그 다음에는)



- 아래 그림과 같이 플레이어의 지점(X, Y, Z)에서 Z방향으로 5블럭을 이동하는 텔레포트 순서도를 만들어 보자. 순서도 그림의 왼쪽에 보이는 <사용할 수 있는 블럭>을 이용하여 순서도를 완성하시오.

### 순서도를 만들어 보자



[텔레포트-플레이어의 지점에서 5칸 옆으로 이동하기: 프로그래밍 해보기]

- \* 실제로 마인크래프트 플러그인을 만들어 보고 싶으신 분들만 보시기 바랍니다. 생략해도 괜찮습니다.
- \* 기본적인 Java 언어에 대한 이해가 있으면 Code를 이해하기에 좋습니다.

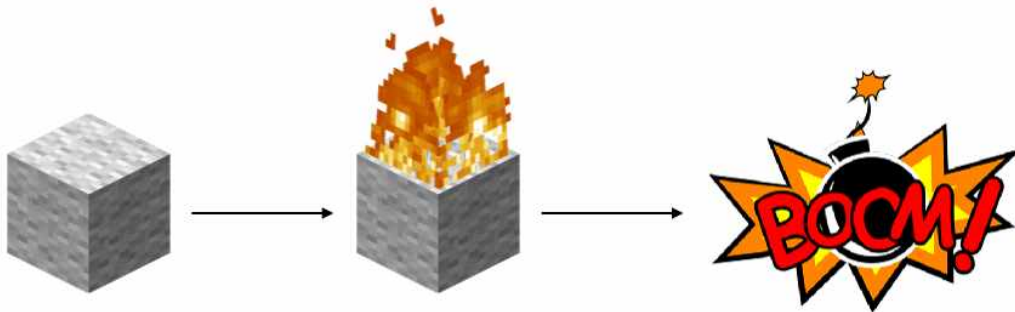
```

1: package javaplugin;
2:
3: import org.bukkit.Location;
4: import org.bukkit.World;
5: import org.bukkit.command.Command;
6: import org.bukkit.command.CommandExecutor;
7: import org.bukkit.command.CommandSender;
8: import org.bukkit.entity.Player;
9: import org.bukkit.plugin.java.JavaPlugin;
10:
11: public class Main extends JavaPlugin implements CommandExecutor{
12:     @Override
13:     public void onEnable() {
14:         System.out.println("플러그인이 활성화되었습니다!");
15:         //getCommand("tp").setExecutor(this);
16:     }
17:
18:     @Override
19:     public boolean onCommand(CommandSender sender, Command cmd, String string, String[] args) {
20:         if(cmd.getName().equalsIgnoreCase("tp")) { //만약, 입력받은 명령어가 tp인 경우
21:             Player player = (Player)sender; //플레이어의 정보 받기
22:             double x = player.getLocation().getX(); //플레이어의 위치를 알아야 하므로, x/y/z좌표 입력
23:             double y = player.getLocation().getY();
24:             double z = player.getLocation().getZ();
25:             World world = player.getWorld(); //World의 정보 받기
26:             player.teleport(new Location(world, x, y, z+5)); //플레이어의 x값(x), 플레이어의 y값(y), 플레
   이어의 z값+5(z+5)
27:         }
28:         return true; //boolean 타입이므로, True/False를 return해야 함
29:     }
30: }

```

[초강력 폭탄 만들기]

이번에는 플레이어가 양털(주황색)에 우측클릭을 하여 불이 붙은 경우,  
터지는 초강력 폭탄을 만들어 보도록 하겠습니다.



<https://youtu.be/xl2HVR3NsCU?t=2m32s>

초강력 폭탄을  
만들어 봅시다.

(브레인스토밍)

- 어떤 폭탄을 만들면 좋을지 생각해 봅시다.

\* 생각해 봐야 할 것

폭탄의 폭발 방법(점화): \_\_\_\_\_

ex) 위에 불을 붙이면 폭발한다.

폭탄으로 설정할 블록: \_\_\_\_\_

ex) 오렌지 양털 블록 (시연에서는 양털 블록만 할 예정임)

폭탄 폭발규모: \_\_\_\_ F 단위 (ex. 크리퍼 4F/TNT 6F)

(명령 구체화하기)

- 한 줄짜리 명령을 만들어보자.

단, 이번에는 '만약( ~인 경우 ) ➡ ~~~을 한다.' 식으로 문장을 만들 것.

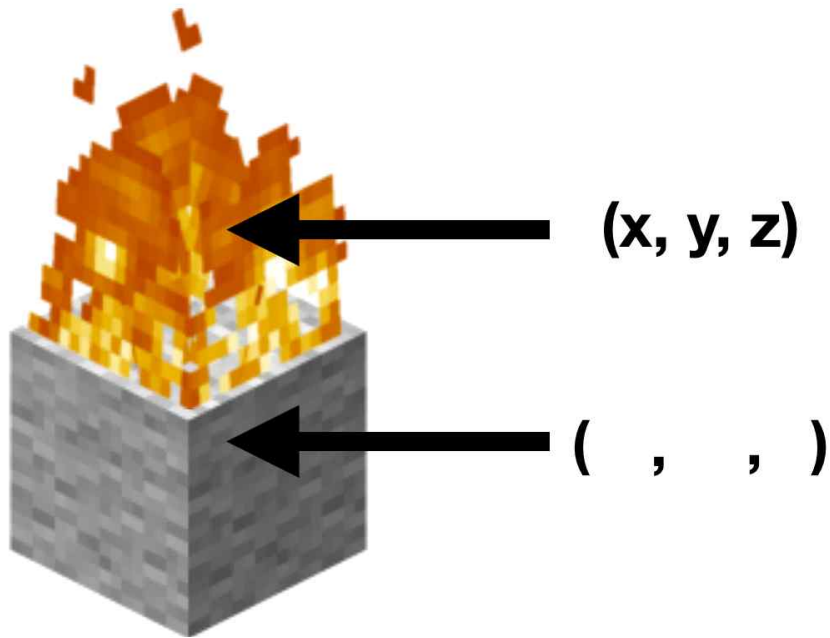
ex) 만약(동생이 휴대폰으로 게임을 한다면) ➡ 엄마가 휴대폰을 뺏는다

만약(  ), then(그 다음에는)





- 불 밑에 설치된 블록이 폭탄인지 알기 위해 폭탄의 좌표를 알고자 한다. 빈 칸을 채우시오.



