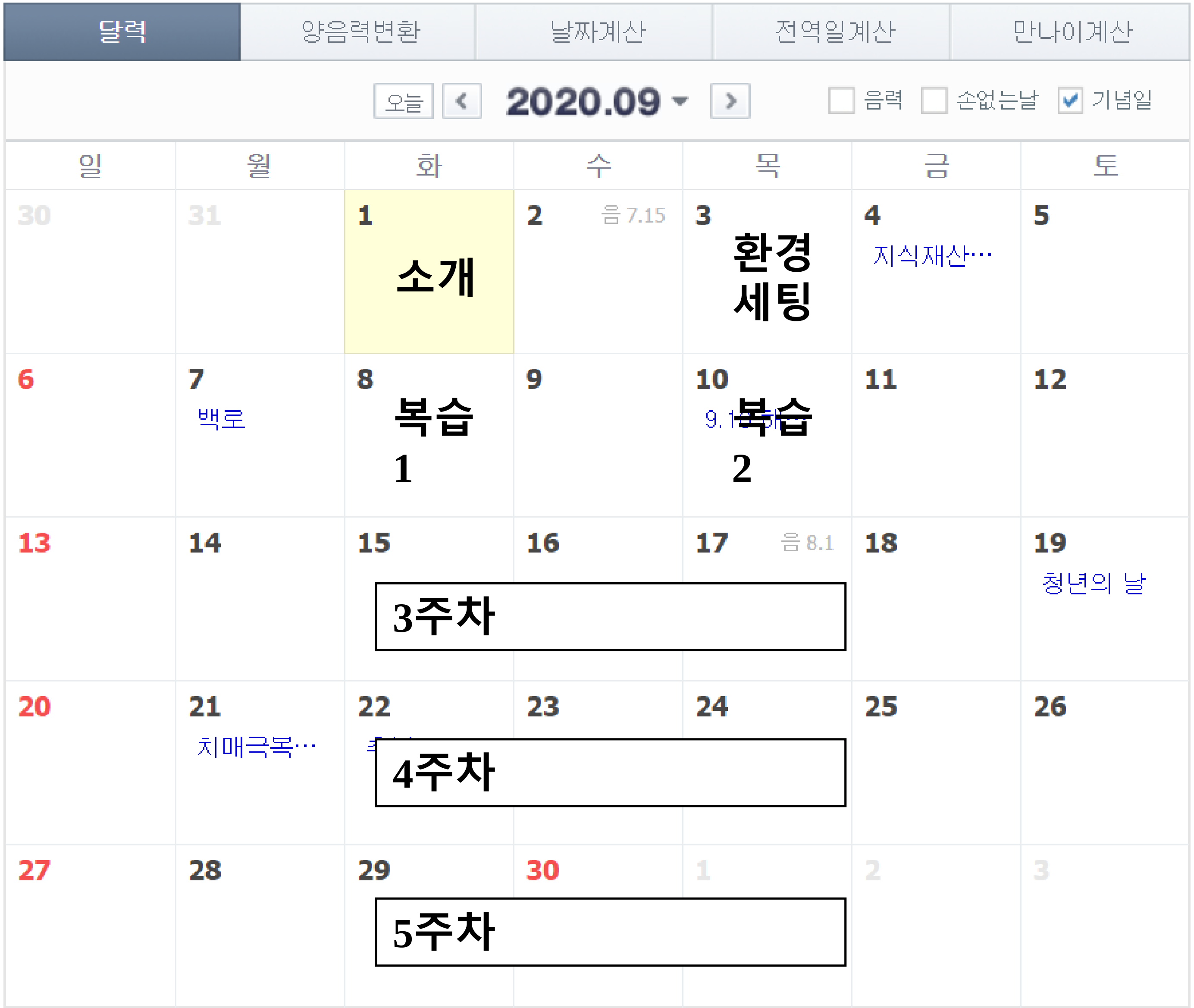
Data Analysis
(Course Overview and Preparation)

