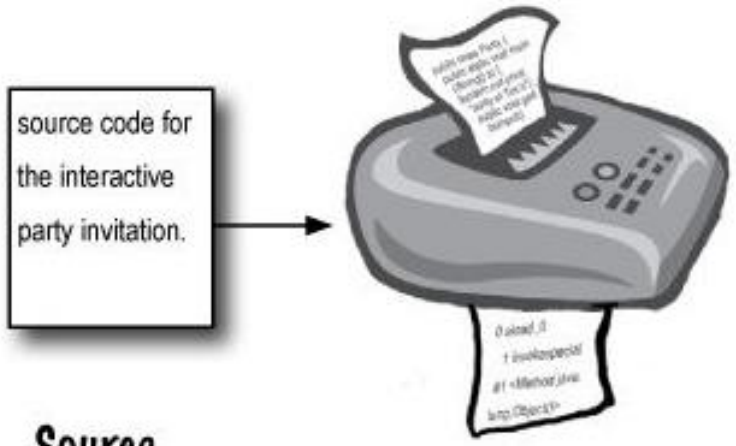
Dive in A Quick Dip: Breaking the Surface

Samkeun Kim <skim@hknu.ac.kr>

http://cyber.hankyong.ac.kr

국립한경대학교
HANKYONG NATIONAL UNIV.

The Way Java Works



Source

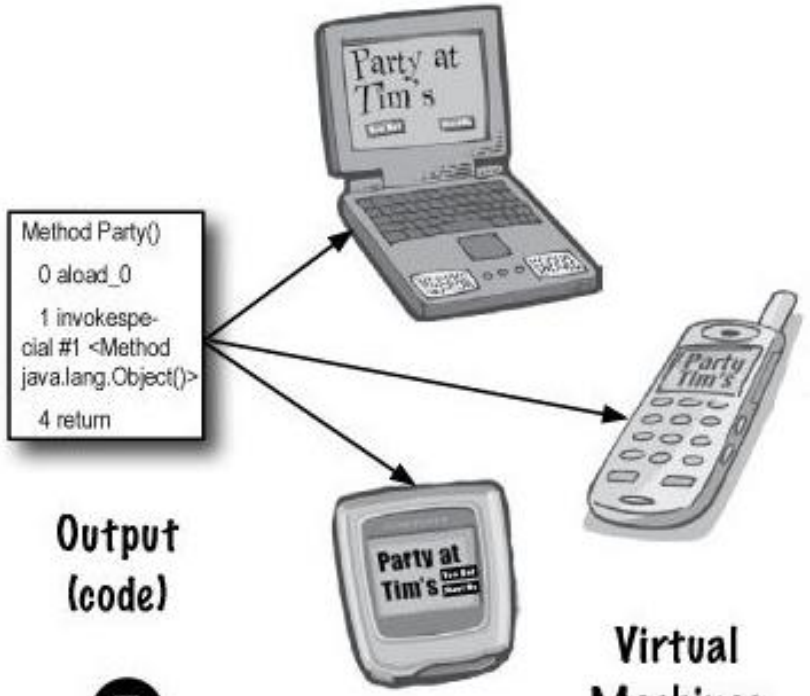
1

Create a source document. Use an established protocol (in this case, the Java language).

Compiler

2

Run your document through a source code compiler. The compiler checks for errors and won't let you compile until it's satisfied that everything will run correctly.



Output (code)

3

The compiler creates a new document, coded into Java ***bytecode***. Any device capable of running Java will be able to interpret/translate this file into something it can run. The compiled bytecode is platform-independent.

Virtual Machines

4

Your friends don't have a physical Java Machine, but they all have a ***virtual*** Java machine (implemented in software) running inside their electronic gadgets. The virtual machine reads and *runs* the bytecode.

What you'll do in Java

Source

1

Type your source code.

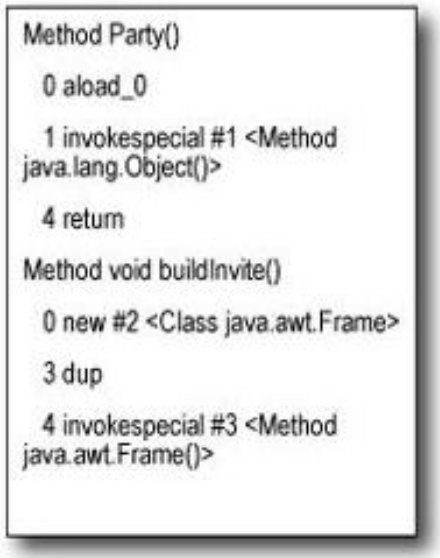
Save as: ***Party.java***

Compiler

2

Compile the ***Party.java*** file by running `javac` (the compiler application). If you don't have errors, you'll get a second document named ***Party.class***

The compiler-generated Party.class file is made up of *bytecodes*.



Output (code)

3

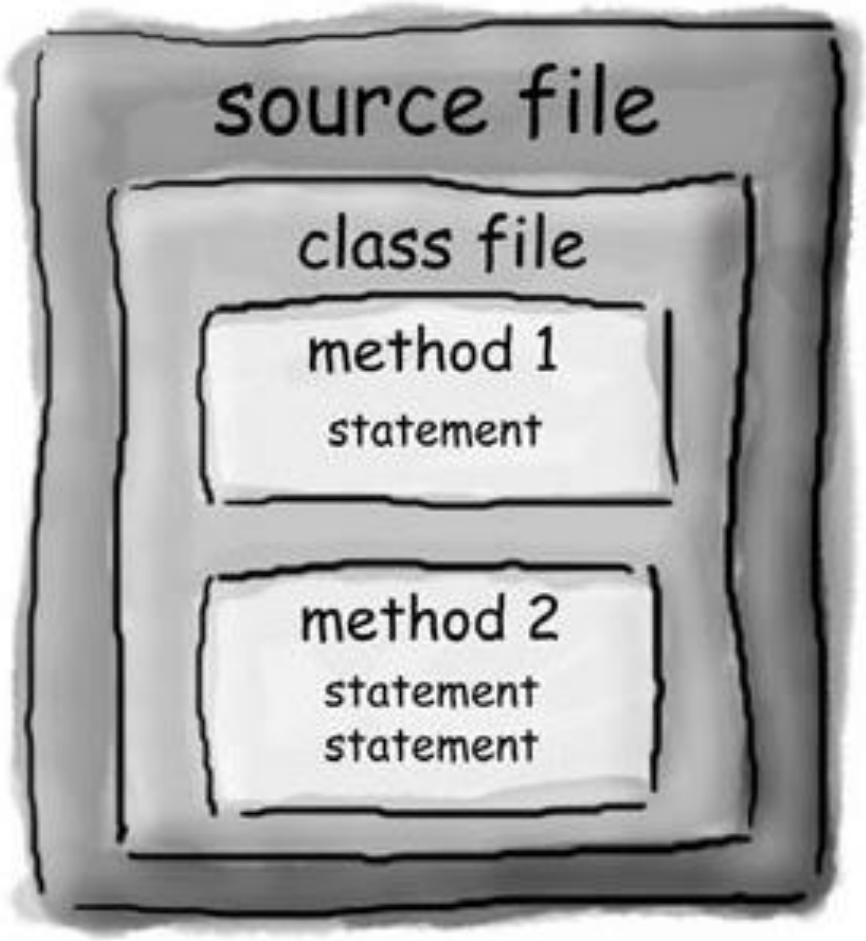
Compiled code: ***Party.class***

Virtual Machines

4

Run the program by starting the Java Virtual Machine (JVM) with the ***Party.class*** file. The JVM translates the *bytecode* into something the underlying platform understands, and runs your program.

Code structure in Java



Put a class in a source file.

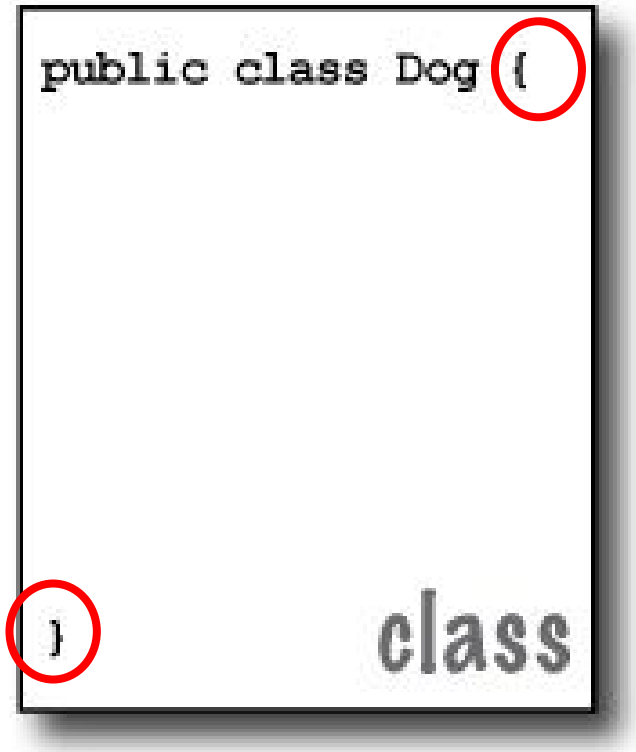
Put methods in a class.

Put statements in a method.

What goes in a source file?

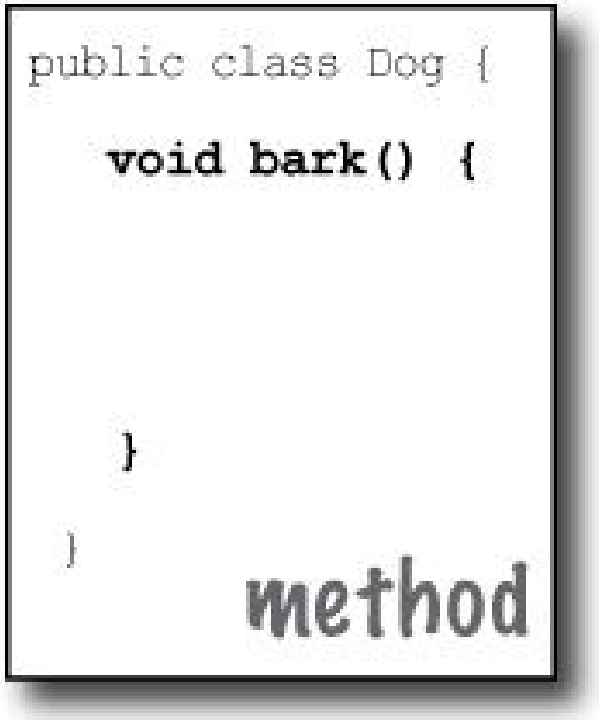
What goes in a source file?

- 소스 코드 파일 (확장자 **.java**) => 클래스 정의를 저장
- 클래스 => 프로그램의 일부분 (단 하나의 클래스만 가지는 프로그램 가능)
- The class must go within a pair of curly braces.



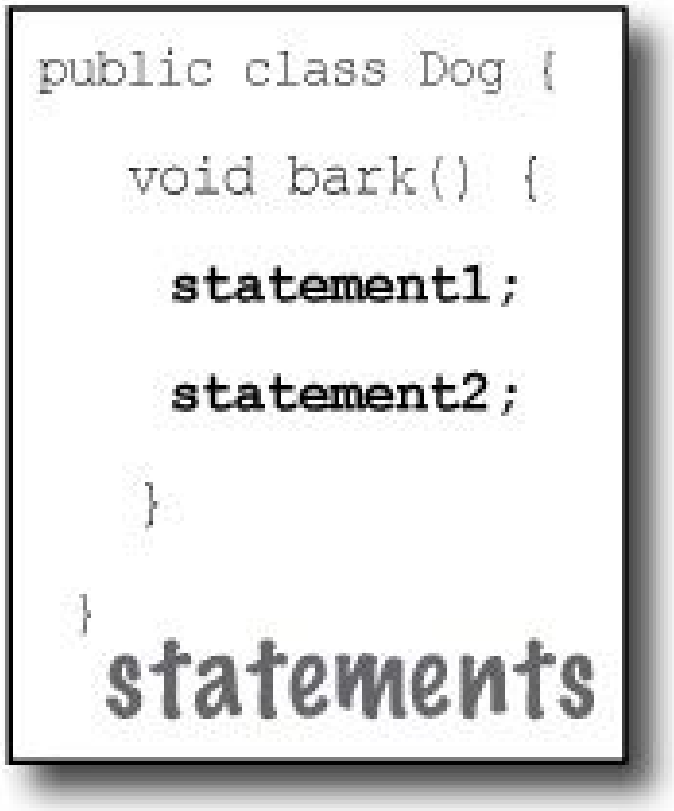
What goes in a class?

- 클래스 => 하나 이상의 메소드를 가진다
- **Dog** 클래스에서 **bark()** 메소드 => **Dog**가 어떻게 짖는가에 대한 명령문을 포함하게 된다
- 모든 메소드는 클래스 내부에 선언되어져야 한다



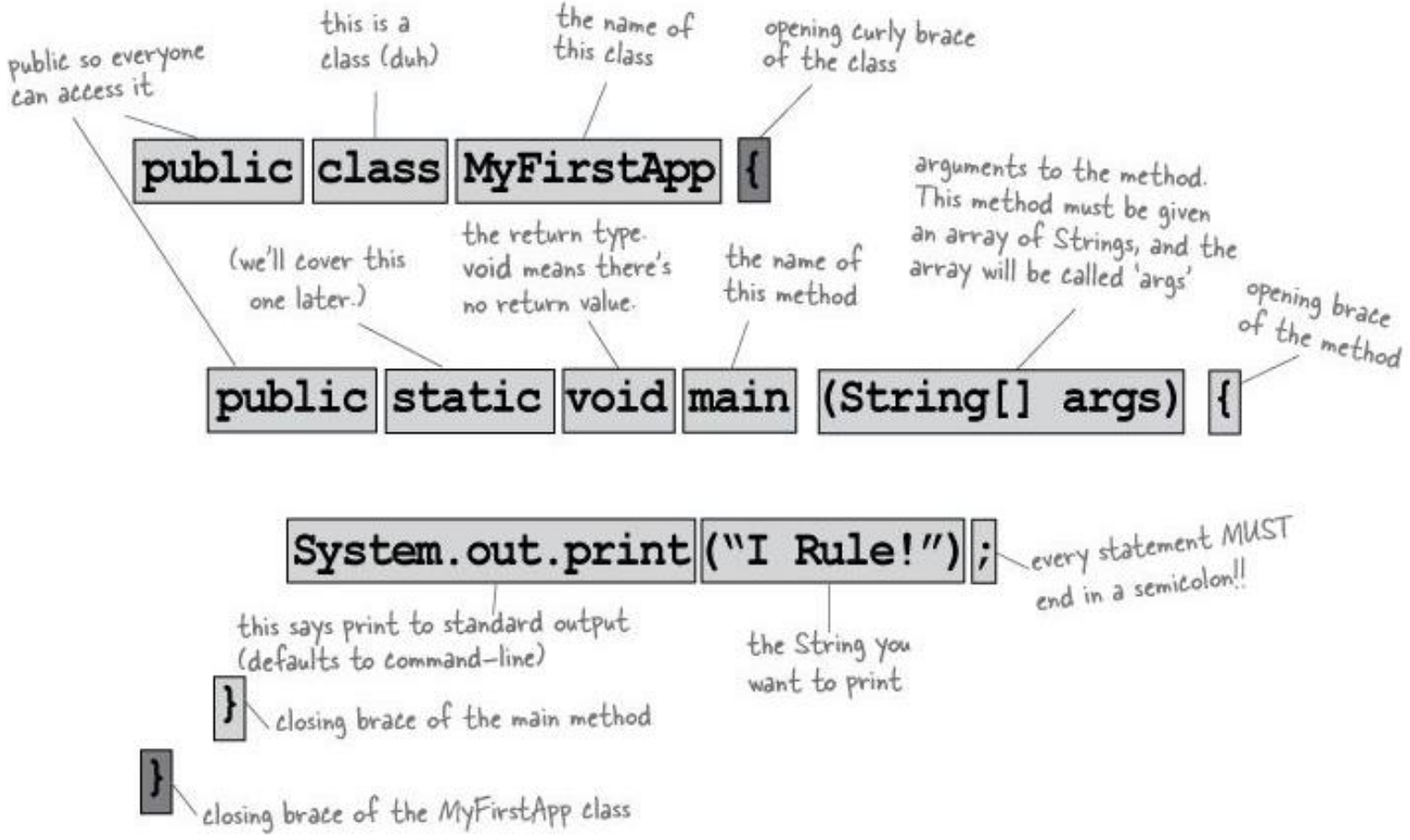
What goes in a method?