- 불이 설치되었는지, 불 밑에 있는 블록이 폭탄인지 구별하는 조건을 넣어 최종 명령을 완성하시오.

만약(  ) then,



만약(  ) then,

### [초강력 폭탄 만들기: 프로그래밍 해보기]

- \* 실제로 마인크래프트 플러그인을 만들어 보고 싶으신 분들만 보시기 바랍니다. 생략해도 괜찮습니다.
- \* 기본적인 Java 언어에 대한 이해가 있으면 Code를 이해하기에 좋습니다.

```

1: package javaplugin;
2:
3: import org.bukkit.*;
4: import org.bukkit.block.Block;
5: import org.bukkit.entity.Player;
6: import org.bukkit.event.EventHandler;
7: import org.bukkit.event.Listener;
8: import org.bukkit.event.block.BlockPlaceEvent;
9: import org.bukkit.material.Wool;
10: import org.bukkit.plugin.java.JavaPlugin;
11:
12: public class Main extends JavaPlugin implements Listener {
13:     @Override
14:     public void onEnable() {
15:         getServer().getPluginManager().registerEvents(this, this); //Event 실행시 꼭 필요함
16:         System.out.println("플러그인이 활성화되었습니다!");
17:     }
18:
19:     @EventHandler
20:     public void onBlockPlace(BlockPlaceEvent e)
21:     {
22:         System.out.println("Block Placed");
23:         if (e.getBlockPlaced().getType().equals(Material.FIRE)) //만약 우클릭하여 설치한 아이템이 불인 경우
24:         {
25:             System.out.println("FIRE INSTALL");
26:             Player player = e.getPlayer(); //플레이어의 정보를 불러온다
27:             World world = player.getWorld(); //World의 정보를 불러온다
28:             double x1 = e.getBlockPlaced().getLocation().getBlockX(); //블럭의 x,y,z 좌표를 읽는다.
29:             double y1 = e.getBlockPlaced().getLocation().getBlockY();
30:             double z1 = e.getBlockPlaced().getLocation().getBlockZ();
31:
32:             Block b1 = new Location(world, x1, y1-1, z1).getBlock(); //(x,y-1,z)좌표: 폭탄블럭을 의미함
33:             if(b1.getType() == Material.WOOL) //만약 블럭이 WOOL인 경우 (폭탄 블럭)
34:             {
35:                 System.out.println("wool boom");
36:                 Wool wool = new Wool(b1.getType(), b1.getData());
37:                 DyeColor color = wool.getColor(); //양털의 색깔을 불러옴
38:                 if(color.equals(DyeColor.ORANGE)) //양털의 색깔이 주황색인 경우 (주황색 양털 = 폭탄)
39:                 {
40:                     System.out.println("BOOM!");
41:                     player.getWorld().createExplosion(b1.getLocation(), 100F); //폭발 발생 (크리퍼 4F)
42:                     player.getWorld().playEffect(b1.getLocation(), Effect.MOBSPAWNER_FLAMES, 10); //BOOM! 효과음 발생
43:                 }
44:             }
45:         }
46:     }
47: }

```

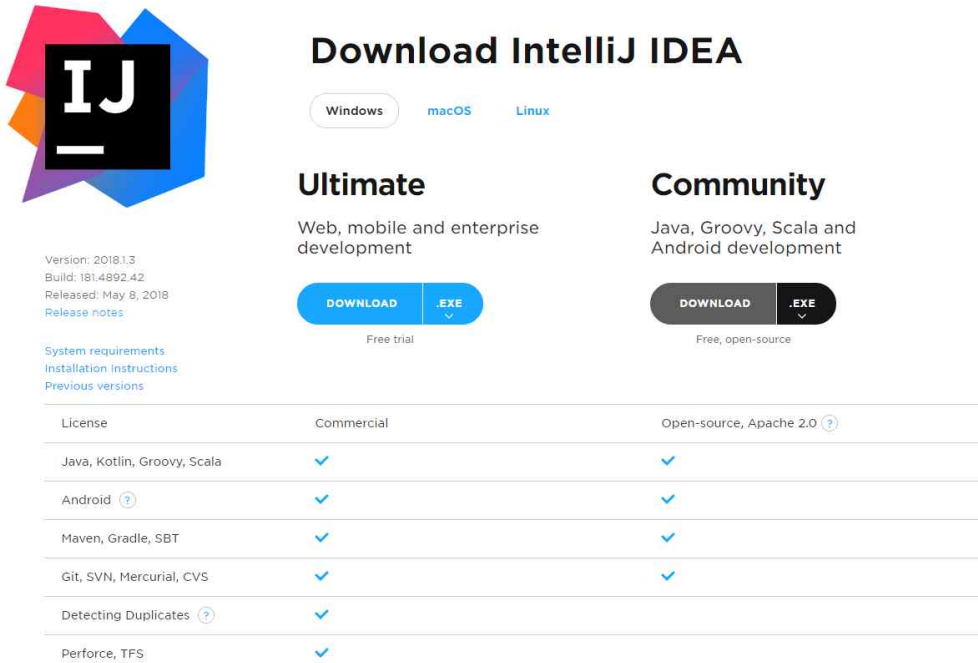
## [부록] IntelliJ 다운로드 방법

\* 가이드는 Windows OS 기준으로 작성하였습니다.