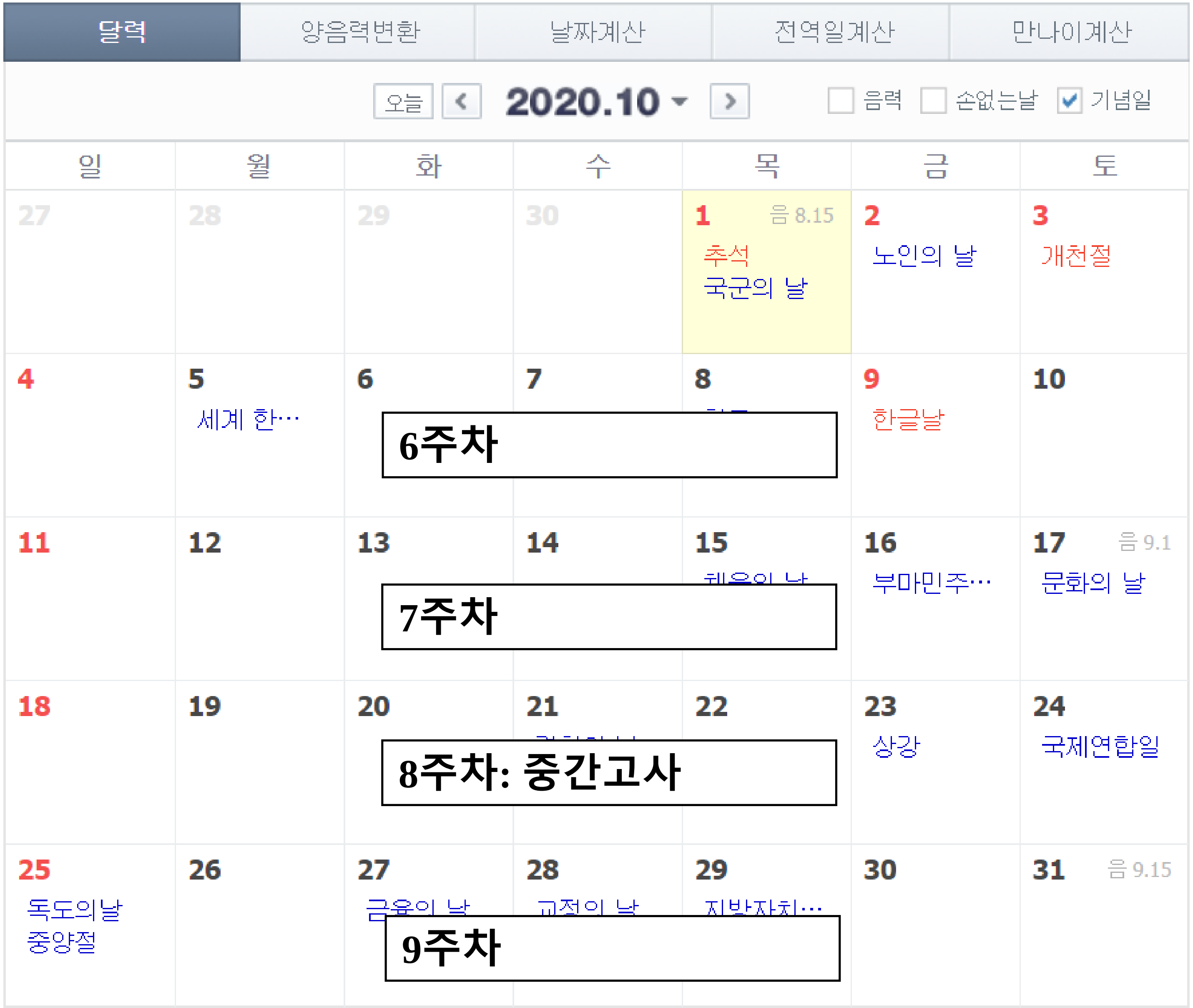
Fall, 2020

Calendar



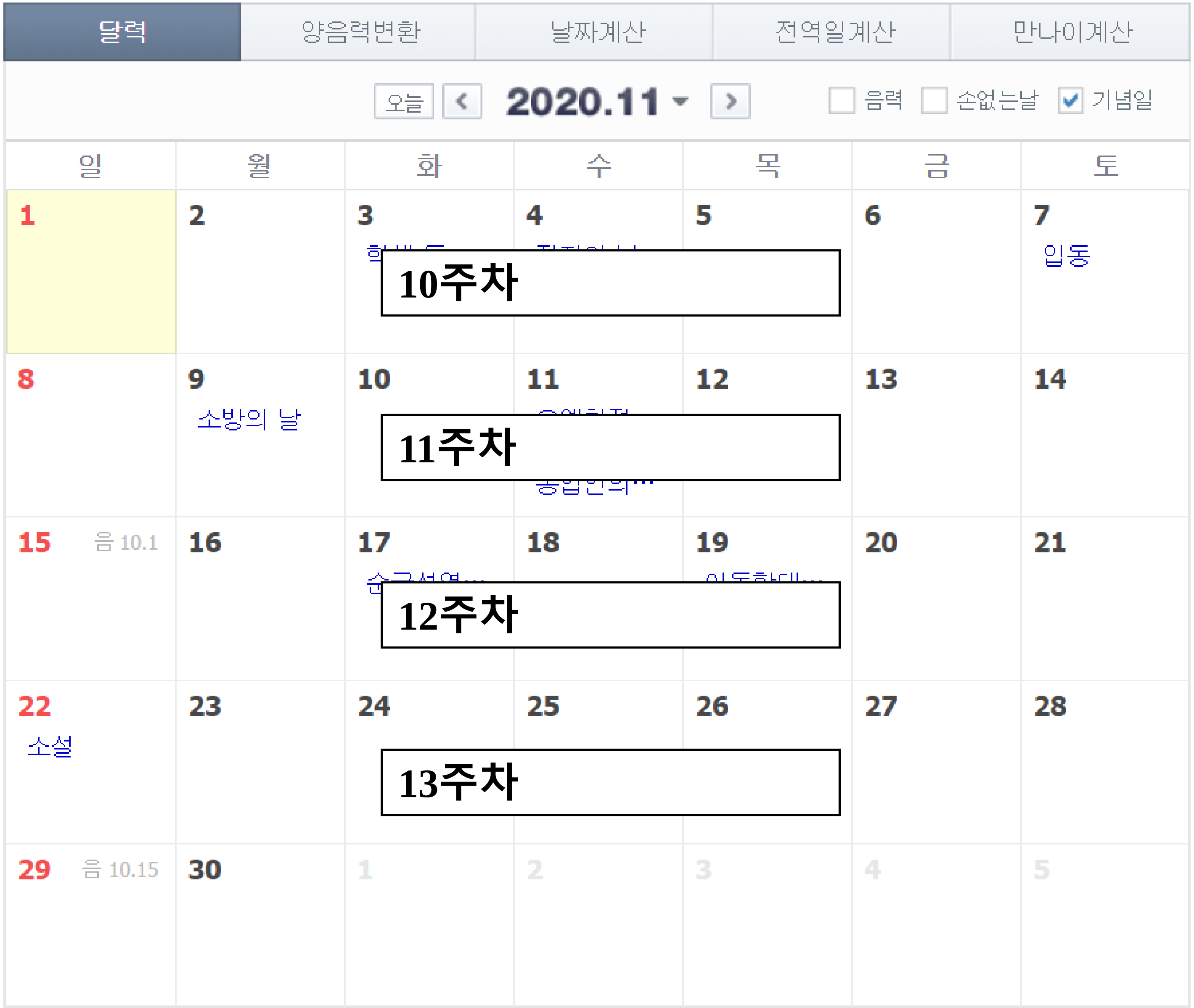
일	월	화	수	목	금	토
30	31	1 소개	2 음 7.15	3 환경 세팅	4 지식재산…	5
6	7 백로	8 복습 1	9	10 9. 1[illegible] 복습 2	11	12
13	14	15 3주차	16	17 음 8.1	18	19 청년의 날
20	21 치매극복…	22 [illegible] 4주차	23	24	25	26
27	28	29 5주차	30	1	2	3

Calendar



일	월	화	수	목	금	토
27	28	29	30	1 음 8.15 추석 국군의 날	2 노인의 날	3 개천절
4	5 세계 한…	6 6주차	7	8	9 한글날	10
11	12	13 7주차	14	15 체육의 날	16 부마민주…	17 음 9.1 문화의 날
18	19	20 8주차: 중간고사	21	22	23 상강	24 국제연합일
25 독도의날 중양절	26	27 금융의 날 9주차	28 교정의 날	29 지방자치…	30	31 음 9.15

Calendar



달력 | 양음력변환 | 날짜계산 | 전역일계산 | 만나이계산

오늘 ‹ 2020.11 ▾ › □ 음력 □ 손없는날 ☑ 기념일

일	월	화	수	목	금	토
1	2	3 학생독…	4 점자의 날	5	6	7 입동
8	9 소방의 날	10	11 유엔참전… 농업인의…	12	13	14
15 음 10.1	16	17 순국선열…	18	19 아동학대…	20	21
22 소설	23	24	25	26	27	28
29 음 10.15	30	1	2	3	4	5

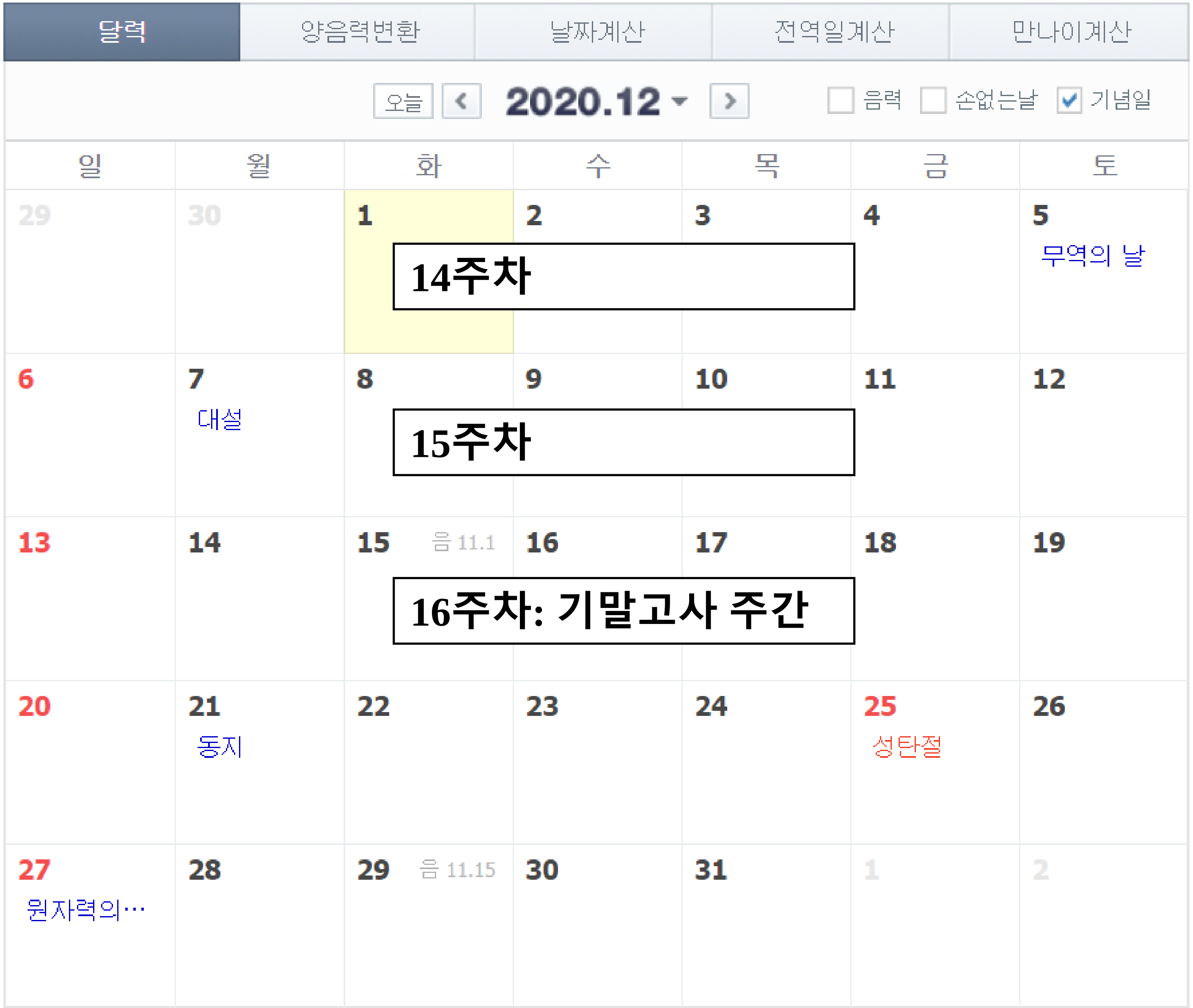
10주차

11주차

12주차

13주차

Calendar



달력 | 양음력변환 | 날짜계산 | 전역일계산 | 만나이계산

오늘 < 2020.12 ▾ > □ 음력 □ 손없는날 ☑ 기념일

일	월	화	수	목	금	토
29	30	1	2	3	4	5 무역의 날
6	7 대설	8	9	10	11	12
13	14	15 음 11.1	16	17	18	19
20	21 동지	22	23	24	25 성탄절	26
27 원자력의…	28	29 음 11.15	30	31	1	2