- 메소드의 중괄호(curly braces) 내에서 그 메소드가 어떻게 수행되어져야 하는 가에 대한 명령문 작성
- 메소드 코드 => 기본적으로 함수 또는 프로시저와 같은 문장들의 집합



Anatomy of a class

- **JVM** => 메소드의 중괄호 { } 사이에 있는 모든 것을 실행
- 모든 자바 애플리케이션 => 적어도 하나의 클래스와 하나의 **main()** 메소드를 포함해야



Writing a class with a main

How do I write Java code so that it will run? And it all begins with **main().**


```
public class MyFirstApp {

   public static void main (String[] args) {
     System.out.println("I Rule!");
     System.out.println("The World");
   }

}
```

❶ Save

`MyFirstApp.java`

❷ Compile

```
javac MyFirstApp.java
```

❸ Run

```
File Edit Window Help Scream
%java MyFirstApp
I Rule!
The World
```

What can you say in the main method?

컴퓨터가 무언가를 하도록 하기 위해 대부분의 프로그래밍 언어에서 말하는 모든 일반적인 것을 말할 수 있다.

코드에서 **JVM**에게 다음과 같이 말할 수 있다:

1. **do something**
   Statements: declarations, assignments, method calls, etc.

```
int x = 3;
String name = "Dirk";
x = x * 17;
System.out.print(" x is " + x);
double d = Math.random();
// this is a comment
```

2. **do something again and again**
   Loops: for and while

```
while (x > 12) {
   x = x - 1;
}
for (int x = 0; x < 10; x = x + 1) {
   System.out.print(" x is now " + x);
}
```

3. Do something under this condition

Branching: if/else tests

```
if (x == 10) {
   System.out.print("x must be 10");
} else {
   System.out.print("x isn't 10");
}
if (( x < 3) & (name.equals("Dirk"))) {
   System.out.println("Gently");
}
   System.out.print("this line runs no matter what");
```

Looping and looping and...

Java에는 **while**, **do-while** 및 **for**의 세 가지 표준 루핑 구문이 있다.

루프의 핵심은 **조건 테스트**이다.

Java에서 조건부 테스트는 결과가 부울 값, 즉 진실 또는 거짓인 결과이다.

Simple boolean tests

아래와 같은 비교 연산자를 사용하여 변수의 값을 검사하여 간단한 부울 테스트를 수행할 수 있다.

< (less than)

> (greater than)

== (equality) (yes, that's two equals signs)

대입 연산자(단일 등호)와 ***equals*** 연산자(두 개의 등호) 간의 차이점에 주목하자.

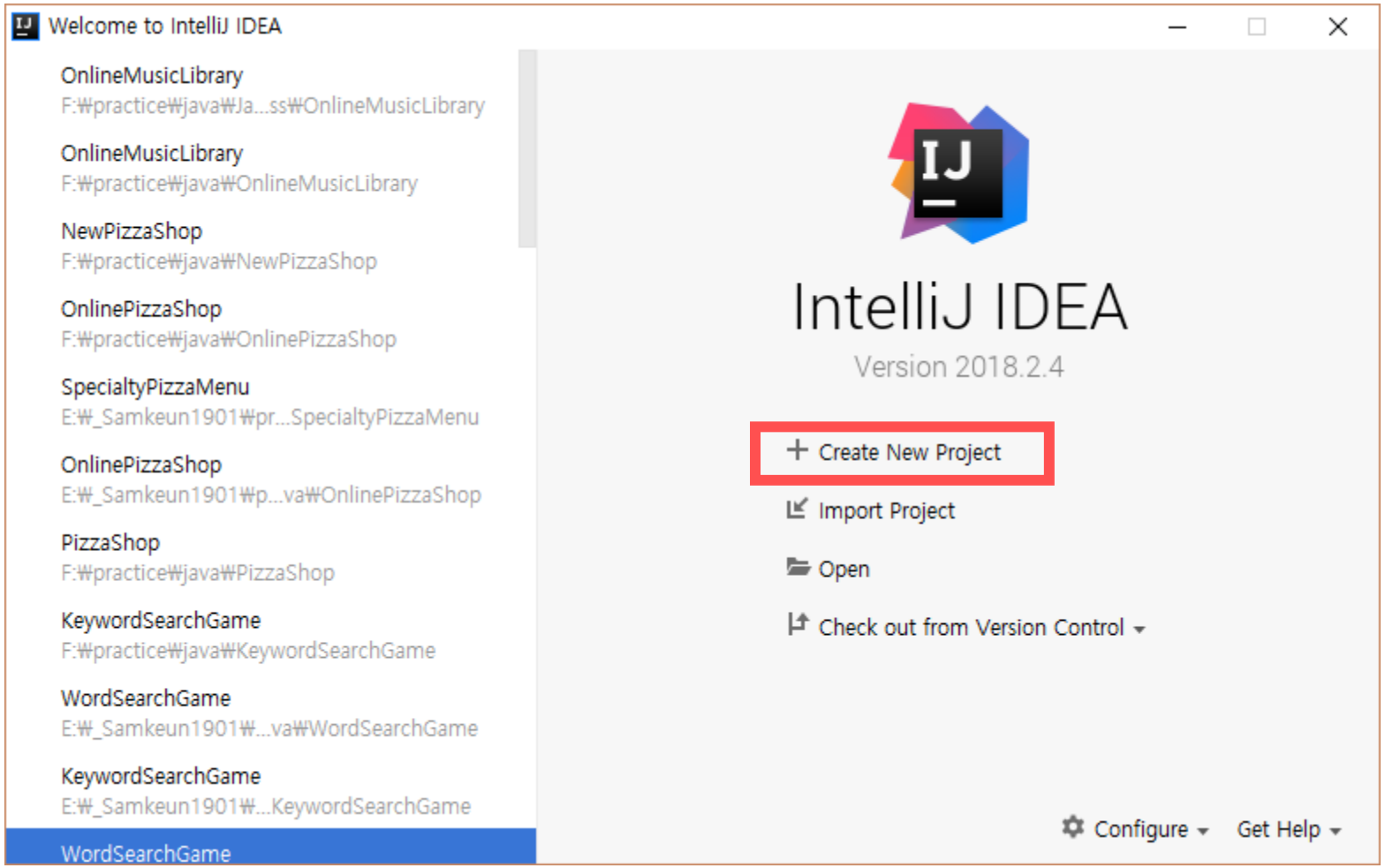
```
int x = 4; // assign 4 to x
while (x > 3) {
   // loop code will run because
   // x is greater than 3
   x = x - 1; // or we'd loop forever
}
int z = 27; //
while (z == 17) {
   // loop code will not run because
   // z is not equal to 17
}
```

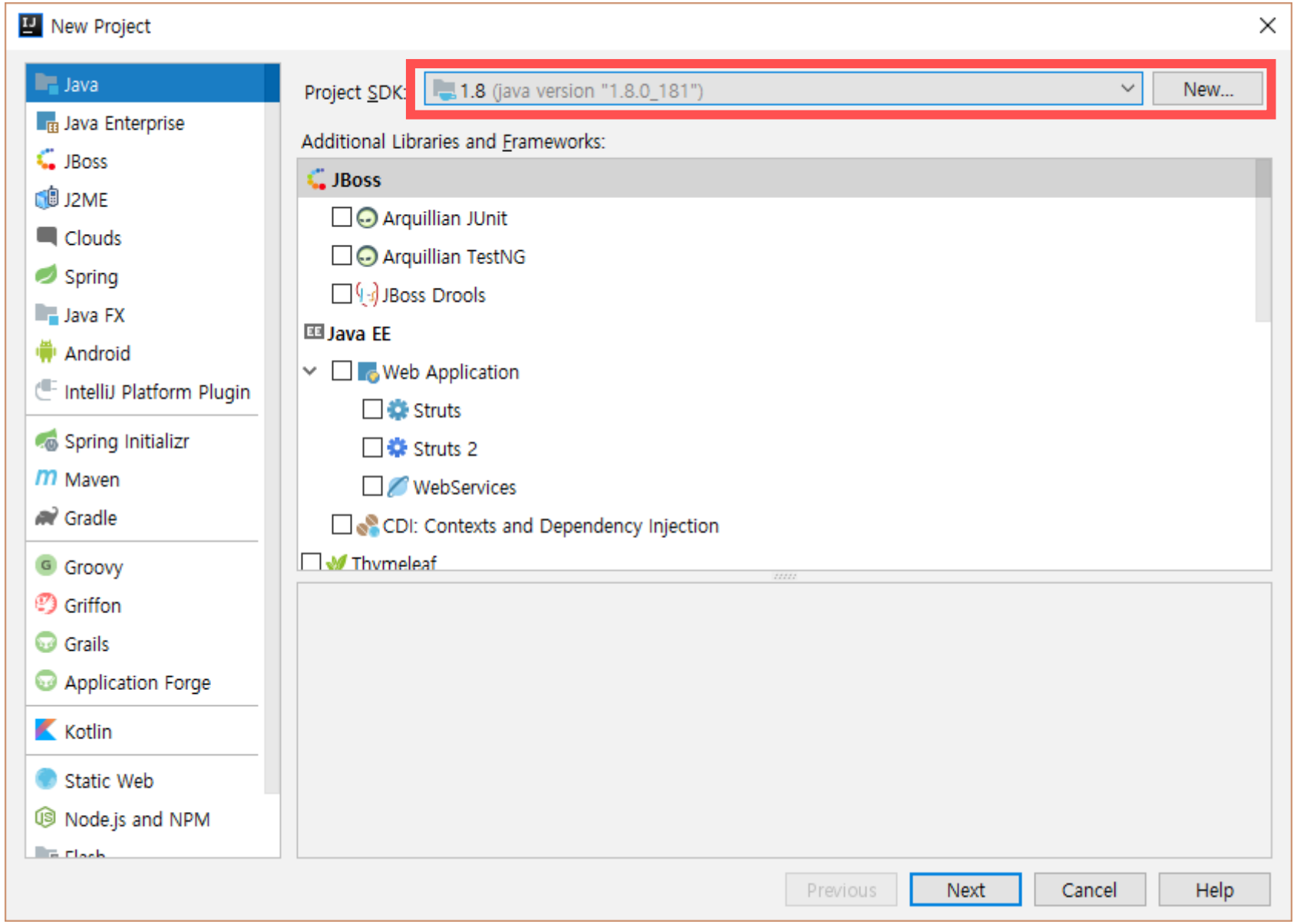
실습과제 1-1 Looping and looping and...