마인크래프트 플러그인을 만들기 위해 Eclipse도 쓰지만, Java 호환이 잘 되는 IntelliJ도 많이 사용한다.

1. <https://www.jetbrains.com/idea/> 에 접속하여, Download를 누른다.

2. Community를 다운로드한다.



**Download IntelliJ IDEA**

Windows macOS Linux

**Ultimate**  
Web, mobile and enterprise development

**Community**  
Java, Groovy, Scala and Android development

Version: 2018.1.3  
Build: 181.4892.42  
Released: May 8, 2018  
[Release notes](#)

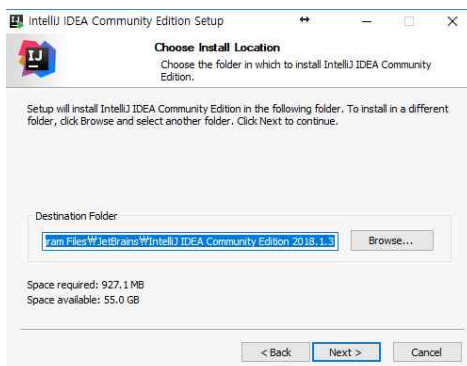
[System requirements](#)  
[Installation instructions](#)  
[Previous versions](#)

License	Commercial	Open-source, Apache 2.0 ?
Java, Kotlin, Groovy, Scala	✓	✓
Android ?	✓	✓
Maven, Gradle, SBT	✓	✓
Git, SVN, Mercurial, CVS	✓	✓
Detecting Duplicates ?	✓	
Perforce, TFS	✓	

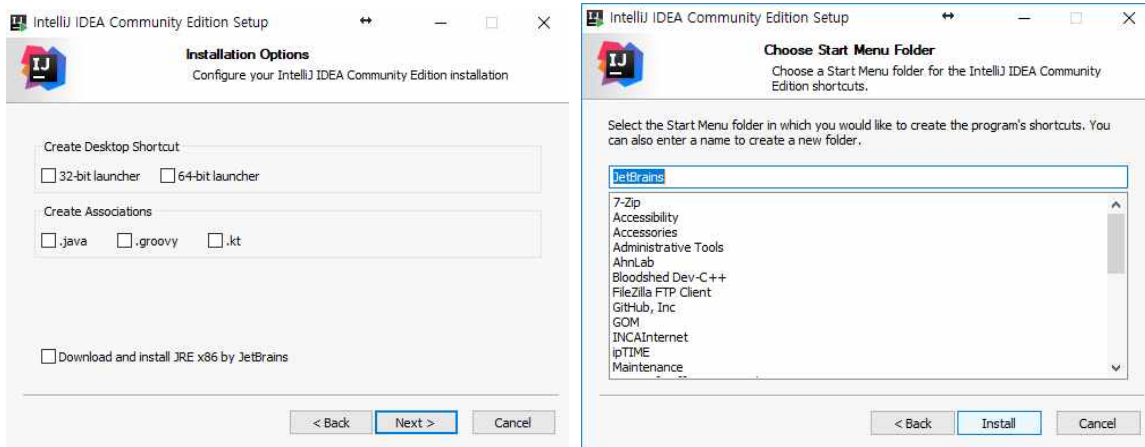
3. Next를 클릭한다.



4. Next를 클릭한다.



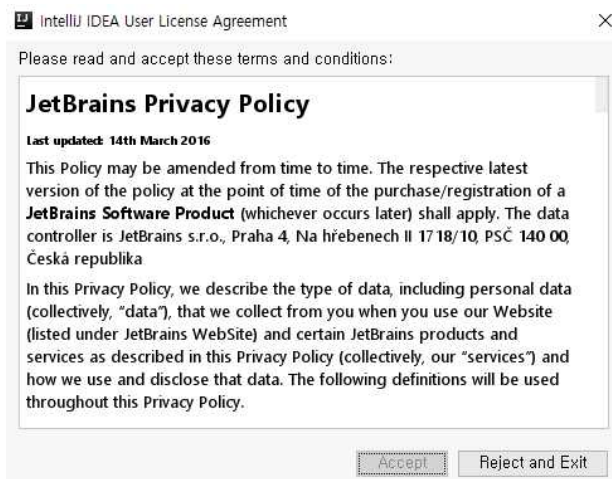
5. Next, Install을 클릭한다.



6. 설치가 완료되었다.



7. 글을 끝까지 읽고 Accept을 누른다.