14주차

15주차

16주차: 기말고사 주간

Preparation

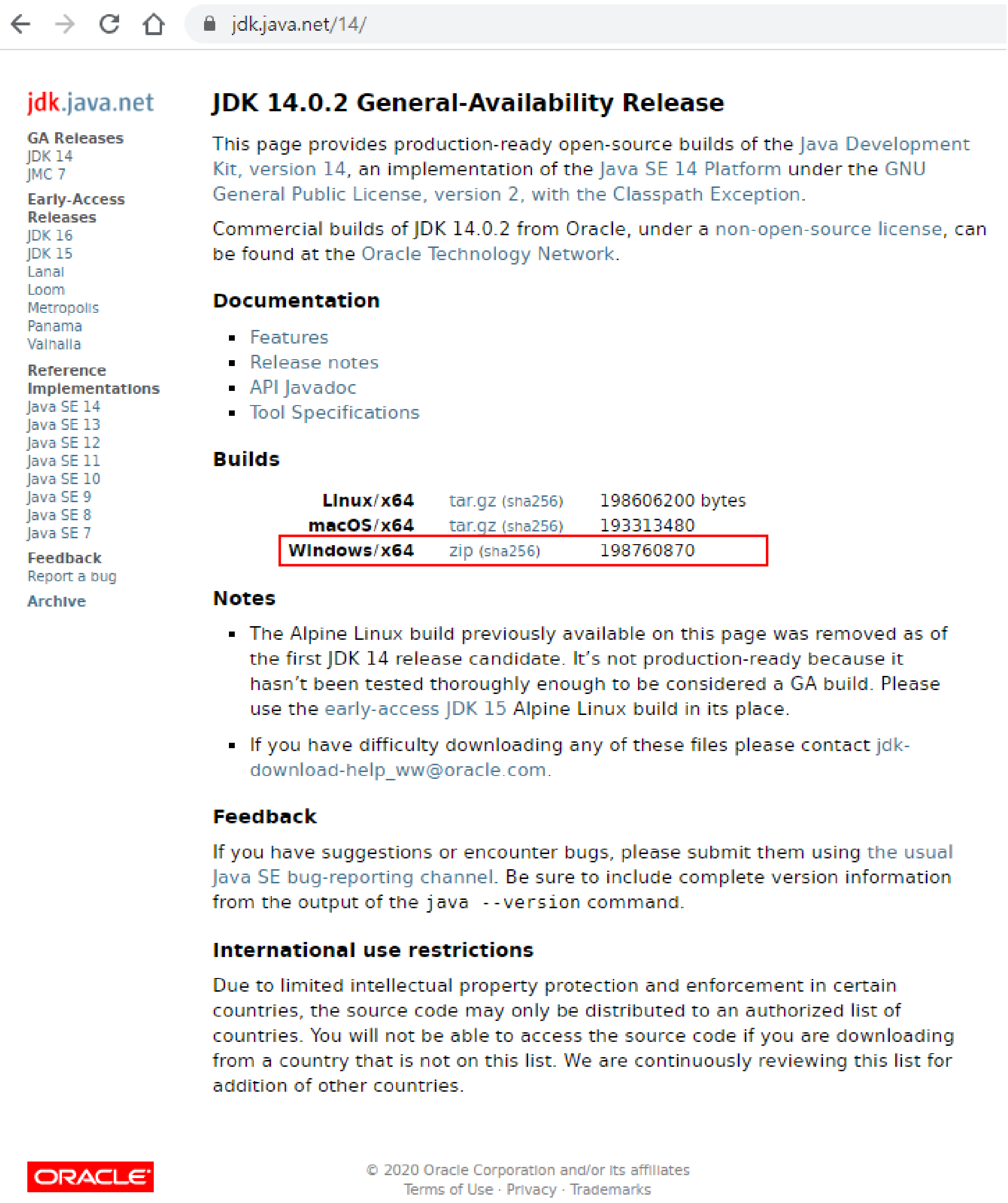
- **Java**

Preparation

- Open JDK 14
- Eclipse IDE
  - Integrated Development Environment (IDE)
  - provides comprehensive facilities to computer programmers
    - Source code editor
    - Build automation tools
    - Debugger
    - …
- Maven
  - Software project management and comprehension tool
  - Manages a project’s build, reporting and documentation

Install Open JDK 14

- https://jdk.java.net/14/
- Windows x64



jdk.java.net

GA Releases
JDK 14
JMC 7

Early-Access Releases
JDK 16
JDK 15
Lanai
Loom
Metropolis
Panama
Valhalla

Reference Implementations
Java SE 14
Java SE 13
Java SE 12
Java SE 11
Java SE 10
Java SE 9
Java SE 8
Java SE 7

Feedback
Report a bug

Archive

JDK 14.0.2 General-Availability Release

This page provides production-ready open-source builds of the Java Development Kit, version 14, an implementation of the Java SE 14 Platform under the GNU General Public License, version 2, with the Classpath Exception.

Commercial builds of JDK 14.0.2 from Oracle, under a non-open-source license, can be found at the Oracle Technology Network.

Documentation

- Features
- Release notes
- API Javadoc
- Tool Specifications

Builds

Linux/x64	tar.gz (sha256)	198606200 bytes
macOS/x64	tar.gz (sha256)	193313480
Windows/x64	zip (sha256)	198760870

Notes

- The Alpine Linux build previously available on this page was removed as of the first JDK 14 release candidate. It's not production-ready because it hasn't been tested thoroughly enough to be considered a GA build. Please use the early-access JDK 15 Alpine Linux build in its place.
- If you have difficulty downloading any of these files please contact jdk-download-help_ww@oracle.com.

Feedback

If you have suggestions or encounter bugs, please submit them using the usual Java SE bug-reporting channel. Be sure to include complete version information from the output of the `java --version` command.

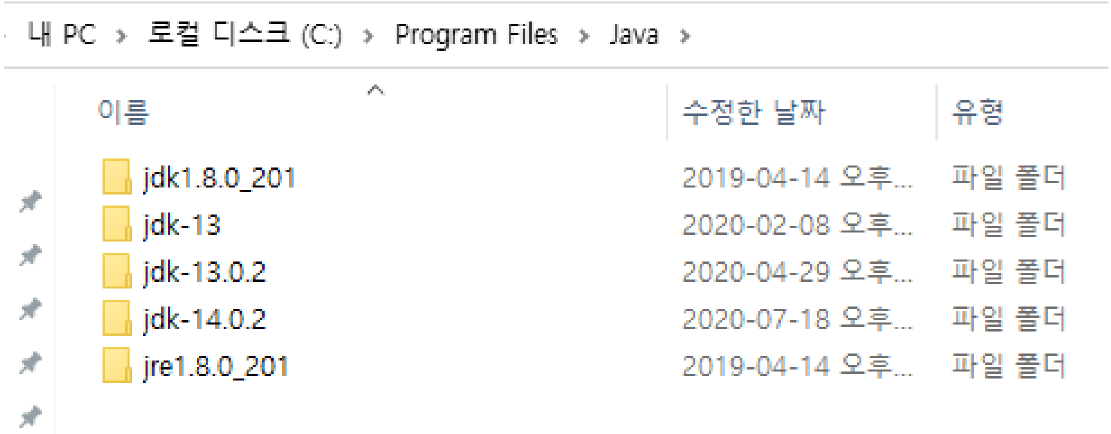
International use restrictions

Due to limited intellectual property protection and enforcement in certain countries, the source code may only be distributed to an authorized list of countries. You will not be able to access the source code if you are downloading from a country that is not on this list. We are continuously reviewing this list for addition of other countries.