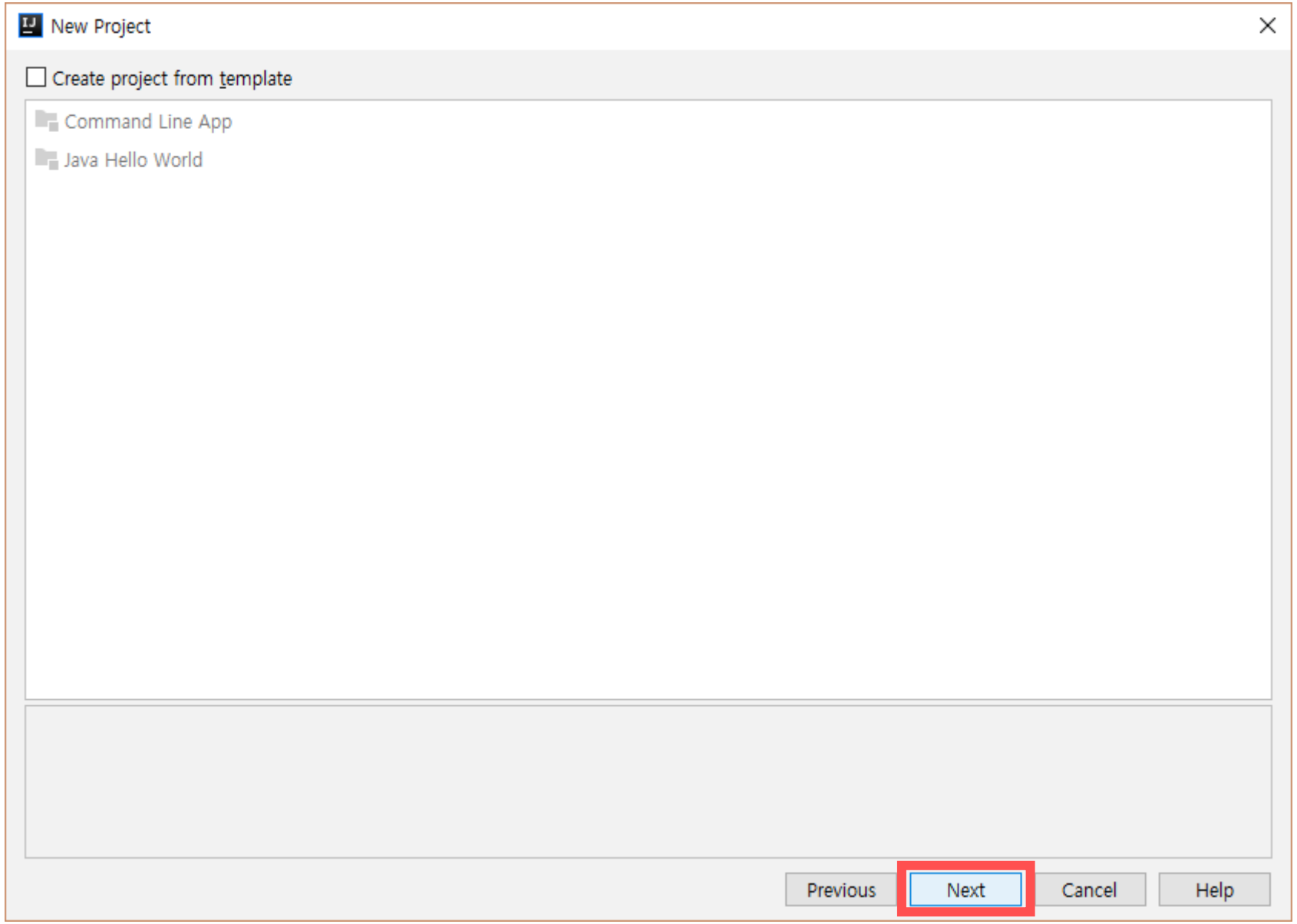
Example of a while loop

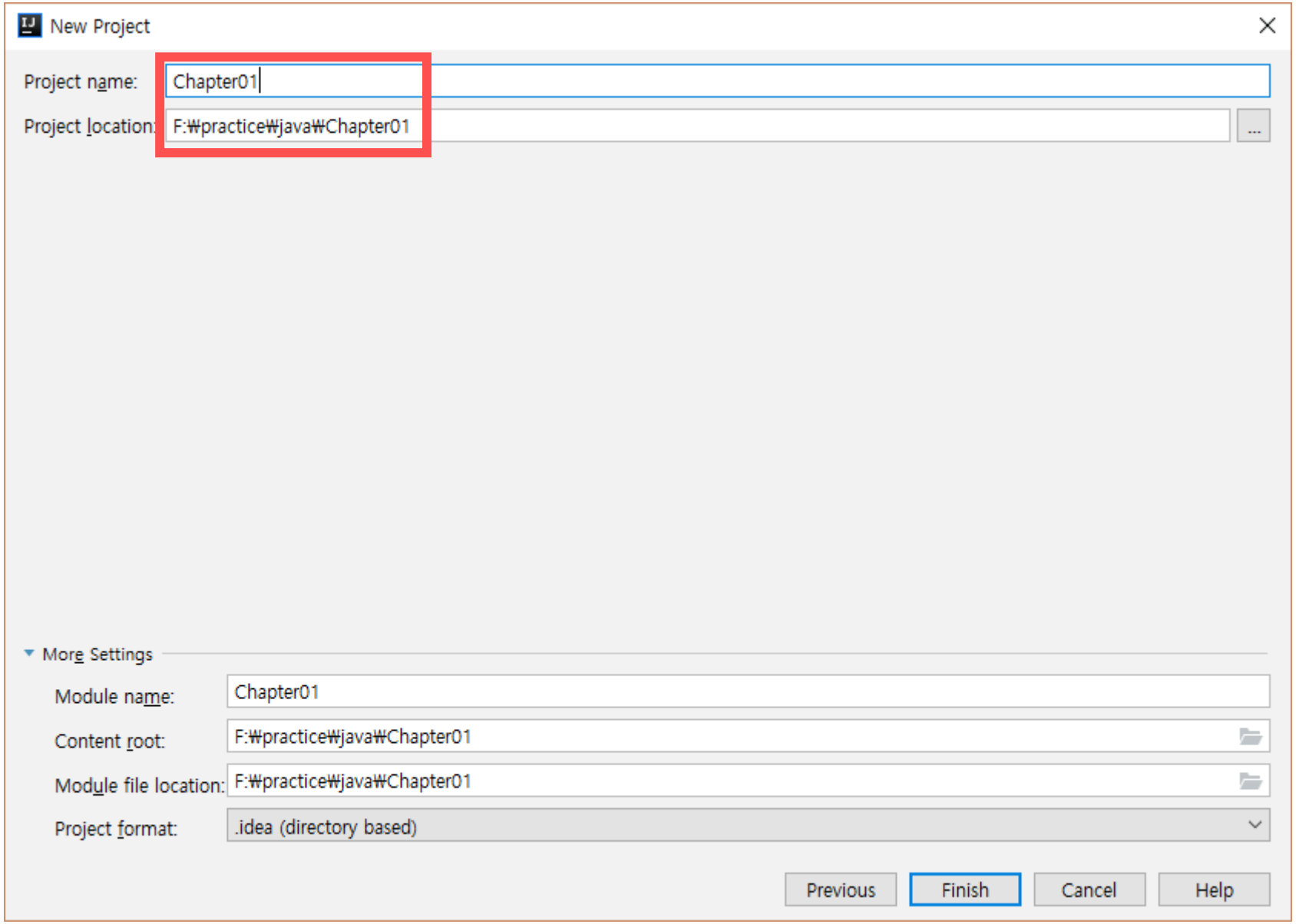
클래스명(Loopy)과 정확히 일치하는 파일명
(Loopy.java)으로 저장해야 한다. (대소문자 구분함)

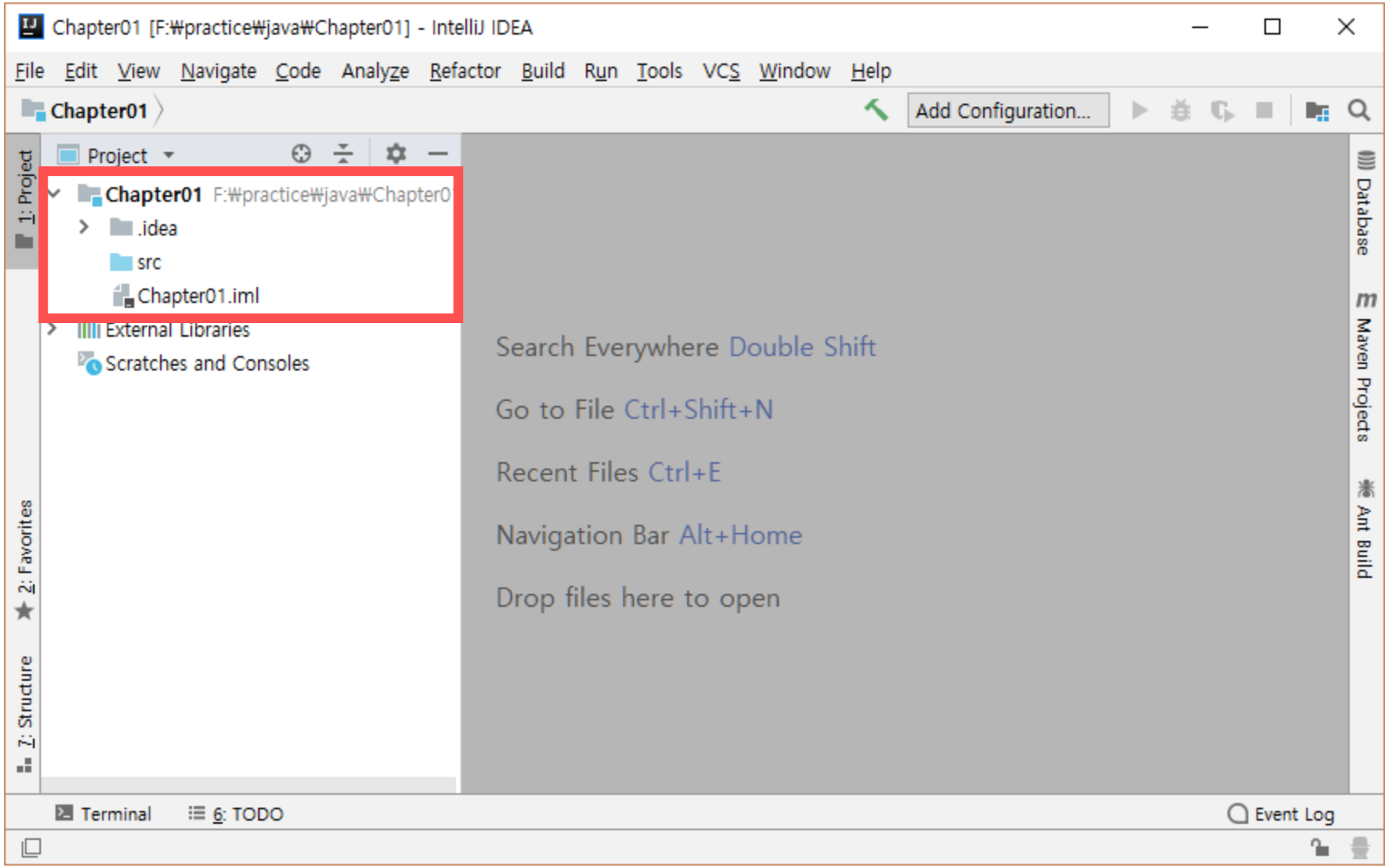
```
public class Loopy {
   public static void main (String[] args) {
     int x = 1;
     System.out.println("Before the Loop");
     while (x < 4) {
       System.out.println("In the loop");
       System.out.println("Value of x is " + x);
         x = x + 1;
     }
     System.out.println("This is after the loop");
   }
```

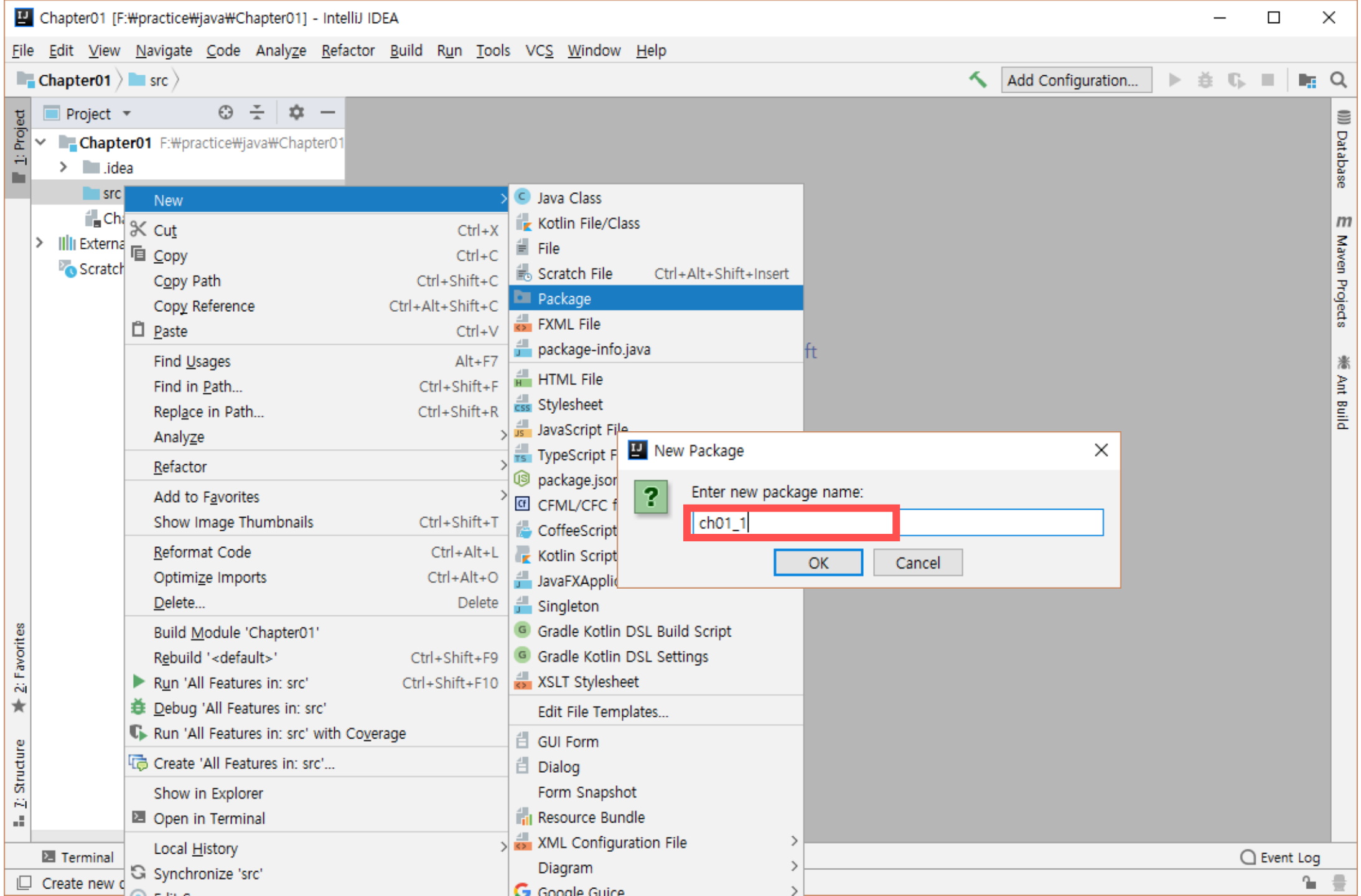

Welcome to IntelliJ IDEA
OnlineMusicLibrary
F:₩practice₩java₩Ja...ss₩OnlineMusicLibrary
OnlineMusicLibrary
F:₩practice₩java₩OnlineMusicLibrary
NewPizzaShop
F:₩practice₩java₩NewPizzaShop
OnlinePizzaShop
F:₩practice₩java₩OnlinePizzaShop
SpecialtyPizzaMenu
E:₩_Samkeun1901₩pr...SpecialtyPizzaMenu
OnlinePizzaShop
E:₩_Samkeun1901₩p...va₩OnlinePizzaShop
PizzaShop
F:₩practice₩java₩PizzaShop
KeywordSearchGame
F:₩practice₩java₩KeywordSearchGame
WordSearchGame
E:₩_Samkeun1901₩...va₩WordSearchGame
KeywordSearchGame
E:₩_Samkeun1901₩...KeywordSearchGame
WordSearchGame
IntelliJ IDEA
Version 2018.2.4
Create New Project
Import Project
Open
Check out from Version Control
Configure
Get Help