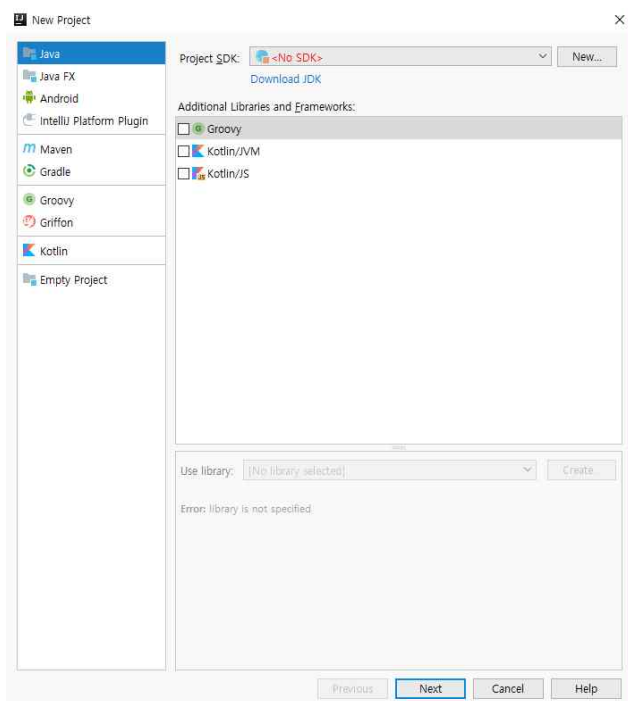


## [부록] IntelliJ로 마인크래프트 플러그인 만들기

8. IntelliJ로 마인크래프트 플러그인을 만들기 위해 Create New Project를 클릭한다.



9. Project SDK 밑에 있는 Download SDK를 클릭한다.



(10~12번은 JDK(Java Development Kit) 가 설치되어 있는 사람에 한해 생략 가능합니다.)

10. Java SDK Download를 클릭하여, 운영체제에 맞는 파일을 다운로드한다.

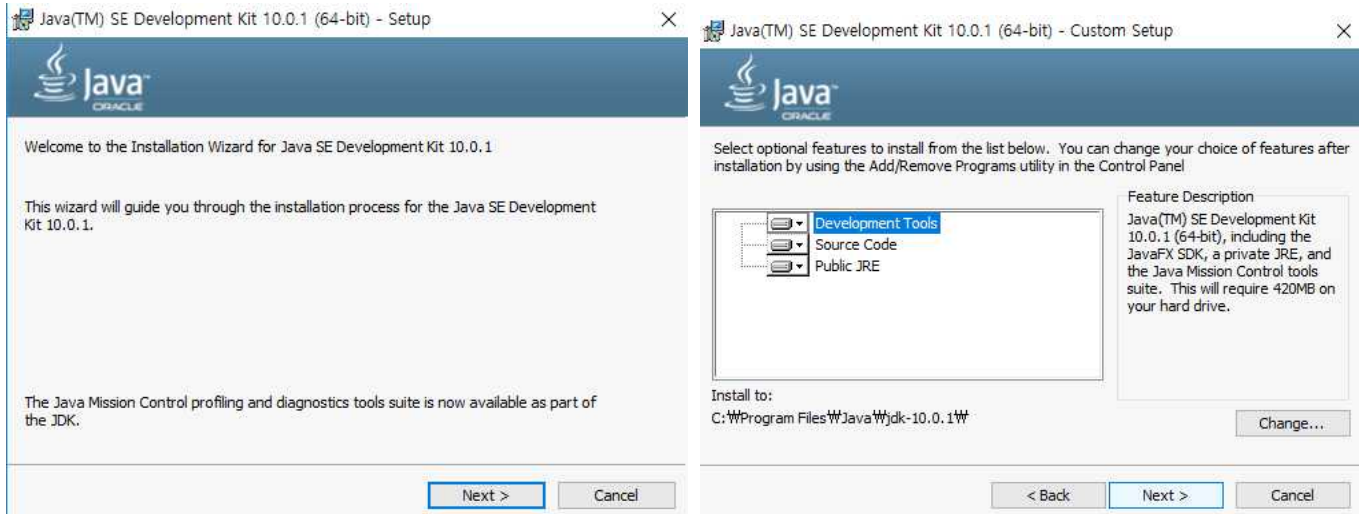
(<https://bit.ly/2qNMjb1> 로 접속하여, continue at your own risk to... 밑에 있는 링크를 다운로드해도 된다.)

## Java SE Downloads



본 활동지(Worksheet)의 무단 전재와 복사, 배포, 상업적인 용도의 이용을 금지합니다.

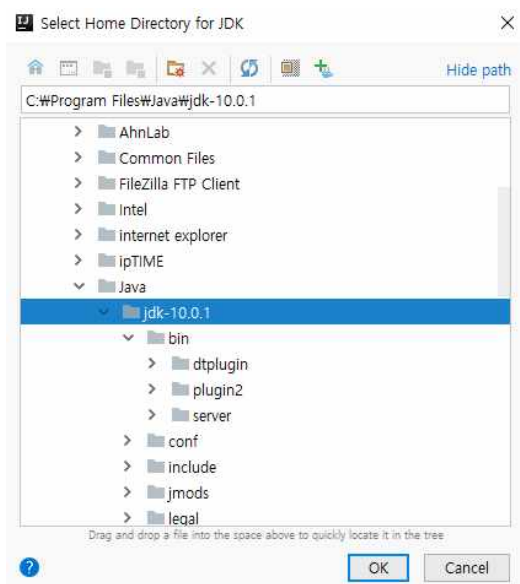
11. Next를 두 번 클릭한다. 설치가 자동으로 진행된다.