ORACLE

Install Open JDK 14

- Unzip



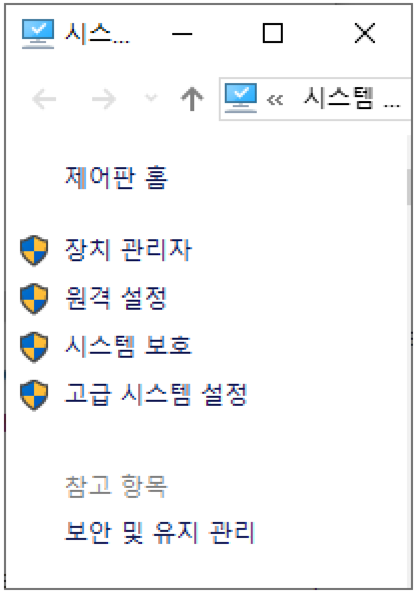
내 PC › 로컬 디스크 (C:) › Program Files › Java ›

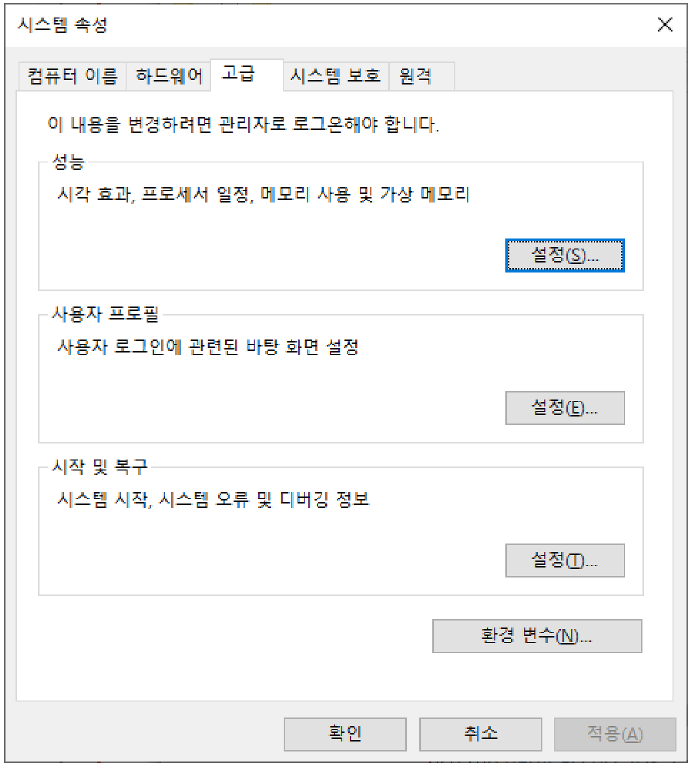
이름	수정한 날짜	유형
jdk1.8.0_201	2019-04-14 오후...	파일 폴더
jdk-13	2020-02-08 오후...	파일 폴더
jdk-13.0.2	2020-04-29 오후...	파일 폴더
jdk-14.0.2	2020-07-18 오후...	파일 폴더
jre1.8.0_201	2019-04-14 오후...	파일 폴더

- Environment Setting
  - 내 PC → 속성 → 고급 시스템 설정 →

Install Open JDK 14

- Environment Setting
  - 내 PC → 속성 → 고급 시스템 설정 → 환경변수 → 시스템 변수 → Path 더블 클릭 → 새로 만들기 → bin 폴더 추가





이 내용을 변경하려면 관리자로 로그온해야 합니다.

성능

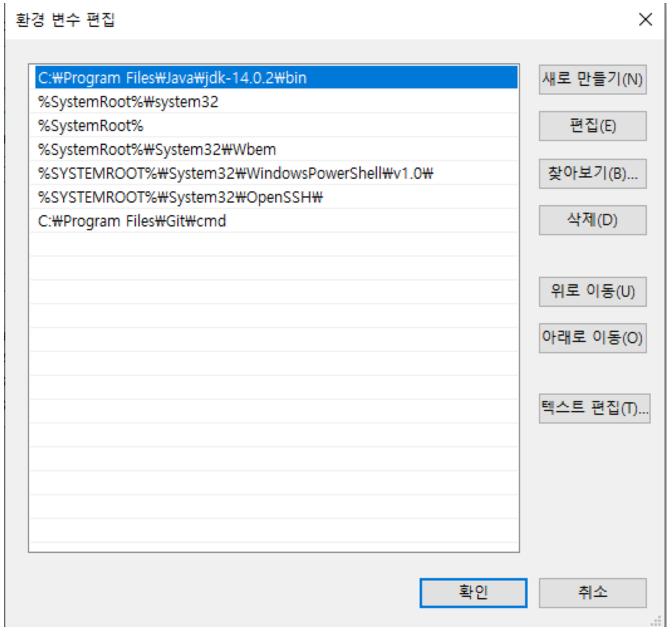
시각 효과, 프로세서 일정, 메모리 사용 및 가상 메모리

사용자 프로필

사용자 로그인에 관련된 바탕 화면 설정

시작 및 복구

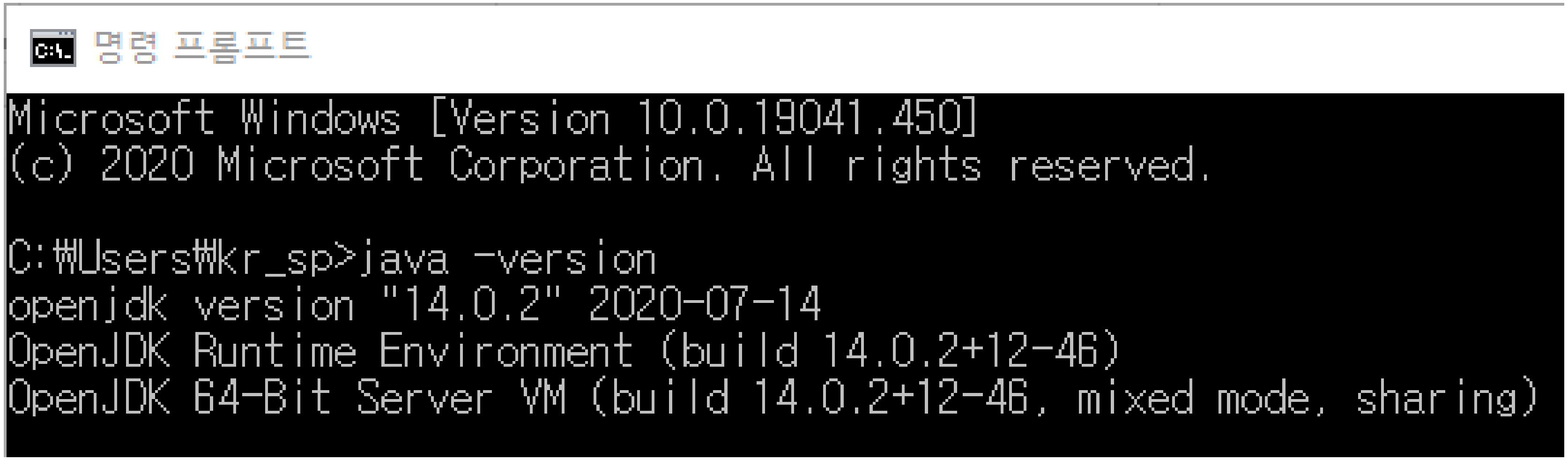
시스템 시작, 시스템 오류 및 디버깅 정보



```
C:₩Program Files₩Java₩jdk-14.0.2₩bin
%SystemRoot%₩system32
%SystemRoot%
%SystemRoot%₩System32₩Wbem
%SYSTEMROOT%₩System32₩WindowsPowerShell₩v1.0₩
%SYSTEMROOT%₩System32₩OpenSSH₩
C:₩Program Files₩Git₩cmd
```