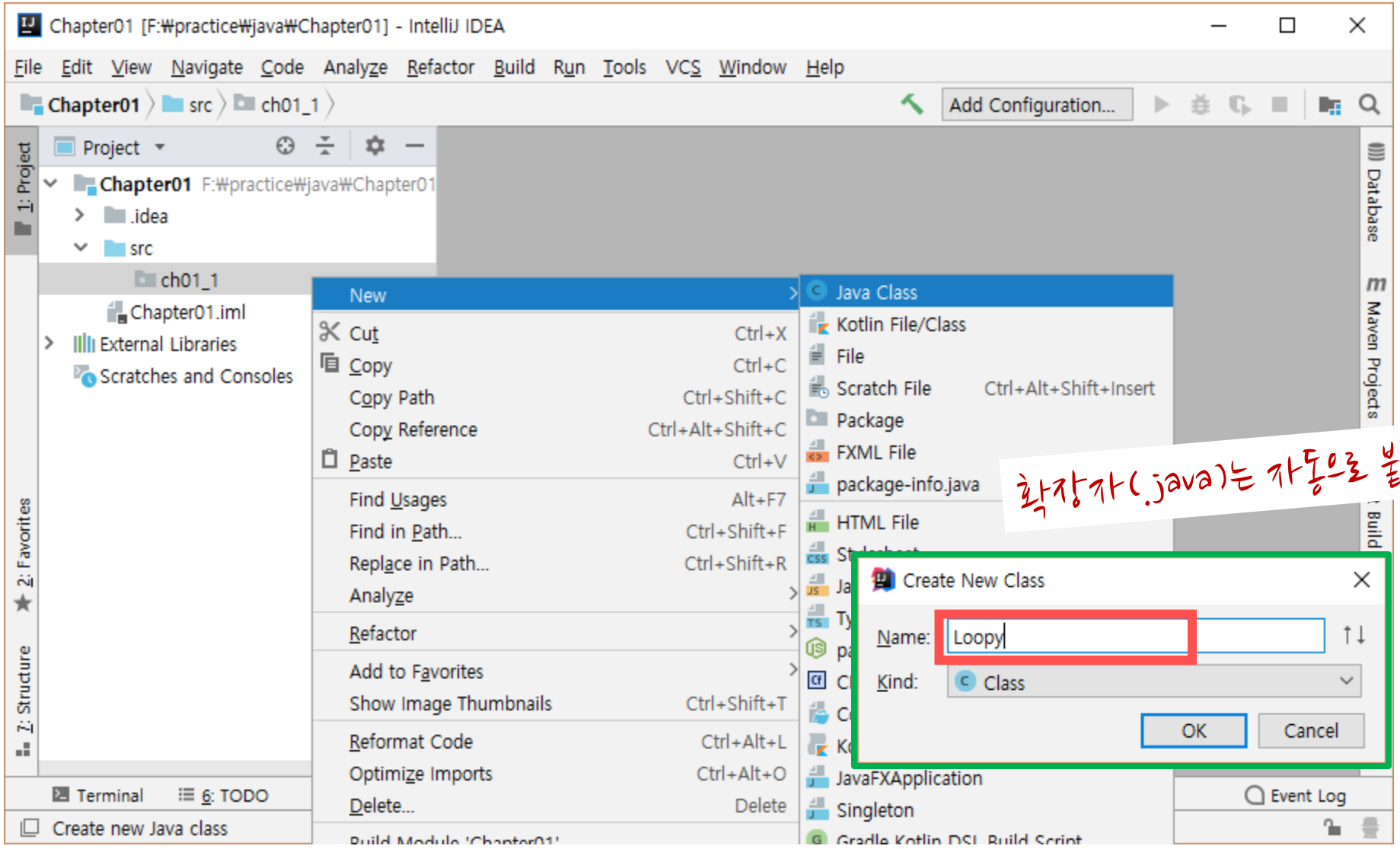

New Project
Java
Java Enterprise
JBoss
J2ME
Clouds
Spring
Java FX
Android
IntelliJ Platform Plugin
Spring Initializr
Maven
Gradle
Groovy
Griffon
Grails
Application Forge
Kotlin
Static Web
Node.js and NPM
Project SDK:
1.8 (java version "1.8.0_181")
New...
Additional Libraries and Frameworks:
JBoss
Arquillian JUnit
Arquillian TestNG
JBoss Drools
Java EE
Web Application
Struts
Struts 2
WebServices
CDI: Contexts and Dependency Injection
Thymeleaf
Previous
Next
Cancel
Help

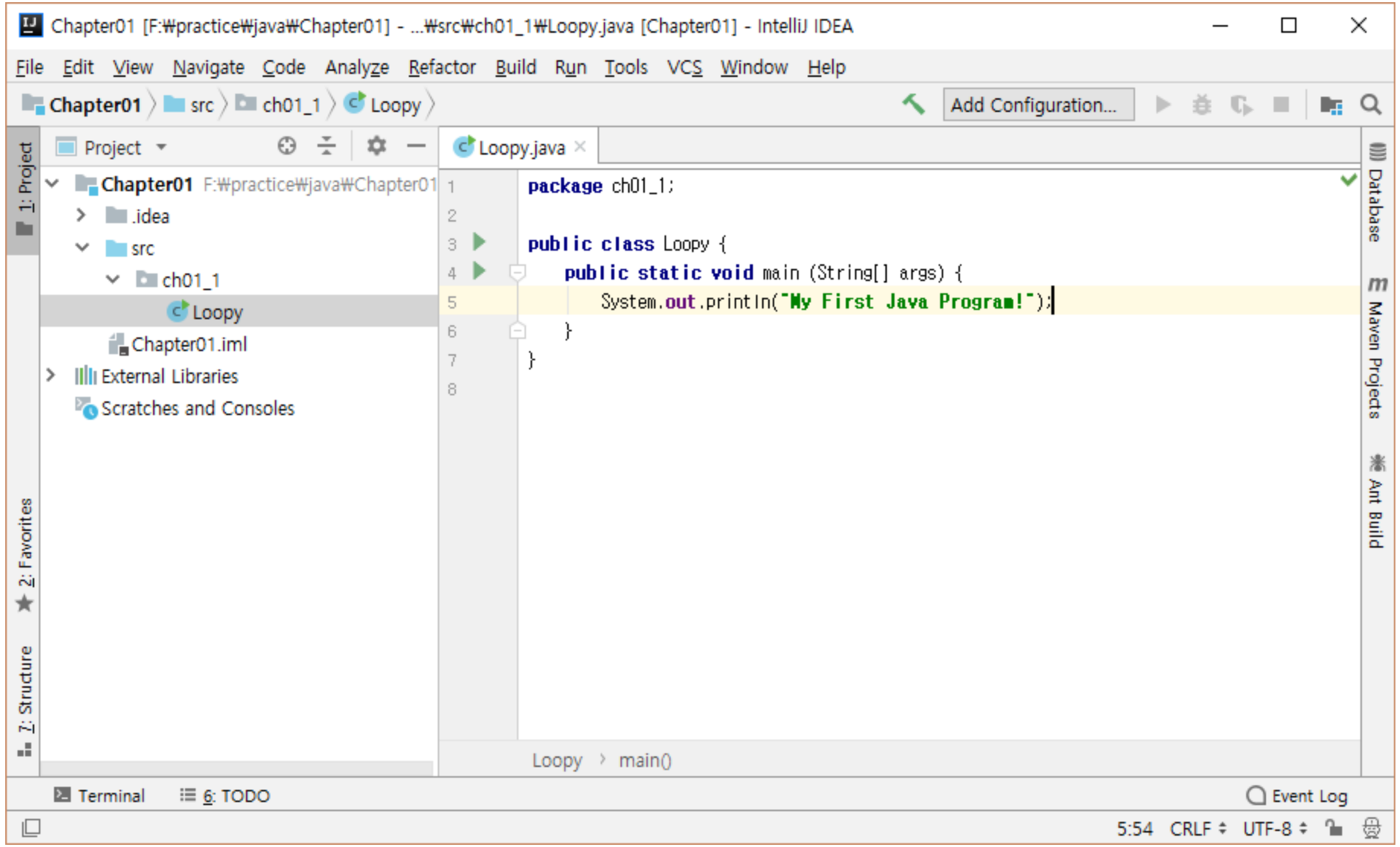

New Project
Create project from template
Command Line App
Java Hello World
Previous
Next
Cancel
Help

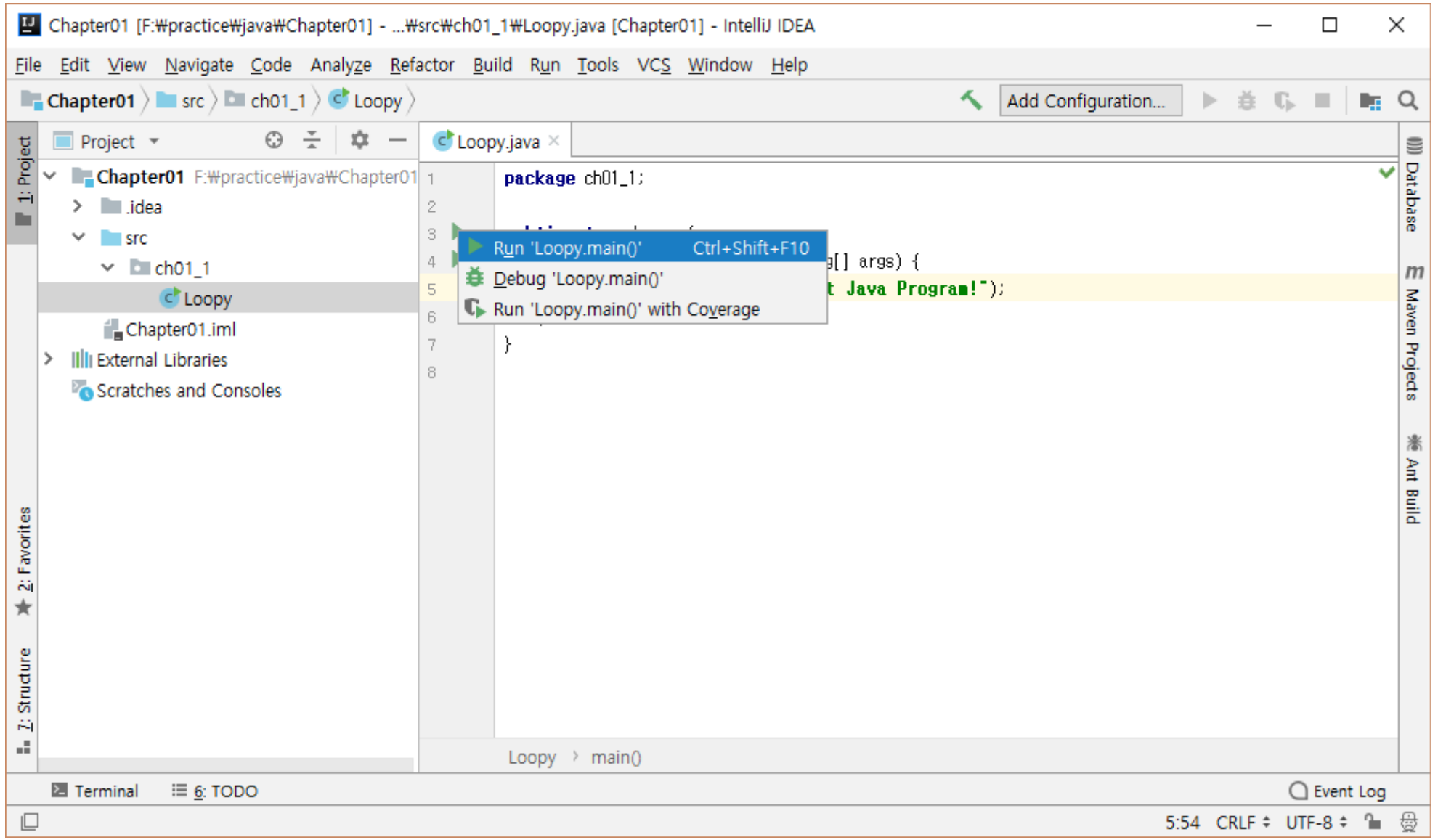

New Project
Project name:
Chapter01
Project location:
F:₩practice₩java₩Chapter01
...
More Settings
Module name:
Chapter01
Content root:
F:₩practice₩java₩Chapter01
Module file location:
F:₩practice₩java₩Chapter01
Project format:
.idea (directory based)
Previous
Finish
Cancel
Help

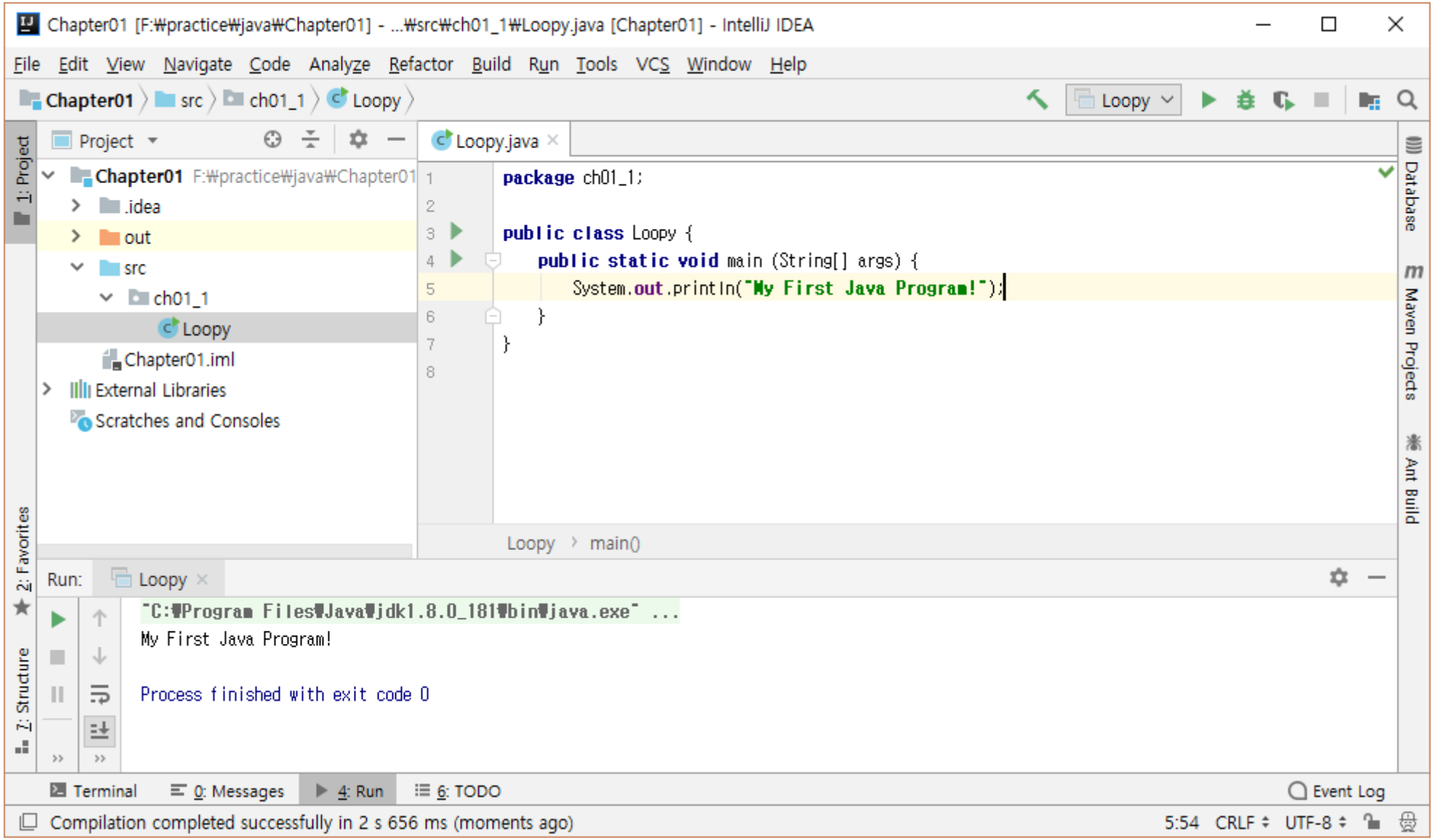

Chapter01 [F:₩practice₩java₩Chapter01] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Chapter01
Add Configuration...
1: Project
Project
Chapter01 F:₩practice₩java₩Chapter0
.idea
src
Chapter01.iml
External Libraries
Scratches and Consoles
Search Everywhere Double Shift
Go to File Ctrl+Shift+N
Recent Files Ctrl+E
Navigation Bar Alt+Home
Drop files here to open
Database
Maven Projects
Ant Build
2: Favorites
7: Structure
Terminal
6: TODO
Event Log