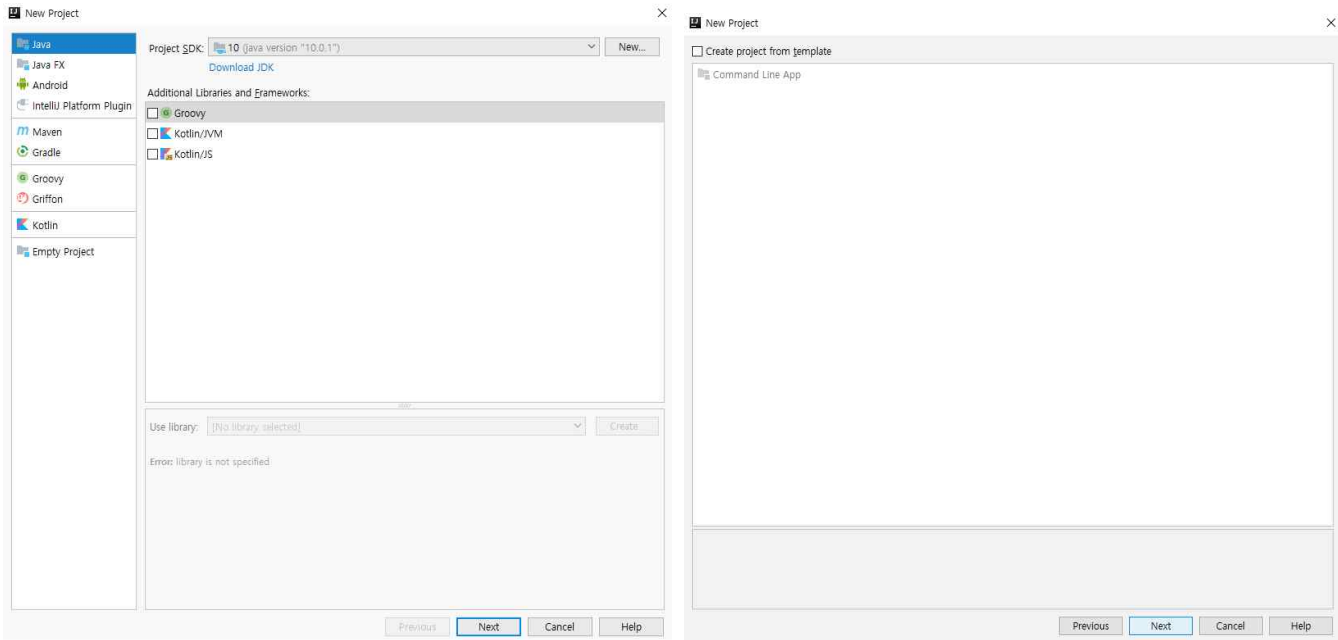
12. 다음을 클릭한다.



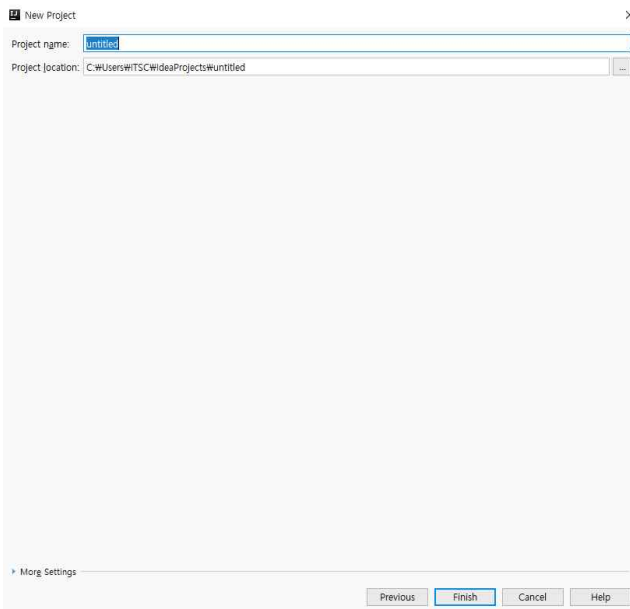
13. Project SDK 옆에 있는 New를 클릭하여, JDK의 위치를 지정한다. 폴더는 [jdk-10.x.x]를 선택한다.



14. 두 번 Next를 클릭한다.

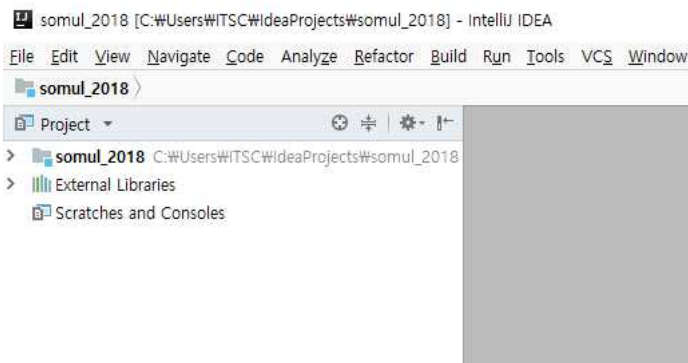


15. 프로젝트 이름을 지정한다. (ex. JavaPlugin)



(굳이 untitled라고 하겠다는 사람은 말리지 않겠다.)

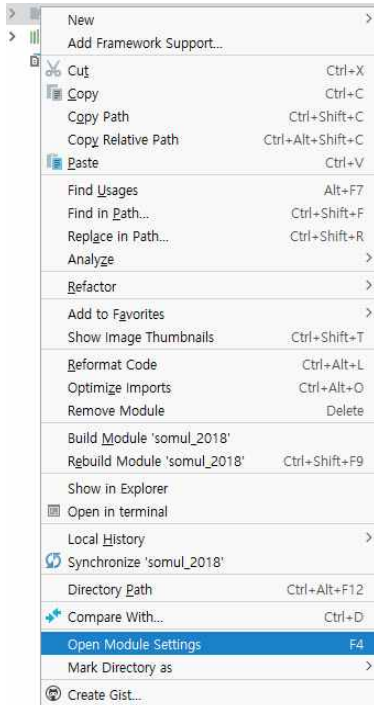
16. File 밑 프로젝트 명을 두 번 눌러 왼쪽 날개를 편다.



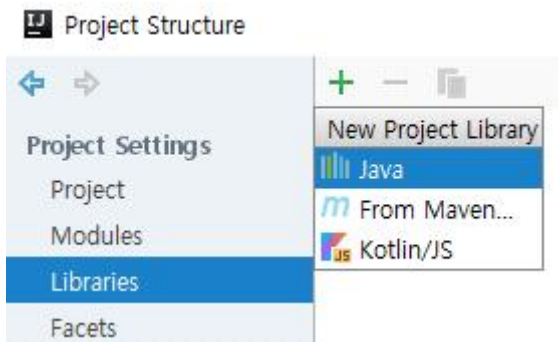
본 활동지(Worksheet)의 무단 전재와 복사, 배포, 상업적인 용도의 이용을 금지합니다.