Install Open JDK 14

- Check
  - CMD
    - Window key → cmd → enter

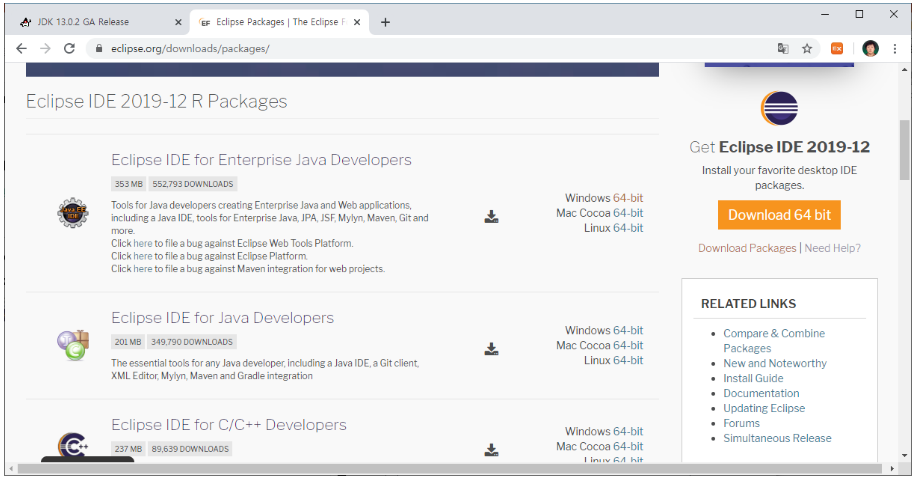


```
Microsoft Windows [Version 10.0.19041.450]
(c) 2020 Microsoft Corporation. All rights reserved.

C:₩Users₩kr_sp>java -version
openjdk version "14.0.2" 2020-07-14
OpenJDK Runtime Environment (build 14.0.2+12-46)
OpenJDK 64-Bit Server VM (build 14.0.2+12-46, mixed mode, sharing)
```

Install Eclipse IDE

- Download recent Eclipse IDE for Java EE Developers
- http://www.eclipse.org/downloads/packages/
- Windows 64-bit
  - Eclipse IDE for Enterprise Java Developers
- Unzip the file to any folder (e.g., E:\eclipse)
- Execute E:\eclipse\eclipse.exe


JDK 13.0.2 GA Release
Eclipse Packages | The Eclipse F
eclipse.org/downloads/packages/

Eclipse IDE 2019-12 R Packages

Eclipse IDE for Enterprise Java Developers

353 MB 552,793 DOWNLOADS

Tools for Java developers creating Enterprise Java and Web applications, including a Java IDE, tools for Enterprise Java, JPA, JSF, Mylyn, Maven, Git and more.
Click here to file a bug against Eclipse Web Tools Platform.
Click here to file a bug against Eclipse Platform.
Click here to file a bug against Maven integration for web projects.

Windows 64-bit
Mac Cocoa 64-bit
Linux 64-bit

Eclipse IDE for Java Developers

201 MB 349,790 DOWNLOADS

The essential tools for any Java developer, including a Java IDE, a Git client, XML Editor, Mylyn, Maven and Gradle integration

Windows 64-bit
Mac Cocoa 64-bit
Linux 64-bit

Eclipse IDE for C/C++ Developers

237 MB 89,639 DOWNLOADS

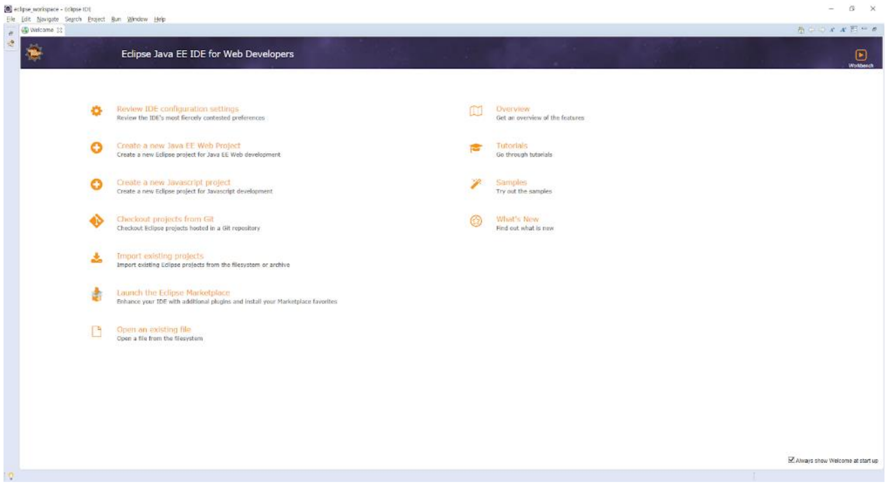
Windows 64-bit
Mac Cocoa 64-bit
Linux 64-bit

Get **Eclipse IDE 2019-12**

Install your favorite desktop IDE packages.

RELATED LINKS

- Compare & Combine Packages
- New and Noteworthy
- Install Guide
- Documentation
- Updating Eclipse
- Forums
- Simultaneous Release


eclipse_workspace - Eclipse IDE
File Edit Navigate Search Project Run Window Help
Welcome
Workbench

Eclipse Java EE IDE for Web Developers

Review IDE configuration settings
Review the IDE's most fiercely contested preferences

Create a new Java EE Web Project
Create a new Eclipse project for Java EE Web development

Create a new Javascript project
Create a new Eclipse project for Javascript development

Checkout projects from Git
Checkout Eclipse projects hosted in a Git repository

Import existing projects
Import existing Eclipse projects from the filesystem or archive

Launch the Eclipse Marketplace
Enhance your IDE with additional plugins and install your Marketplace favorites

Open an existing file
Open a file from the filesystem

Overview
Get an overview of the features

Tutorials
Go through tutorials

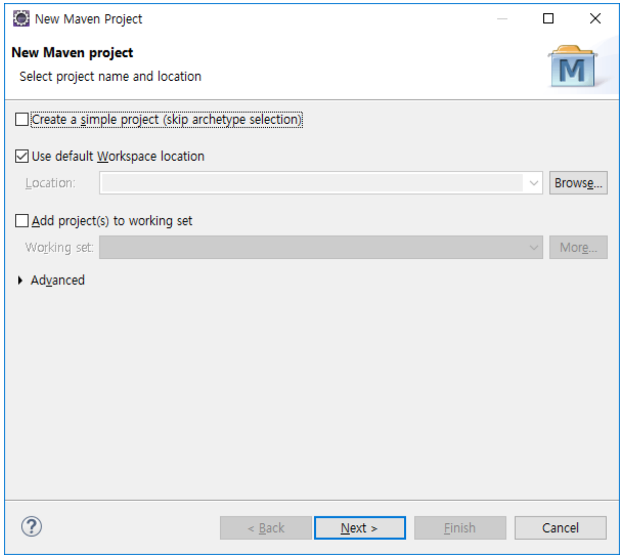
Samples
Try out the samples

What's New
Find out what is new

☑ Always show Welcome at start up

First Application

- File → New → Maven Project
- New Maven project
  - Maven-archetype-quickstart
- Group id
  - kr.ac.sejong.data_analysis
- Artifact id
  - helloworld
- Version
  - 0.0.1
- Package → kr.ac.sejong.data_analysis.helloworld



New Maven Project

New Maven project

Select project name and location

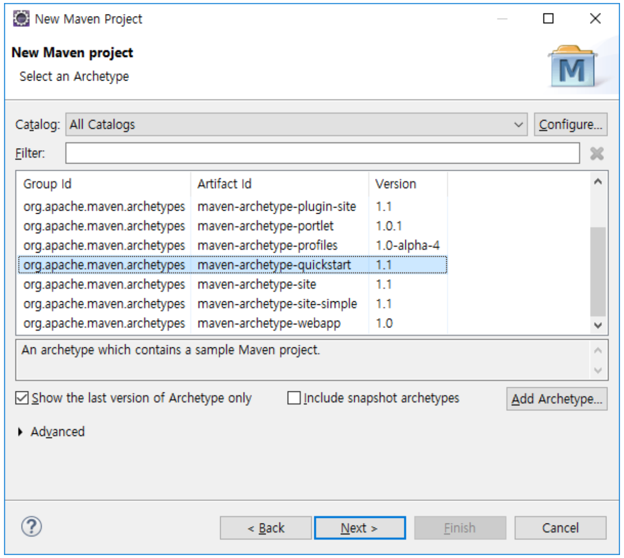
□ Create a simple project (skip archetype selection)