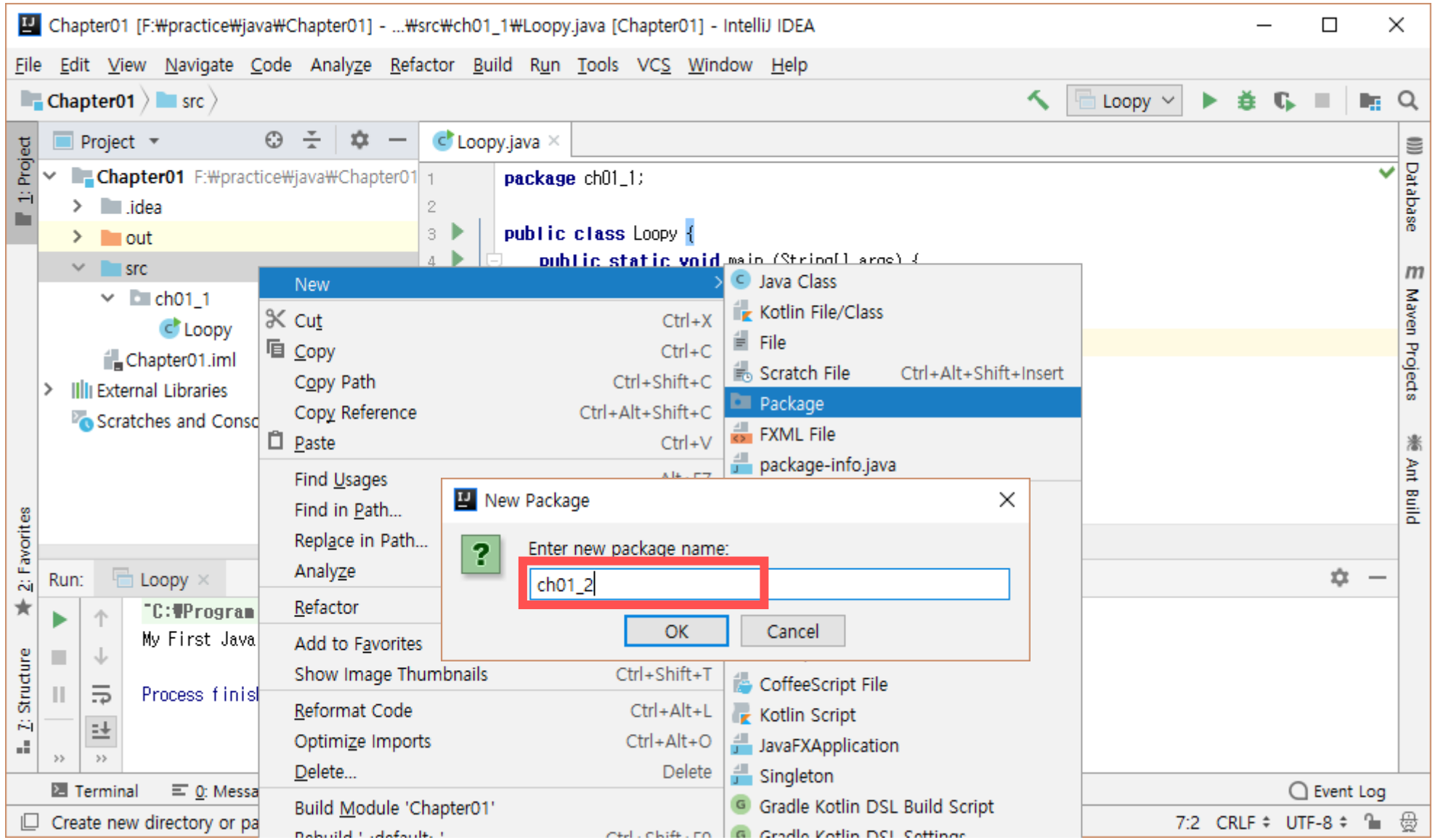

Chapter01 [F:₩practice₩java₩Chapter01] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Chapter01 src
Add Configuration...
Project
Chapter01 F:₩practice₩java₩Chapter01
.idea
src
External Libraries
Scratches and Consoles
New
Cut Ctrl+X
Copy Ctrl+C
Copy Path Ctrl+Shift+C
Copy Reference Ctrl+Alt+Shift+C
Paste Ctrl+V
Find Usages Alt+F7
Find in Path... Ctrl+Shift+F
Replace in Path... Ctrl+Shift+R
Analyze
Refactor
Add to Favorites
Show Image Thumbnails Ctrl+Shift+T
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Delete... Delete
Build Module 'Chapter01'
Rebuild '<default>' Ctrl+Shift+F9
Run 'All Features in: src' Ctrl+Shift+F10
Debug 'All Features in: src'
Run 'All Features in: src' with Coverage
Create 'All Features in: src'...
Show in Explorer
Open in Terminal
Local History
Synchronize 'src'
Java Class
Kotlin File/Class
File
Scratch File Ctrl+Alt+Shift+Insert
Package
FXML File
package-info.java
HTML File
Stylesheet
JavaScript File
TypeScript File
package.json
CFML/CFC file
CoffeeScript
Kotlin Script
JavaFXApplication
Singleton
Gradle Kotlin DSL Build Script
Gradle Kotlin DSL Settings
XSLT Stylesheet
Edit File Templates...
GUI Form
Dialog
Form Snapshot
Resource Bundle
XML Configuration File
Diagram
Google Guice
New Package
Enter new package name:
ch01_1
OK
Cancel
1: Project
2: Favorites
7: Structure
Database
Maven Projects
Ant Build
Terminal
Event Log

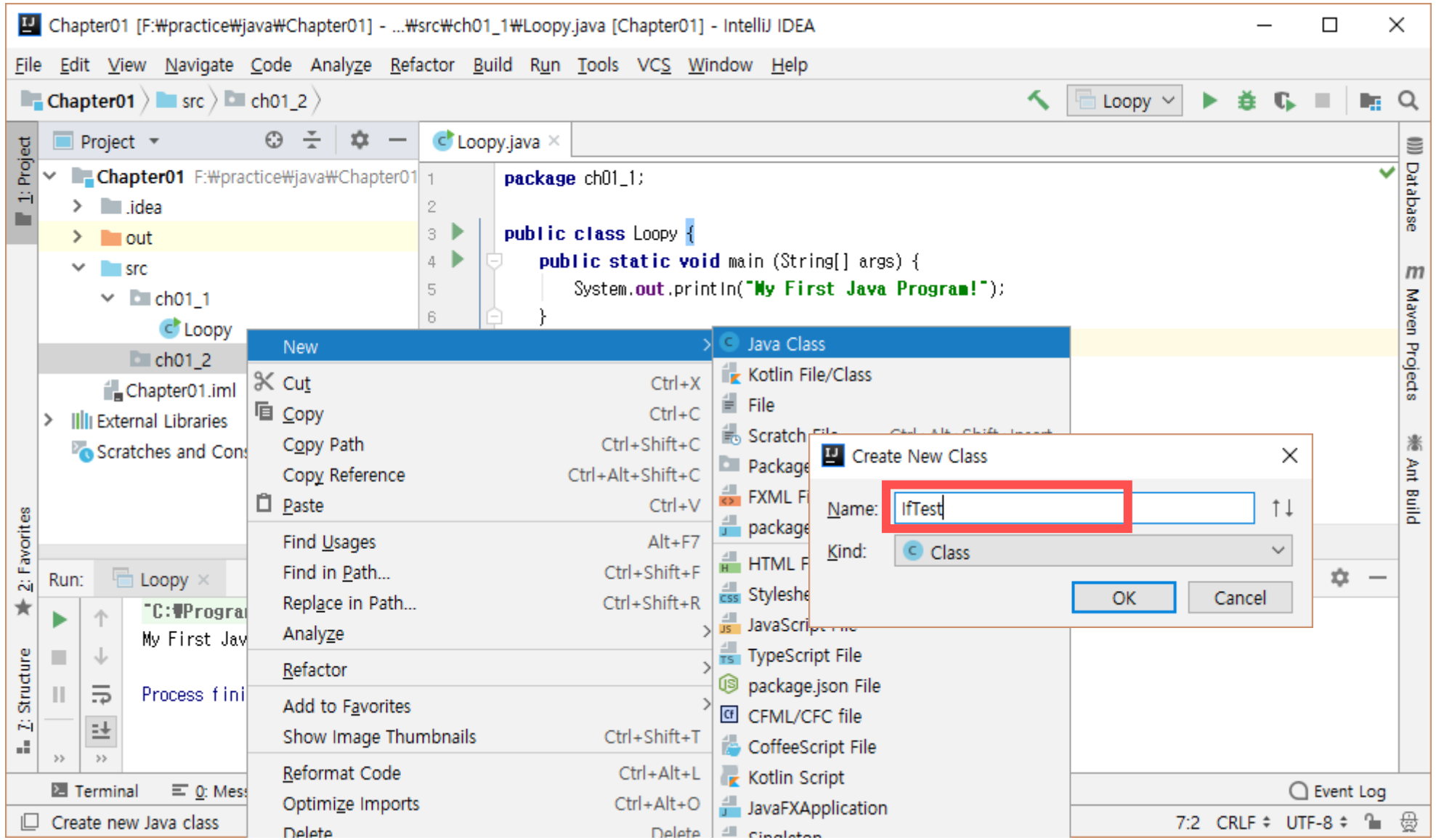

Chapter01 [F:₩practice₩java₩Chapter01] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Chapter01 src ch01_1
Add Configuration...
Project
Chapter01 F:₩practice₩java₩Chapter01
.idea
src
ch01_1
Chapter01.iml
External Libraries
Scratches and Consoles
New
Cut Ctrl+X
Copy Ctrl+C
Copy Path Ctrl+Shift+C
Copy Reference Ctrl+Alt+Shift+C
Paste Ctrl+V
Find Usages Alt+F7
Find in Path... Ctrl+Shift+F
Replace in Path... Ctrl+Shift+R
Analyze
Refactor
Add to Favorites
Show Image Thumbnails Ctrl+Shift+T
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Delete... Delete
Java Class
Kotlin File/Class
File
Scratch File Ctrl+Alt+Shift+Insert
Package
FXML File
package-info.java
HTML File
JavaFXApplication
Singleton
Create New Class
Name: Loopy
Kind: Class
OK
Cancel
확장자(.java)는 자동으로 붙는다!
1: Project
2: Favorites
7: Structure
Database
Maven Projects
Terminal
6: TODO
Event Log
Create new Java class

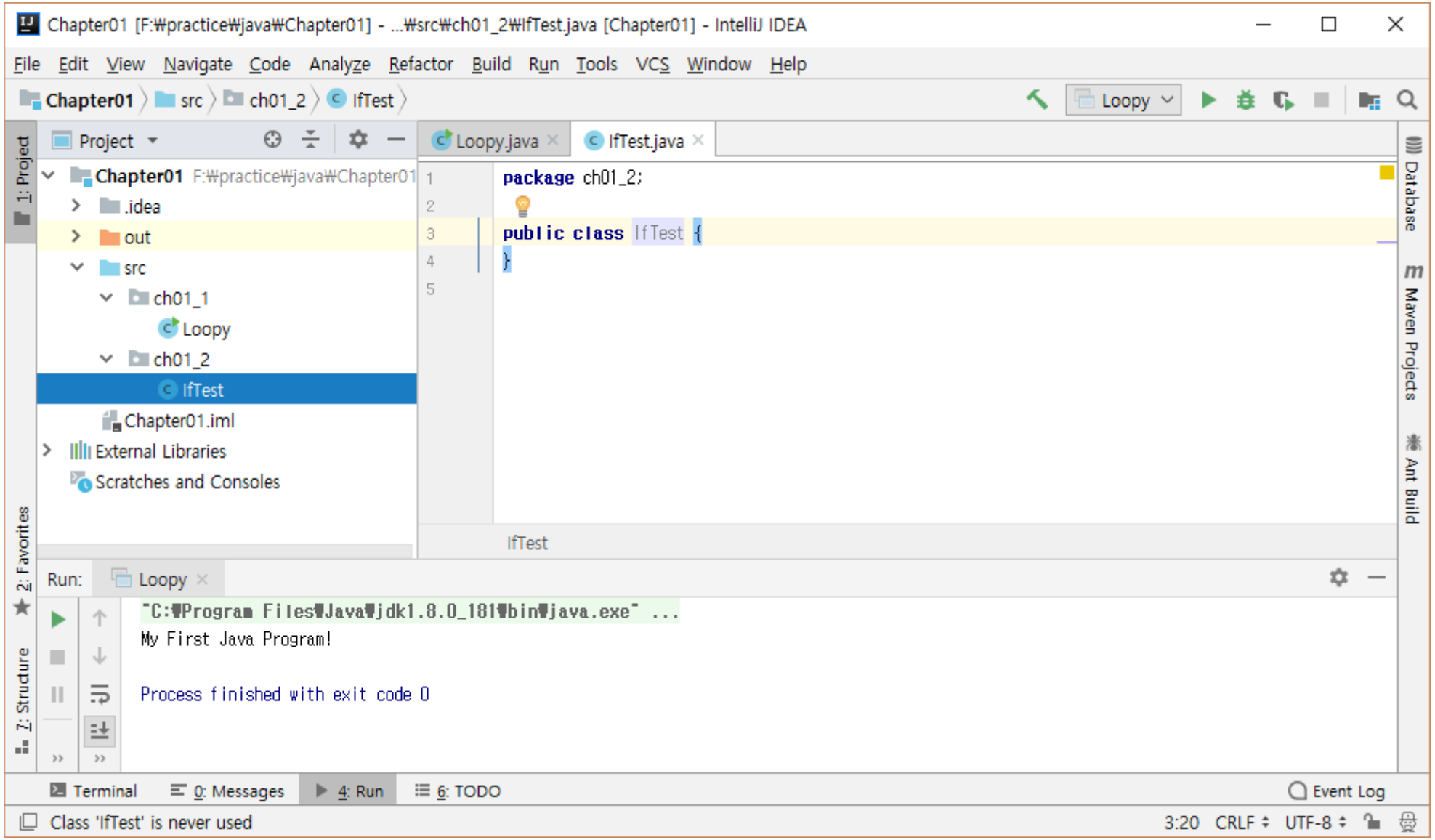

Chapter01 [F:₩practice₩java₩Chapter01] - ...₩src₩ch01_1₩Loopy.java [Chapter01] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Chapter01 src ch01_1 Loopy
Add Configuration...
Project
Chapter01 F:₩practice₩java₩Chapter01
.idea
src
ch01_1
Loopy
Chapter01.iml
External Libraries
Scratches and Consoles
Loopy.java
package ch01_1;
public class Loopy {
public static void main (String[] args) {
System.out.println("My First Java Program!");
}
}
Loopy › main()
1: Project
2: Favorites
7: Structure
Database
Maven Projects
Ant Build
Terminal
6: TODO
Event Log
5:54 CRLF UTF-8