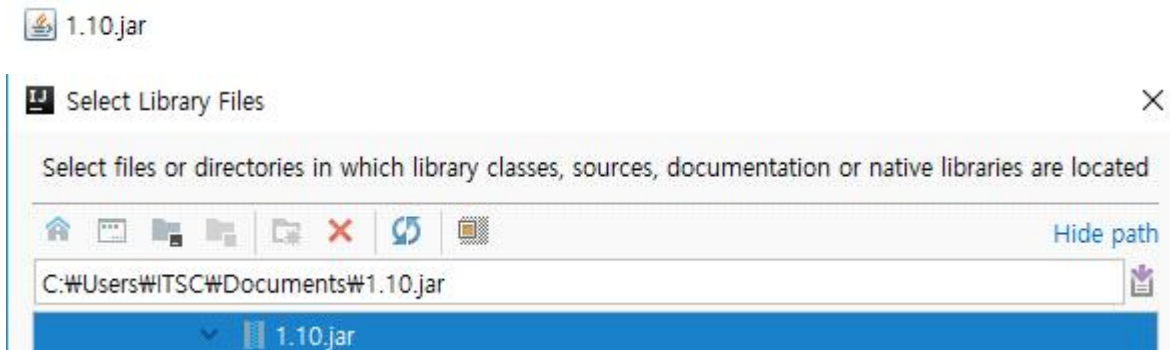
17. Project 밑 [프로젝트 명]에 우측클릭을 해 (File 밑 프로젝트 명 아님!) Open Moudle Setting을 클릭한다.



18. Project Structure → Libraries → + →  를 클릭한다.

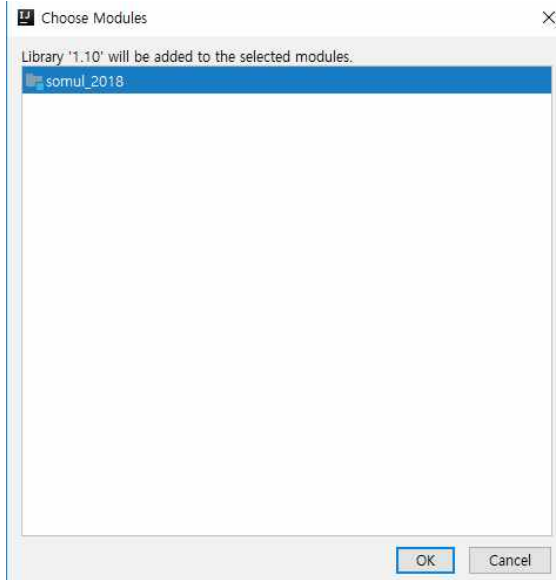


19. 다운받은 1.10.jar 버킷 파일 경로를 지정한다.

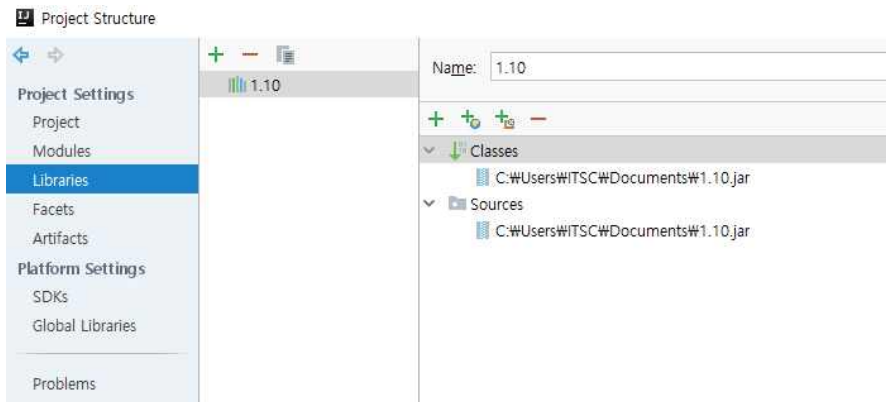




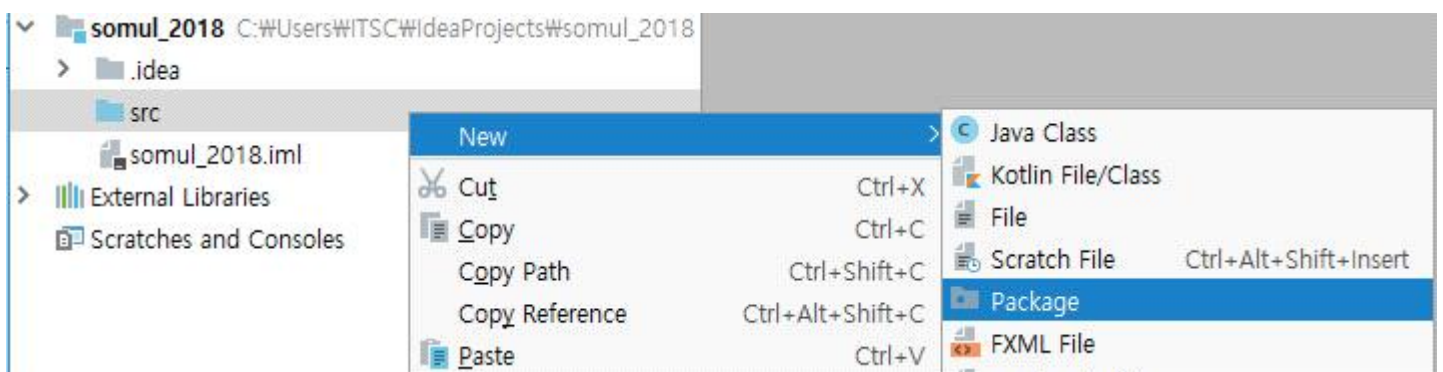
20. OK를 누른다.