☑ Use default Workspace location

Location:

□ Add project(s) to working set

Working set:



New Maven Project

New Maven project

Select an Archetype

Catalog: All Catalogs

Filter:

Group Id	Artifact Id	Version
org.apache.maven.archetypes	maven-archetype-plugin-site	1.1
org.apache.maven.archetypes	maven-archetype-portlet	1.0.1
org.apache.maven.archetypes	maven-archetype-profiles	1.0-alpha-4
org.apache.maven.archetypes	maven-archetype-quickstart	1.1
org.apache.maven.archetypes	maven-archetype-site	1.1
org.apache.maven.archetypes	maven-archetype-site-simple	1.1
org.apache.maven.archetypes	maven-archetype-webapp	1.0

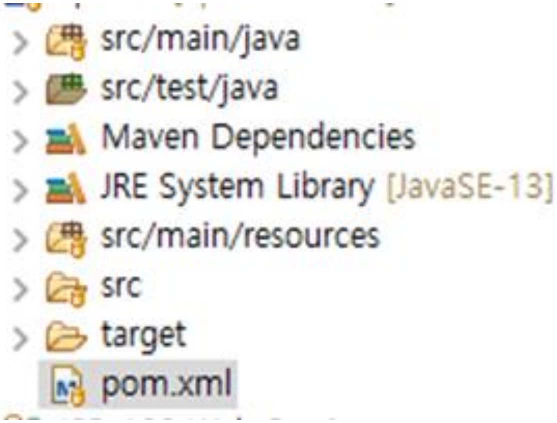
An archetype which contains a sample Maven project.

☑ Show the last version of Archetype only

□ Include snapshot archetypes

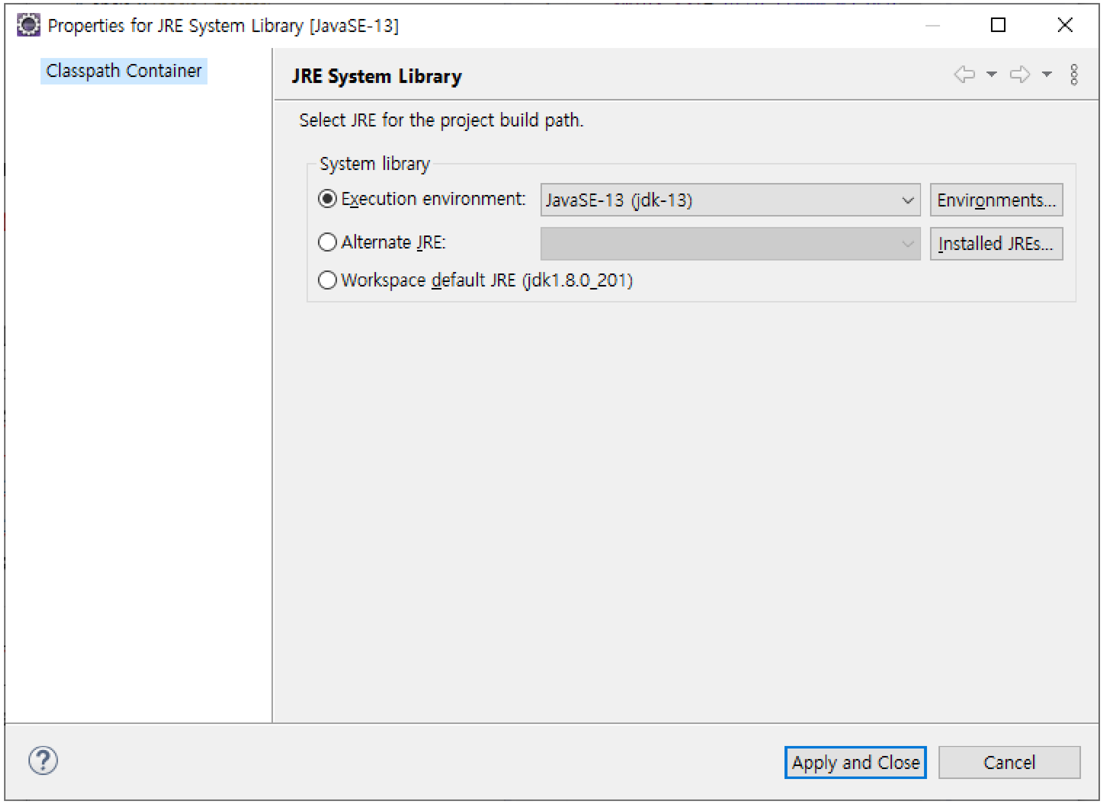
First Application

- Using Oracle Java 14
  - Update pom.xml
    - <properties>
      - <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
      - <maven.compiler.source>14</maven.compiler.source>
      - <maven.compiler.target>14</maven.compiler.target>
    - </properties>
  - Ctrl + space and see the magic
  - Right click for helloworld → Maven → Update Project
  - See JRE System Library changed
  - If jre is used, change it to jdk



First Application

- Right click of JRE System Library → Properties
- Installed JREs
- Add
- Standard VM
- Set JRE home as jdk folder
- Finish


Properties for JRE System Library [JavaSE-13]

JRE System Library

Select JRE for the project build path.

System library

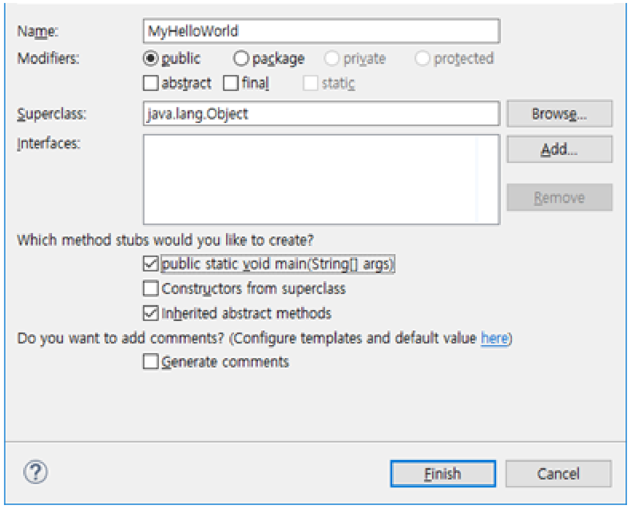
● Execution environment: JavaSE-13 (jdk-13)

○ Alternate JRE:

○ Workspace default JRE (jdk1.8.0_201)

First Application

- Right click of package kr.ac.sejong.data_analysis.helloworld → New → Class
- Create MyHelloWorld with public static void main
- In the main method
  - String message = “My”;
  - message += “ Hello”;
  - message += “ World!”;
  - System.out.println(“My Hello World!”);   // Feel IDE   ctrl+space



Name: MyHelloWorld

Modifiers: ● public ○ package ○ private ○ protected

□ abstract □ final □ static

Superclass: java.lang.Object

Interfaces:

Which method stubs would you like to create?

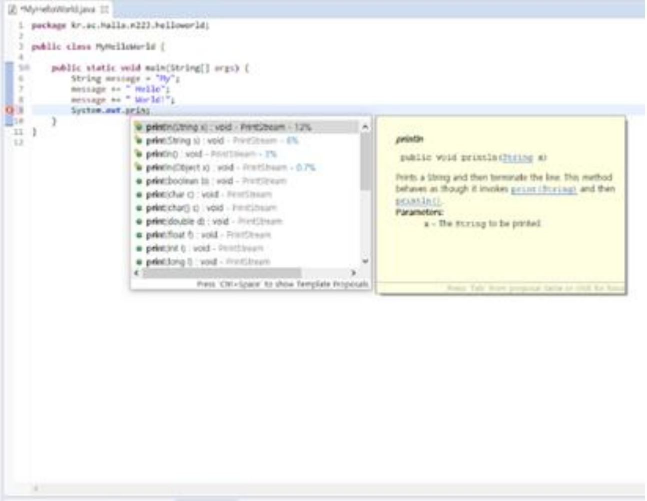
☑ public static void main(String[] args)

□ Constructors from superclass

☑ Inherited abstract methods

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

□ Generate comments



```
package kr.ac.halla.m223.helloworld;

public class MyHelloWorld {

    public static void main(String[] args) {
        String message = "My";
        message += " Hello";
        message += " World!";
        System.out.prin;
    }
}
```

```
println(String x) : void - PrintStream - 12%
print(String s) : void - PrintStream - 6%
println() : void - PrintStream - 3%
println(Object x) : void - PrintStream - 0.7%
print(boolean b) : void - PrintStream
print(char c) : void - PrintStream
print(char[] s) : void - PrintStream
print(double d) : void - PrintStream
print(float f) : void - PrintStream
print(int i) : void - PrintStream
print(long l) : void - PrintStream
Press 'Ctrl+Space' to show Template Proposals
```

println

public void println(String x)

Prints a String and then terminate the line. This method behaves as though it invokes print(String) and then println().