Chapter01 [F:₩practice₩java₩Chapter01] - ...₩src₩ch01_1₩Loopy.java [Chapter01] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Chapter01 src ch01_1 Loopy
Add Configuration...
Project
Chapter01 F:₩practice₩java₩Chapter01
.idea
src
ch01_1
Loopy
Chapter01.iml
External Libraries
Scratches and Consoles
Loopy.java
1 package ch01_1;
2
3
4
5
6
7 }
8
Run 'Loopy.main()' Ctrl+Shift+F10
Debug 'Loopy.main()'
Run 'Loopy.main()' with Coverage
g[] args) {
t Java Program!");
Loopy › main()
1: Project
2: Favorites
7: Structure
Database
Maven Projects
Ant Build
Terminal
6: TODO
Event Log
5:54 CRLF UTF-8


Chapter01 [F:₩practice₩java₩Chapter01] - ...₩src₩ch01_1₩Loopy.java [Chapter01] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Chapter01 src ch01_1 Loopy
Loopy
Project
Chapter01 F:₩practice₩java₩Chapter01
.idea
out
src
ch01_1
Loopy
Chapter01.iml
External Libraries
Scratches and Consoles
Loopy.java
1 package ch01_1;
2
3 public class Loopy {
4 public static void main (String[] args) {
5 System.out.println("My First Java Program!");
6 }
7 }
8
Loopy › main()
Run: Loopy
"C:₩Program Files₩Java₩jdk1.8.0_181₩bin₩java.exe" ...
My First Java Program!
Process finished with exit code 0
Terminal 0: Messages 4: Run 6: TODO Event Log
Compilation completed successfully in 2 s 656 ms (moments ago)
5:54 CRLF UTF-8
1: Project
2: Favorites
7: Structure
Database
Maven Projects
Ant Build


Chapter01 [F:₩practice₩java₩Chapter01] - ...₩src₩ch01_1₩Loopy.java [Chapter01] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Chapter01 src
Loopy
Project
Chapter01 F:₩practice₩java₩Chapter01
.idea
out
src
ch01_1
Loopy
Chapter01.iml
External Libraries
Scratches and Consc
Loopy.java
package ch01_1;
public class Loopy {
public static void main (String[] args) {
New
Cut Ctrl+X
Copy Ctrl+C
Copy Path Ctrl+Shift+C
Copy Reference Ctrl+Alt+Shift+C
Paste Ctrl+V
Find Usages
Find in Path...
Replace in Path...
Analyze
Refactor
Add to Favorites
Show Image Thumbnails Ctrl+Shift+T
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Delete... Delete
Build Module 'Chapter01'
Java Class
Kotlin File/Class
File
Scratch File Ctrl+Alt+Shift+Insert
Package
FXML File
package-info.java
CoffeeScript File
Kotlin Script
JavaFXApplication
Singleton
Gradle Kotlin DSL Build Script
New Package
Enter new package name:
ch01_2
OK
Cancel
Run: Loopy
"C:₩Progra
My First Java
Process finis
Terminal
0: Messa
Event Log
Create new directory or pa
7:2 CRLF UTF-8
1: Project
2: Favorites
7: Structure
Database
Maven Projects
Ant Build


Chapter01 [F:₩practice₩java₩Chapter01] - ...₩src₩ch01_1₩Loopy.java [Chapter01] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Chapter01 src ch01_2
Loopy
Project
Chapter01 F:₩practice₩java₩Chapter01
.idea
out
src
ch01_1
Loopy
ch01_2
Chapter01.iml
External Libraries
Scratches and Cons
Loopy.java
package ch01_1;
public class Loopy {
public static void main (String[] args) {
System.out.println("My First Java Program!");
}
New
Cut Ctrl+X
Copy Ctrl+C
Copy Path Ctrl+Shift+C
Copy Reference Ctrl+Alt+Shift+C
Paste Ctrl+V
Find Usages Alt+F7
Find in Path... Ctrl+Shift+F
Replace in Path... Ctrl+Shift+R
Analyze
Refactor
Add to Favorites
Show Image Thumbnails Ctrl+Shift+T
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Delete
Java Class
Kotlin File/Class
File
Scratch
Package
FXML
package
HTML
Styleshe
JavaScript
TypeScript File
package.json File
CFML/CFC file
CoffeeScript File
Kotlin Script
JavaFXApplication
Create New Class
Name: IfTest
Kind: Class
OK
Cancel
Run: Loopy
"C:₩Progra
My First Jav
Process fini
Terminal
0: Mess
Create new Java class
Event Log
7:2 CRLF UTF-8
1: Project
2: Favorites
7: Structure
Database
Maven Projects
Ant Build


Chapter01 [F:₩practice₩java₩Chapter01] - ...₩src₩ch01_2₩IfTest.java [Chapter01] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Chapter01 src ch01_2 IfTest
Loopy
Project
Chapter01 F:₩practice₩java₩Chapter01
.idea
out
src
ch01_1
Loopy
ch01_2
IfTest
Chapter01.iml
External Libraries
Scratches and Consoles
Loopy.java
IfTest.java
1 package ch01_2;
2
3 public class IfTest {
4 }
5
IfTest
Run:
Loopy
"C:₩Program Files₩Java₩jdk1.8.0_181₩bin₩java.exe" ...
My First Java Program!
Process finished with exit code 0
1: Project
2: Favorites
7: Structure
Database
Maven Projects
Ant Build
Terminal
0: Messages
4: Run
6: TODO
Event Log
Class 'IfTest' is never used
3:20 CRLF UTF-8

실습과제 1-2 Conditional branching

1)
```
class IfTest {
  public static void main (String[] args) {
    int x = 3;
    if (x == 3) {
      System.out.println("x must be 3");
    }
    System.out.println("This runs no matter what");
  }
}
```

2)
```
class IfTest2 {
  public static void  main (String[] args) {
    int x = 2;
    if (x == 3) {
      System.out.println("x must be 3");
    } else {
       System.out.println("x is NOT 3");
    }
    System.out.println("This runs no matter what");
  }
}
```

Coding a Serious Business Application

Before you look at the code on this page, think for a moment about how you would code that classic children's favorite, "**99 bottles of beer**."

```
public class BeerSong {
   public static void main (String[] args) {
     int beerNum = 99;
     String word = "bottles";

   while (beerNum > 0) {

      if (beerNum == 1) {
        word = "bottle"; // singular, as in ONE bottle.
      }

      System.out.println(beerNum + " " + word + " of beer on the wall");
      System.out.println(beerNum + " " + word + " of beer.");
      System.out.println("Take one down.");
      System.out.println("Pass it around.");
      beerNum = beerNum - 1;

      if (beerNum > 0) {
         System.out.println(beerNum + " " + word + " of beer on the wall");
      } else {
         System.out.println("No more bottles of beer on the wall");
     } // end else
   } // end while loop
 } // end main method
} // end class
```

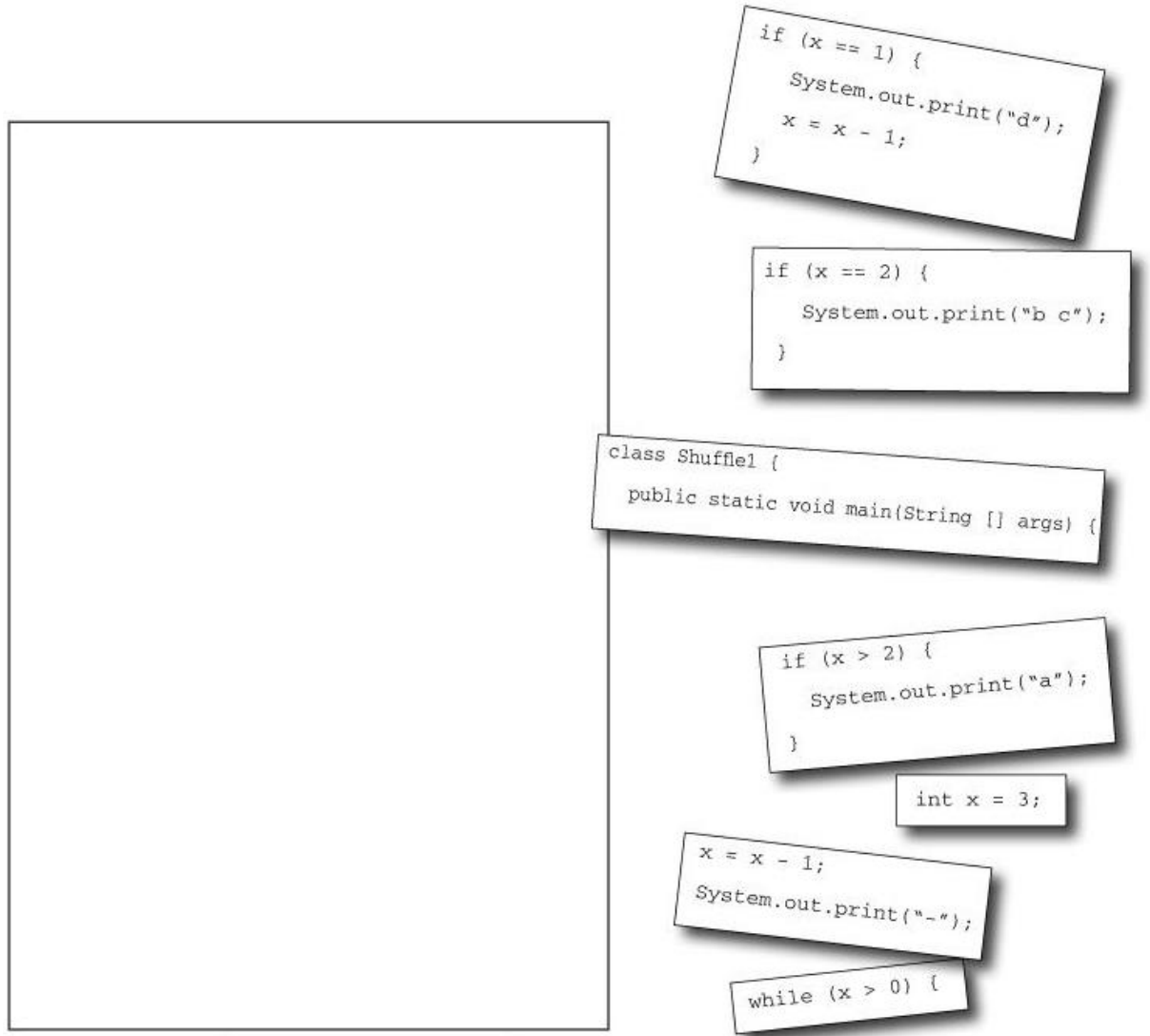
실습과제 1-3 Coding a Serious Business Application

BeerSong 구현하기

+

There's still one little flaw in our code. It compiles and runs, but the output isn't 100% perfect. See if you can spot the flaw, and fix it.

실습과제 1-4 Code Magnets



```
if (x == 1) {
  System.out.print("d");
  x = x - 1;
}
```

```
if (x == 2) {
  System.out.print("b c");
}
```

```
class Shuffle1 {
  public static void main(String [] args) {
```

```
if (x > 2) {
  System.out.print("a");
}
```

```
int x = 3;
```

```
x = x - 1;
System.out.print("-");
```

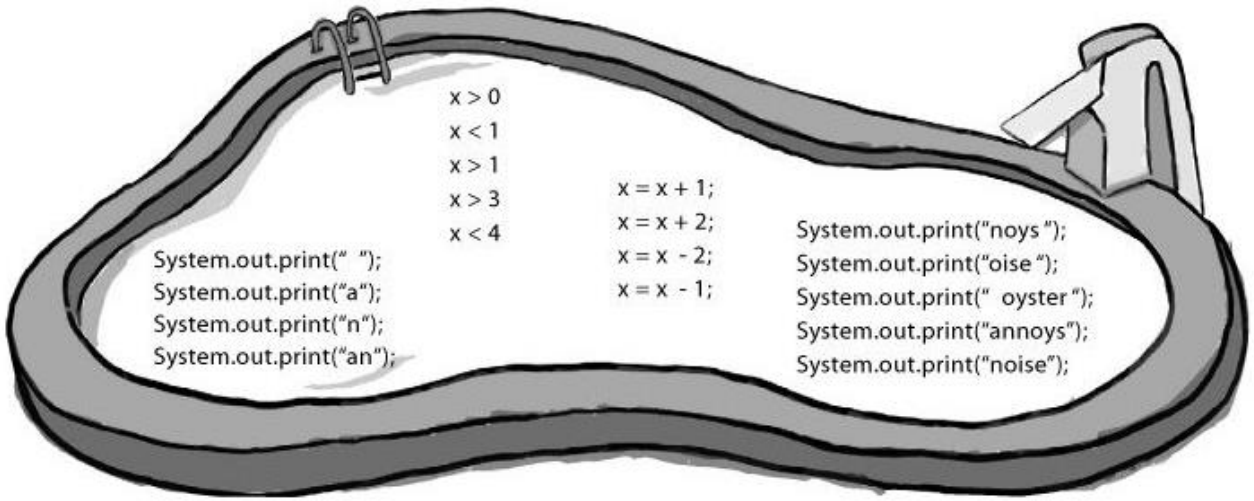
```
while (x > 0) {
```

Output:

```
File Edit Window Help Sleep
% java Shuffle1
a-b c-d
```

실습과제 1-5 Pool Puzzle

```
File  Edit  Window  Help  Cheat
%java PoolPuzzleOne
a noise
annoys
an oyster
```



Note: Each snippet from the pool can be used only once!