21. 이렇게 설정이 된 경우 성공적으로 버킷 파일을 불러온 것이다.



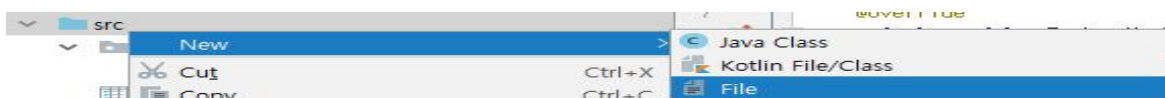
22. 프로젝트명 – **src**를 우클릭하여, [New] - [Package]를 만든다.



23. **패키지명**을 우클릭하여, [New] - [Java Class]를 만든다. Class명은 **[Main]** 으로 하도록 한다.



24. **src**를 우클릭하여, [New] - [File]을 만든다. 파일명은 **[plugin.yml]**(전부 소문자) 로 만든다.



본 활동지(Worksheet)의 무단 전재와 복사, 배포, 상업적인 용도의 이용을 금지합니다.

25.

①. Main.java의 Code를 아래와 같이 작성한다.

```
package javaplugin;

import org.bukkit.command.CommandExecutor;
import org.bukkit.plugin.java.JavaPlugin;

public class Main extends JavaPlugin {
    @Override
    public void onEnable() {
        System.out.println("플러그인이 활성화되었습니다!");
    }
}
```

②. plugin.yml은 아래와 같이 작성한다. (한 글자라도 틀리면 안됨!)

```
name: Main
version: 1.0
main: javaplugin.Main
```

name에는 클래스명 즉 Main을 쓰고, main에는 패키지명.클래스명을 쓴다.

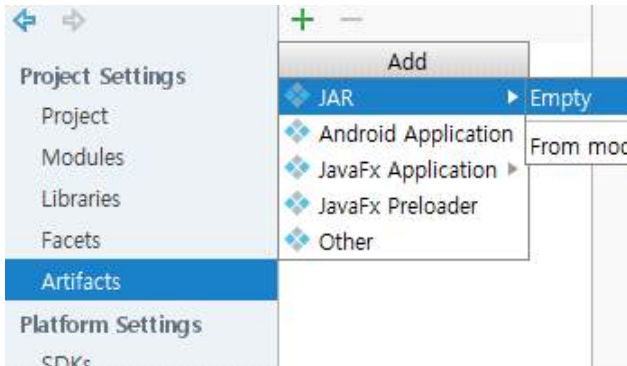
예를 들어, 패키지명이 **somul**이고 클래스명이 **Main**인 경우, main: 에는 **somul.Main**을 쓰면 된다.

이를 정확히 작성하지 않거나, 디렉토리가 일치하지 않을 경우 마인크래프트 플러그인 실행 시 에러가 표출된다.  
아래 에러는 디렉토리가 일치하지 않아 발생한 에러이다. (src에 바로 plugin.yml 있어야 함!)