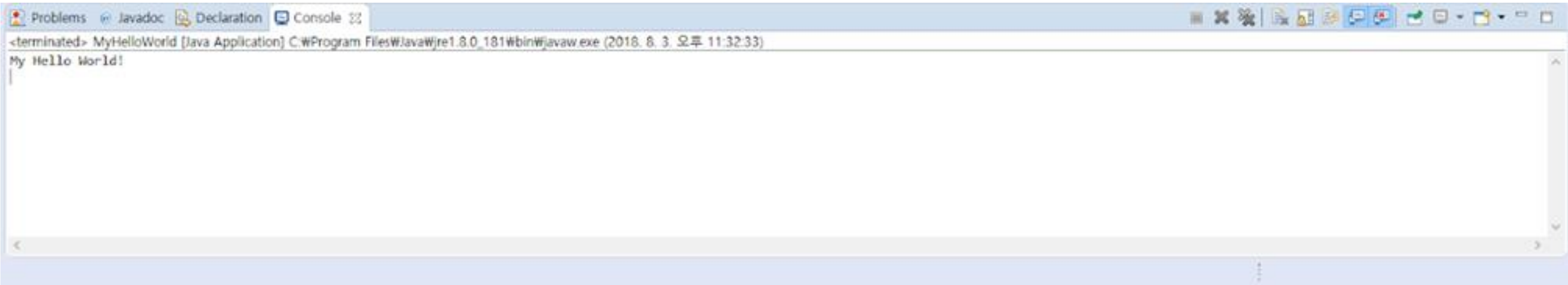
Parameters:

x - The String to be printed

- HINT
- DOCUMENTATION

First Application

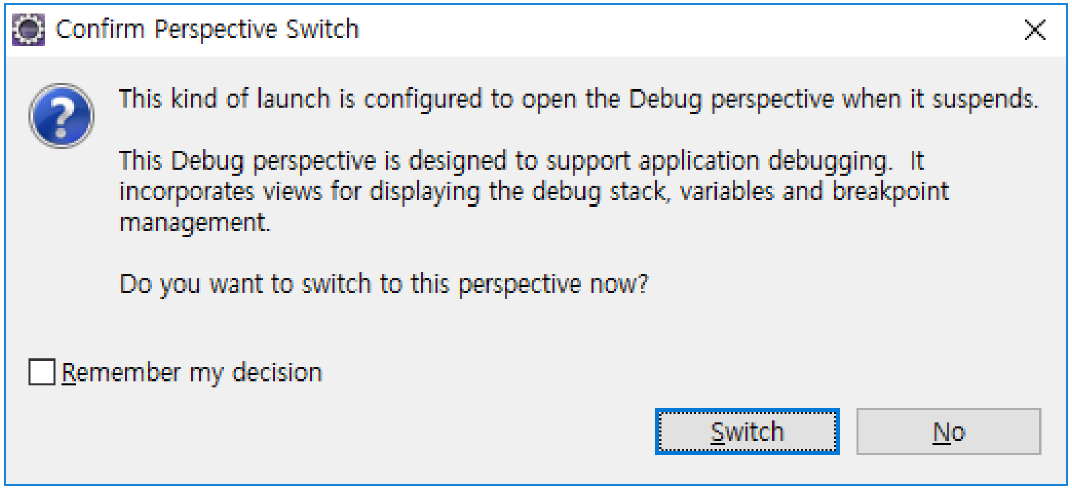
- Run MyHelloWorld
    - Right click for MyHelloWorld.java → Run As → Java Application



```
<terminated> MyHelloWorld [Java Application] C:₩Program Files₩Java₩jre1.8.0_181₩bin₩javaw.exe (2018. 8. 3. 오후 11:32:33)
My Hello World!
```

First Application

- Debug My Hello World
  - Double click at Line 7 (i.e., message += “ Hello”;)
- Right click for MyHelloWorld → Debug As → Java Application



This kind of launch is configured to open the Debug perspective when it suspends.

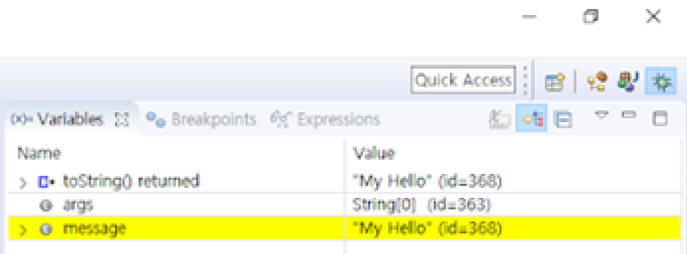
This Debug perspective is designed to support application debugging. It incorporates views for displaying the debug stack, variables and breakpoint management.

Do you want to switch to this perspective now?

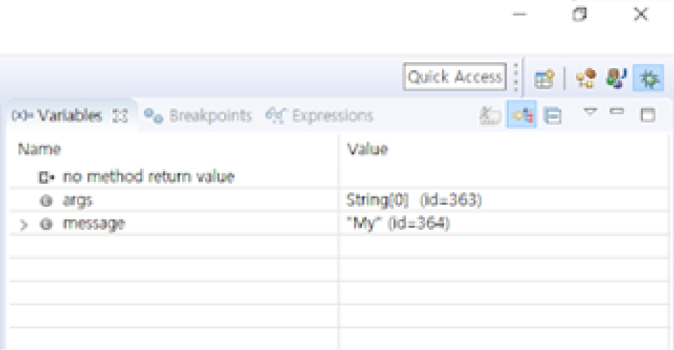
□ Remember my decision

First Application

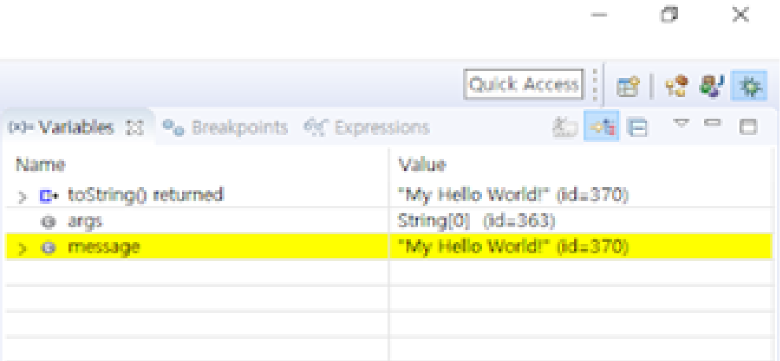
- Step Over (F6)
- Step In (F5)



Name	Value
toString() returned	"My Hello" (id=368)
args	String[0] (id=363)
message	"My Hello" (id=368)

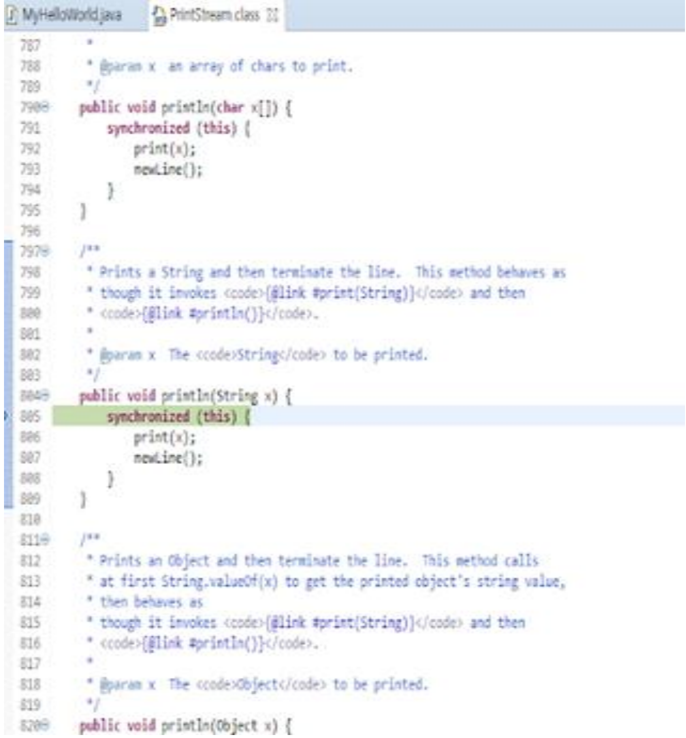


Name	Value
no method return value	
args	String[0] (id=363)
message	"My" (id=364)



Name	Value
toString() returned	"My Hello World!" (id=370)
args	String[0] (id=363)
message	"My Hello World!" (id=370)

- Step into println method!