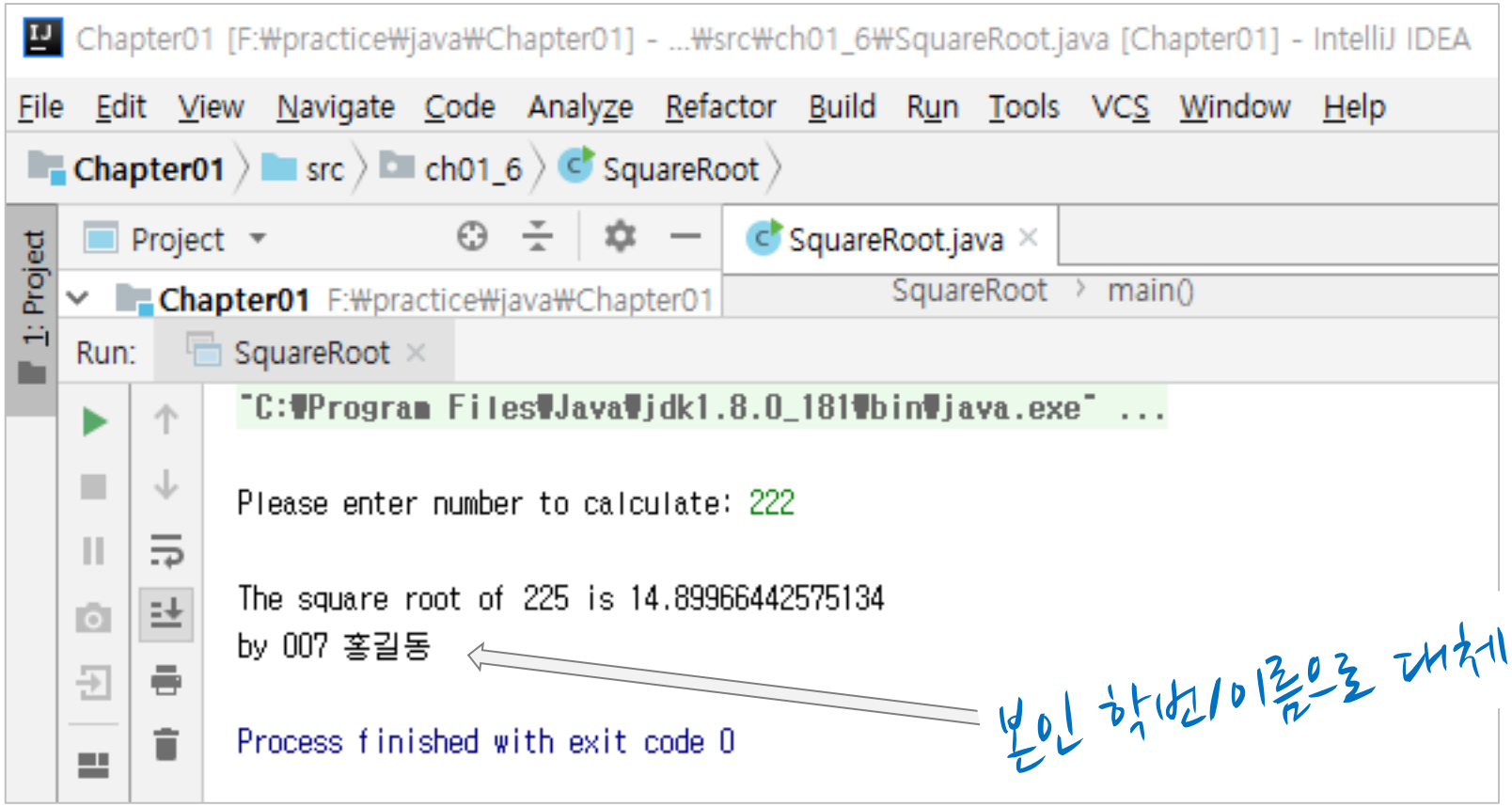
```
class PoolPuzzleOne {
  public static void main(String [] args) {
    int x = 0;

    while ( _____ ) {

      _____
      if ( x < 1 ) {
        _____
      }
      _____
      if ( _____ ) {
        _____
        _____
      }
      if ( x == 1 ) {
        _____
      }
      if ( _____ ) {
        _____
      }
      System.out.println("");
      _____
    }
  }
}
```

실습과제 1-6

제곱근 구하기: 임의의 정수를 입력 받아 제곱근을 계산하여 출력하는 프로그램을 작성하시오.



실습과제 1-7 PhraseOMatic 예제

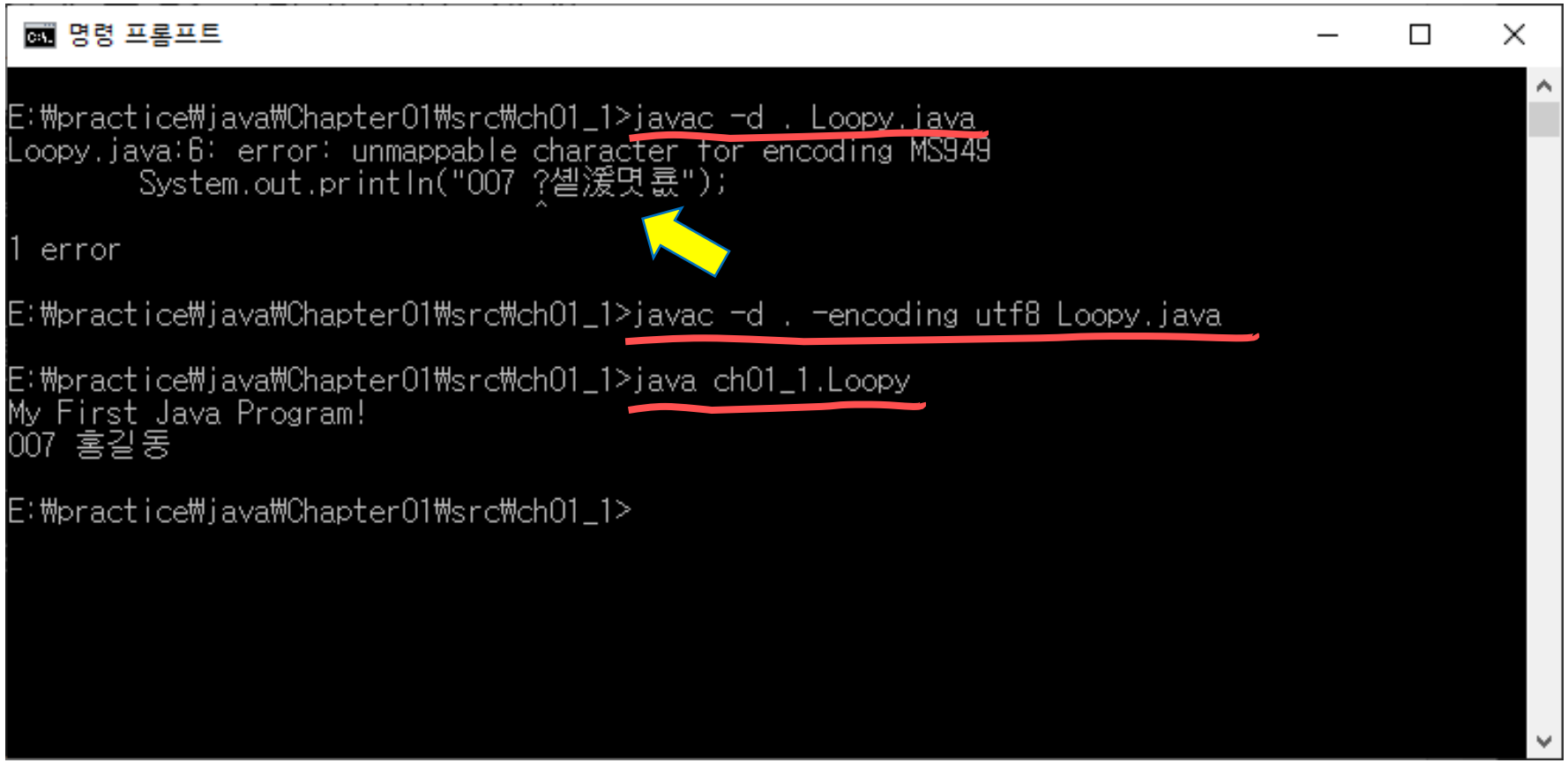
에디터(like EditPlus) 작업 => 파일명: PhraseOMatic.java 파일명: 대소문자 구분

```
1  package Ch01_7;
2
3  public class PhraseOMatic {
4      public static void main(String[] args) {
5          String[] wordListOne = {"24/7", "multi-Tier", "30,000 foot", "B-to-B",
6                  "win-win", "front-end", "web-based", "pervasive", "smart", "six-sigma",
7                  "critical-path", "dynamic"};
8          String[] wordListTwo = {"empowered", "sticky", "value-added", "oriented", "centric",
9                  "distributed", "clustered", "branded", "outside-the-box", "positioned", "networked",
10                 "focused", "leveraged", "aligned", "targeted", "shared", "cooperative", "accelerated"};
11         String[] wordListThree = {"process", "tipping-point", "solution", "architecture", "core competency",
12                 "strategy", "mindshare", "portal", "space", "vision", "paradigm", "mission"};
13         int oneLength = wordListOne.length;
14         int twoLength = wordListTwo.length;
15         int threeLength = wordListThree.length;
16         int rand1 = (int) (Math.random() * oneLength);
17         int rand2 = (int) (Math.random() * twoLength);
18         int rand3 = (int) (Math.random() * threeLength);
19
20         String phrase = wordListOne[rand1] + " " + wordListTwo[rand2] + " " + wordListThree[rand3];
21         System.out.println("What we need is a " + phrase);
22     }
23 }
24
```

패키지명 부여!

DOS command line으로 실행 (Optional)

명령 프롬프트 창 열기: 윈도 검색: 층 => 명령 프롬프트 선택!



패키지 (package)