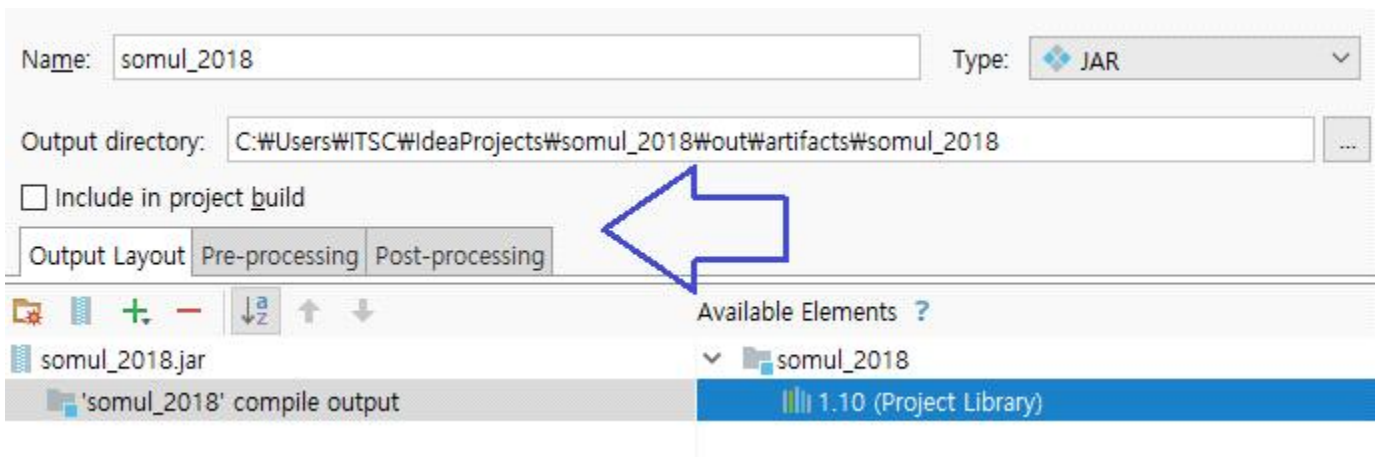
```
[16:33:59 ERROR]: Could not load 'plugins\somul_2018.jar' in folder 'plugins'
org.bukkit.plugin.InvalidDescriptionException: Invalid plugin.yml
    at org.bukkit.plugin.java.JavaPluginLoader.getPluginDescription(JavaPluginLoader.java:154) ~[1.10.jar:git-Spigot-078e2f8-7f9fbe5]
    at org.bukkit.plugin.SimplePluginManager.loadPlugins(SimplePluginManager.java:133) [1.10.jar:git-Spigot-078e2f8-7f9fbe5]
    at org.bukkit.craftbukkit.v1_10_R1.CraftServer.loadPlugins(CraftServer.java:297) [1.10.jar:git-Spigot-078e2f8-7f9fbe5]
    at org.bukkit.craftbukkit.v1_10_R1.CraftServer.reload(CraftServer.java:744) [1.10.jar:git-Spigot-078e2f8-7f9fbe5]
    at org.bukkit.Bukkit.reload(Bukkit.java:539) [1.10.jar:git-Spigot-078e2f8-7f9fbe5]
    at org.bukkit.command.defaults.ReloadCommand.execute(ReloadCommand.java:25) [1.10.jar:git-Spigot-078e2f8-7f9fbe5]
    at org.bukkit.command.SimpleCommandMap.dispatch(SimpleCommandMap.java:141) [1.10.jar:git-Spigot-078e2f8-7f9fbe5]
    at org.bukkit.craftbukkit.v1_10_R1.CraftServer.dispatchCommand(CraftServer.java:646) [1.10.jar:git-Spigot-078e2f8-7f9fbe5]
    at org.bukkit.craftbukkit.v1_10_R1.CraftServer.dispatchServerCommand(CraftServer.java:632) [1.10.jar:git-Spigot-078e2f8-7f9fbe5]
    at net.minecraft.server.v1_10_R1.DedicatedServer.aL(DedicatedServer.java:438) [1.10.jar:git-Spigot-078e2f8-7f9fbe5]
    at net.minecraft.server.v1_10_R1.DedicatedServer.D(DedicatedServer.java:401) [1.10.jar:git-Spigot-078e2f8-7f9fbe5]
    at net.minecraft.server.v1_10_R1.MinecraftServer.C(MinecraftServer.java:668) [1.10.jar:git-Spigot-078e2f8-7f9fbe5]
    at net.minecraft.server.v1_10_R1.MinecraftServer.run(MinecraftServer.java:567) [1.10.jar:git-Spigot-078e2f8-7f9fbe5]
    at java.base/java.lang.Thread.run(Unknown Source) [?:?]
Caused by: java.io.FileNotFoundException: Jar does not contain plugin.yml
... 14 more
```

26. 프로젝트명을 우측클릭하여, 아가와 같이 Open Moudle Setting을 누른다.

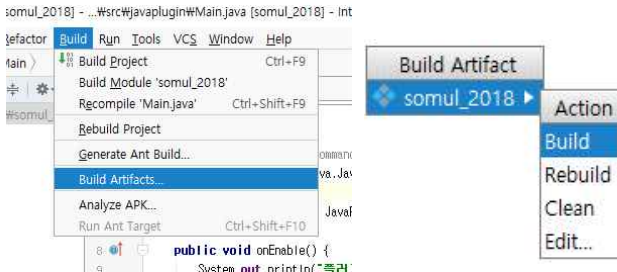
[Artifacts] - [JAR] - [Empty]를 누른다.



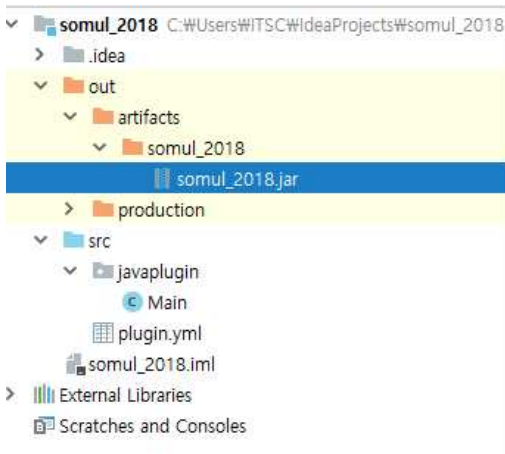
27. JAR 파일의 이름을 만들고, '프로젝트명' compile output을 클릭하여, 왼쪽으로 옮긴다. OK를 눌러 빠져나간다.



28. [Build] - [Build Artifacts]를 클릭한다. 이후 Build를 클릭한다.



29. JAR 파일이 out 디렉토리 내부에 생성된 것을 볼 수 있다. 이제 이 파일을 마인크래프트 서버에 넣으면 끝난다.



제공해드린 1.10.jar 파일을 구동하여 (cmd 창에서, java -jar 1.10.jar) 플러그인을 테스트해보기 바랍니다.

**본 활동지(Worksheet)의 무단 전재와 복사, 배포, 상업적인 용도의 이용을 금지합니다.**

30. 아래와 같은 메시지가 표출되고, 에러가 발생하지 않았다면 성공이다.

```
[16:41:05 INFO]: Max TNT Explosions: 100
[16:41:05 INFO]: Random Lighting Updates: false
[16:41:05 INFO]: Custom Map Seeds: Village: 10387312 Feature: 14357617
[16:41:05 INFO]: Structure Info Saving: true
[16:41:05 INFO]: Set PluginClassLoader as parallel capable
[16:41:05 INFO]: [Main] Loading Main v1.0
[16:41:05 INFO]: [Main] Enabling Main v1.0
[16:41:05 INFO]: 플러그인이 활성화되었습니다!
[16:41:05 INFO]: Server permissions file permissions.yml is empty, ignoring it
[16:41:05 INFO]: CONSOLE: Reload complete.
```

즐겁게 진행할 실습을 따라 하시기 바랍니다!