패키지란

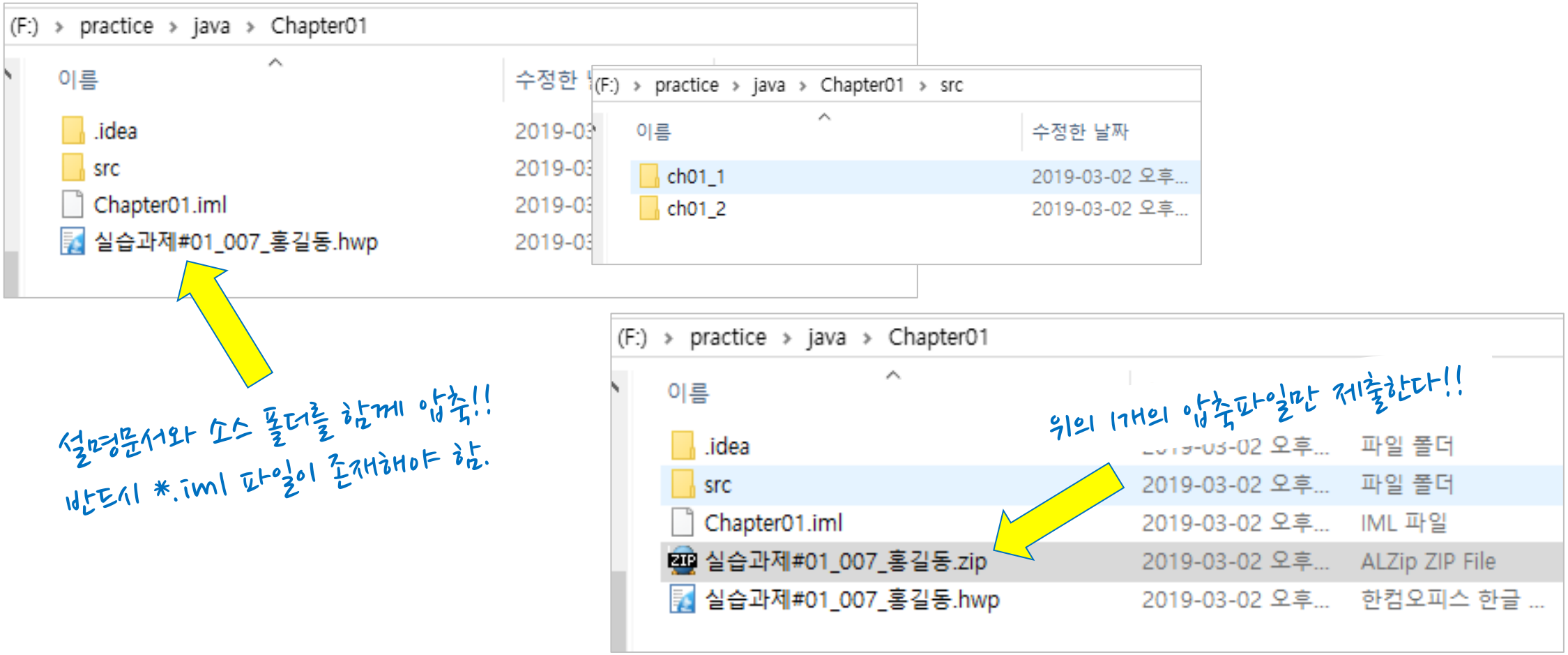
- ✓ 관련된 클래스 또는 인터페이스들을 묶어놓은 상자와 같다

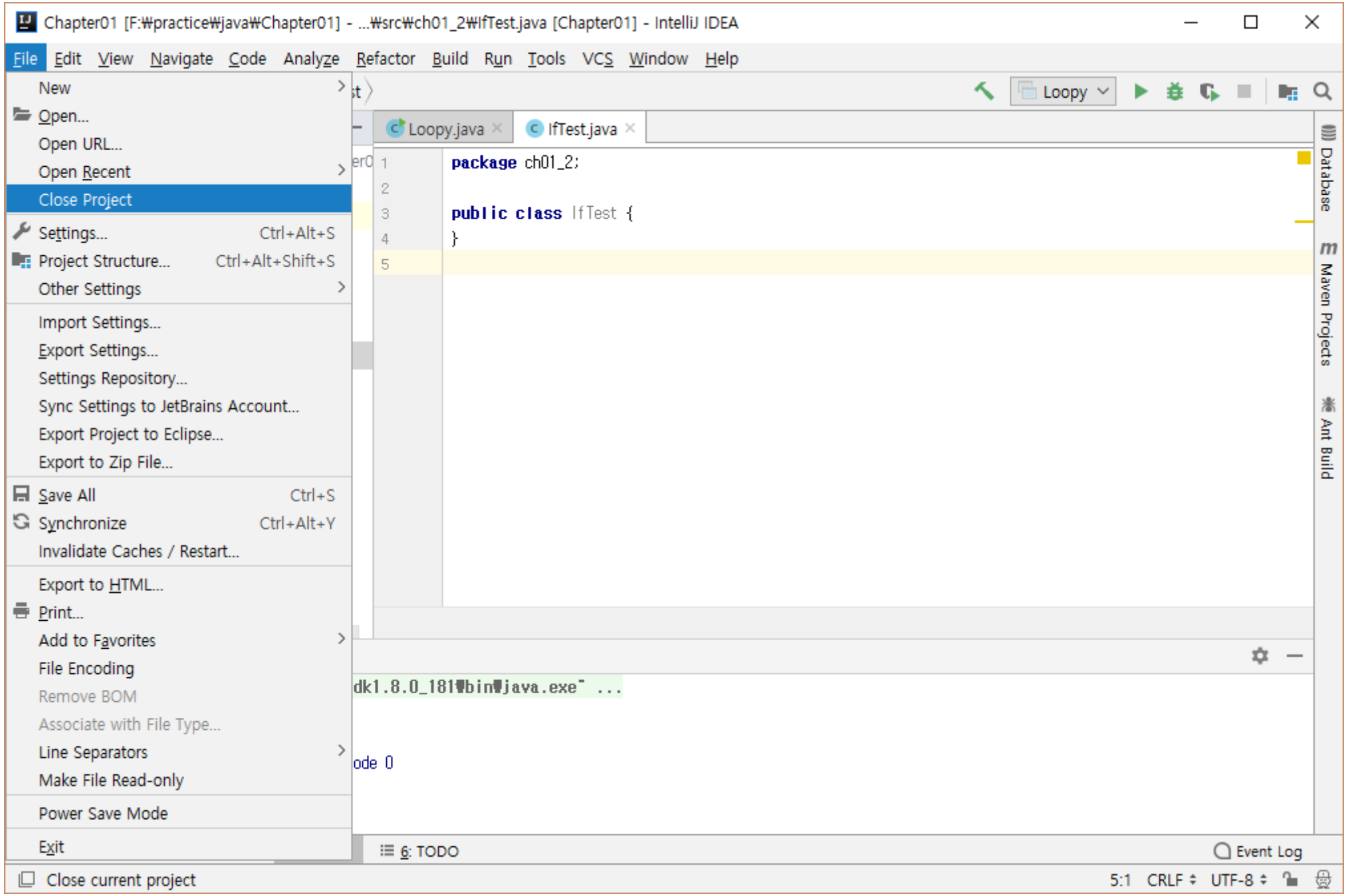
특징

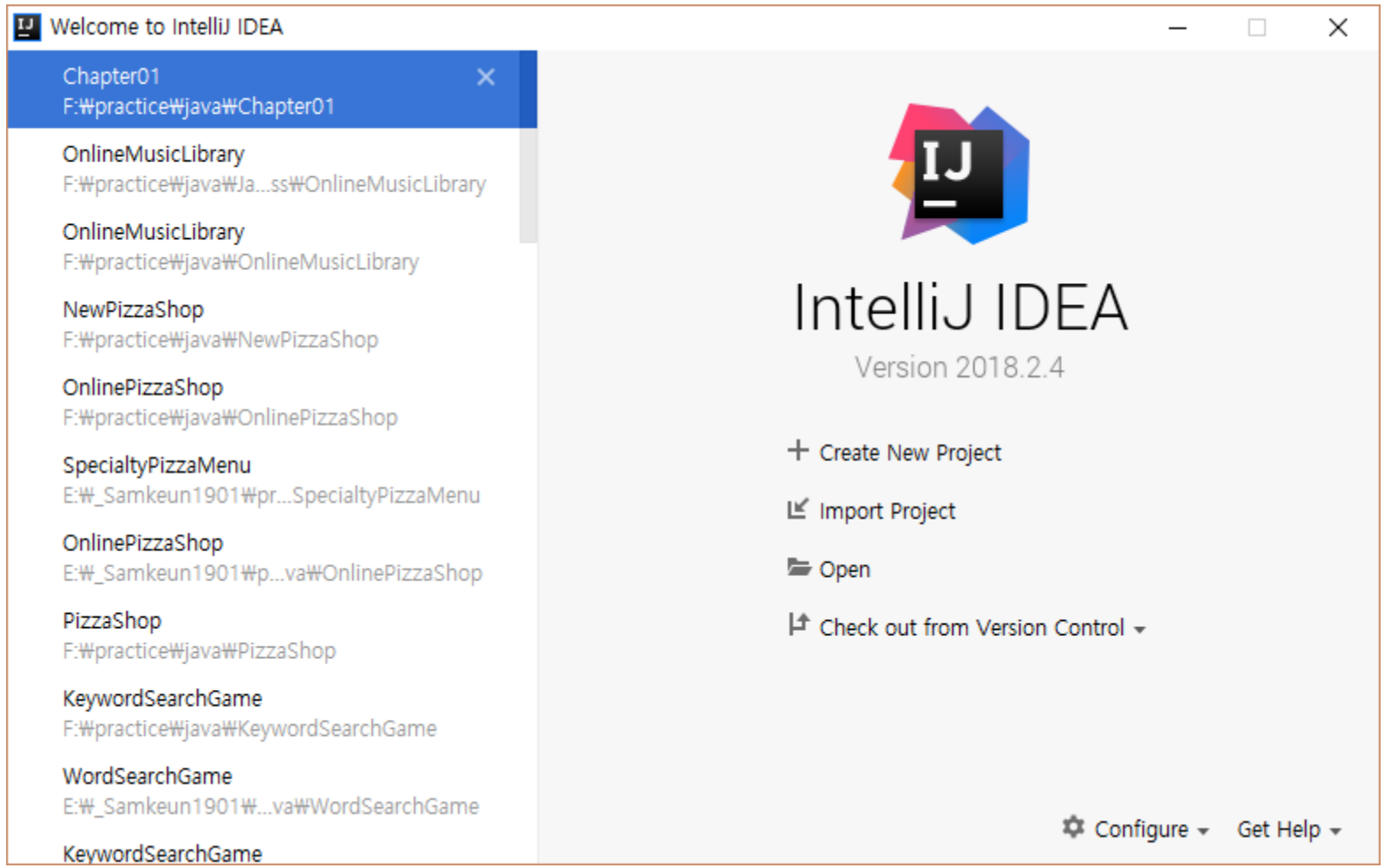
- ✓ **클래스** 들을 하나로 묶어놓은 것이다
- ✓ **클래스** 들의 이름 중복으로 발생하는 충돌을 막아준다
- ✓ **클래스** 들을 기능 별로 분류할 수 있어 필요한 클래스의 식별을 용이하게 한다

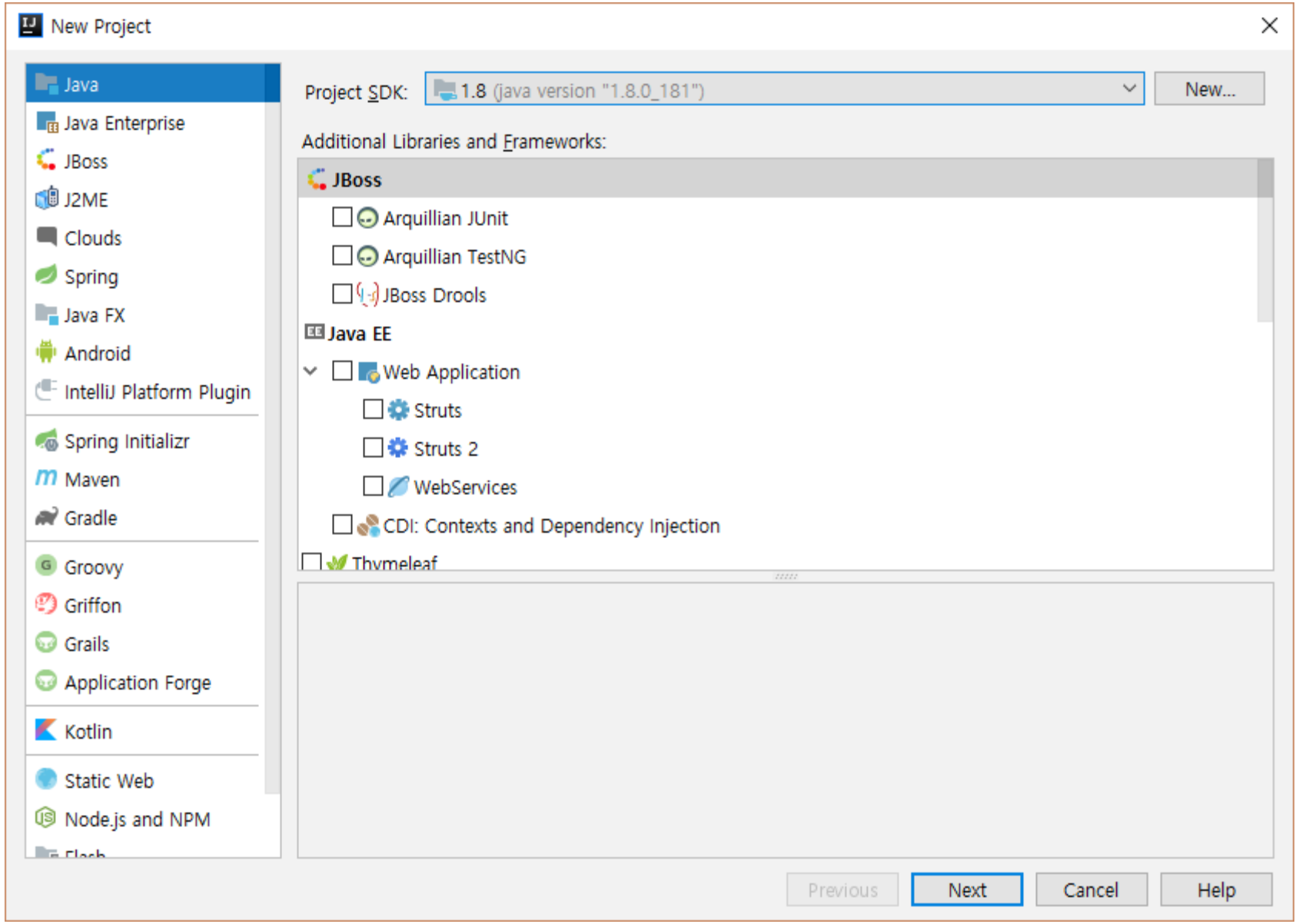
package 패키지 이름/패키지 경로; // ex) **package com.skimok**

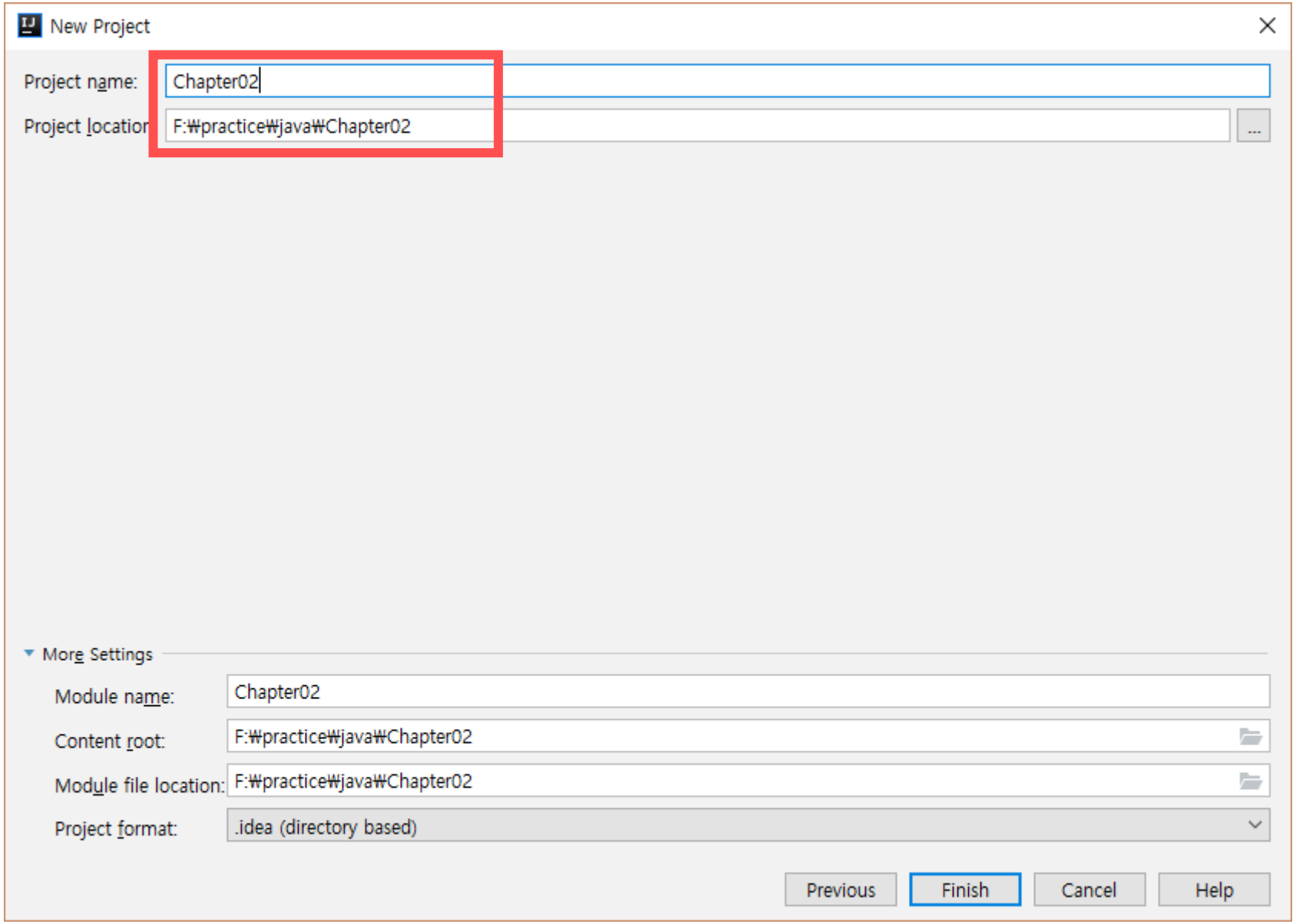
설명문서/과제 소스 코드 제출




Chapter01 [F:₩practice₩java₩Chapter01] - ...₩src₩ch01_2₩IfTest.java [Chapter01] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
New
Open...
Open URL...
Open Recent
Close Project
Settings... Ctrl+Alt+S
Project Structure... Ctrl+Alt+Shift+S
Other Settings
Import Settings...
Export Settings...
Settings Repository...
Sync Settings to JetBrains Account...
Export Project to Eclipse...
Export to Zip File...
Save All Ctrl+S
Synchronize Ctrl+Alt+Y
Invalidate Caches / Restart...
Export to HTML...
Print...
Add to Favorites
File Encoding
Remove BOM
Associate with File Type...
Line Separators
Make File Read-only
Power Save Mode
Exit
Loopy
Loopy.java
IfTest.java
1 package ch01_2;
2
3 public class IfTest {
4 }
5
dk1.8.0_181₩bin₩java.exe" ...
ode 0
Database
Maven Projects
Ant Build
6: TODO
Event Log
Close current project
5:1 CRLF UTF-8


Welcome to IntelliJ IDEA
Chapter01
F:₩practice₩java₩Chapter01
OnlineMusicLibrary
F:₩practice₩java₩Ja...ss₩OnlineMusicLibrary
OnlineMusicLibrary
F:₩practice₩java₩OnlineMusicLibrary
NewPizzaShop
F:₩practice₩java₩NewPizzaShop
OnlinePizzaShop
F:₩practice₩java₩OnlinePizzaShop
SpecialtyPizzaMenu
E:₩_Samkeun1901₩pr...SpecialtyPizzaMenu
OnlinePizzaShop
E:₩_Samkeun1901₩p...va₩OnlinePizzaShop
PizzaShop
F:₩practice₩java₩PizzaShop
KeywordSearchGame
F:₩practice₩java₩KeywordSearchGame
WordSearchGame
E:₩_Samkeun1901₩...va₩WordSearchGame
KeywordSearchGame
IntelliJ IDEA
Version 2018.2.4
Create New Project
Import Project
Open
Check out from Version Control
Configure
Get Help


New Project
Java
Java Enterprise
JBoss
J2ME
Clouds
Spring
Java FX
Android
IntelliJ Platform Plugin
Spring Initializr
Maven
Gradle
Groovy
Griffon
Grails
Application Forge
Kotlin
Static Web
Node.js and NPM
Project SDK:
1.8 (java version "1.8.0_181")
New...
Additional Libraries and Frameworks:
JBoss
Arquillian JUnit
Arquillian TestNG
JBoss Drools
Java EE
Web Application
Struts
Struts 2
WebServices
CDI: Contexts and Dependency Injection
Thymeleaf
Previous
Next
Cancel
Help


New Project
Project name:
Chapter02
Project location:
F:₩practice₩java₩Chapter02
...
More Settings
Module name:
Chapter02
Content root:
F:₩practice₩java₩Chapter02
Module file location:
F:₩practice₩java₩Chapter02
Project format:
.idea (directory based)
Previous
Finish
Cancel
Help