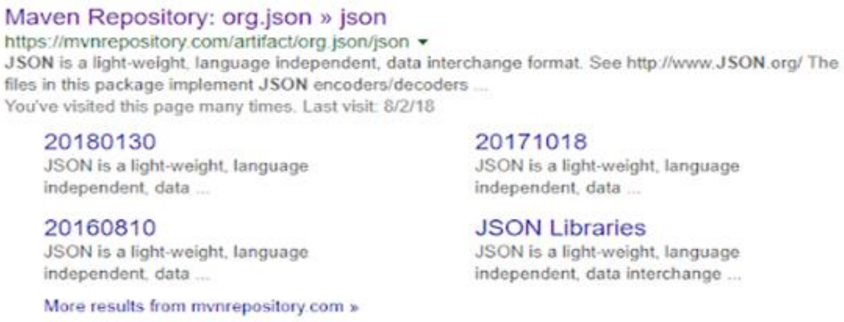
```
     *
     * @param x  an array of chars to print.
     */
    public void println(char x[]) {
        synchronized (this) {
            print(x);
            newLine();
        }
    }

    /**
     * Prints a String and then terminate the line.  This method behaves as
     * though it invokes <code>{@link #print(String)}</code> and then
     * <code>{@link #println()}</code>.
     *
     * @param x  The <code>String</code> to be printed.
     */
    public void println(String x) {
        synchronized (this) {
            print(x);
            newLine();
        }
    }

    /**
     * Prints an Object and then terminate the line.  This method calls
     * at first String.valueOf(x) to get the printed object's string value,
     * then behaves as
     * though it invokes <code>{@link #print(String)}</code> and then
     * <code>{@link #println()}</code>.
     *
     * @param x  The <code>Object</code> to be printed.
     */
    public void println(Object x) {
```

Second Application

- Can use external library such as org.json.json
- Originally
  - You need to download json.jar file
  - Add it to your build path
- With Maven
  - Search google “maven json”
  - Copy dependency and add it to pom.xml



Maven Repository: org.json » json

https://mvnrepository.com/artifact/org.json/json ▾

JSON is a light-weight, language independent, data interchange format. See http://www.**JSON**.org/ The files in this package implement **JSON** encoders/decoders ...

You've visited this page many times. Last visit: 8/2/18

20180130

JSON is a light-weight, language independent, data ...

20171018

JSON is a light-weight, language independent, data ...

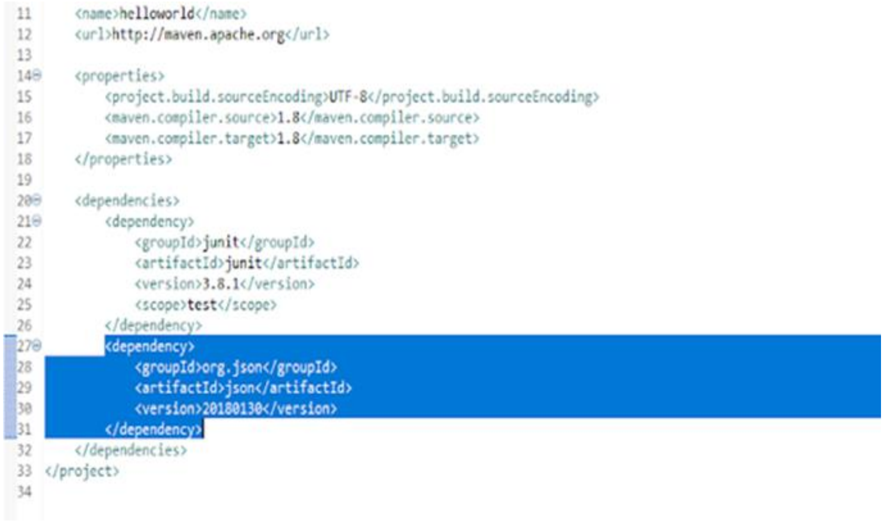
20160810

JSON is a light-weight, language independent, data ...

JSON Libraries

JSON is a light-weight, language independent, data interchange ...

More results from mvnrepository.com »



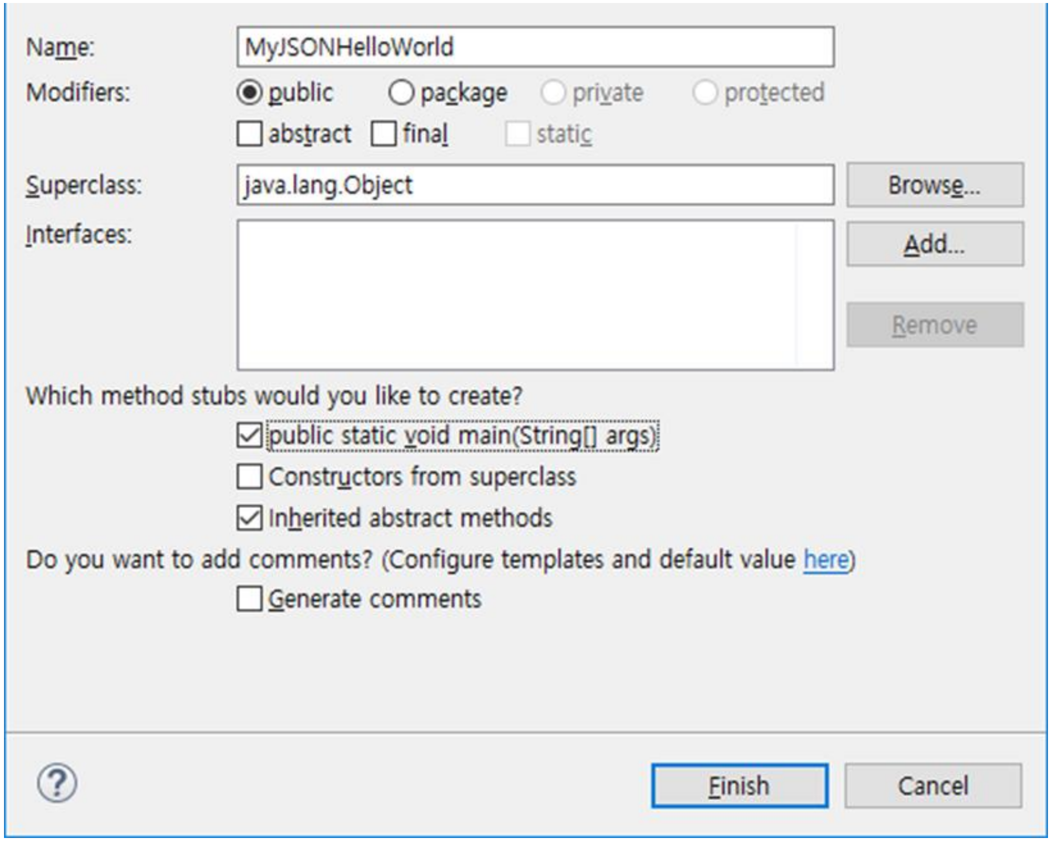
```
    <name>helloworld</name>
    <url>http://maven.apache.org</url>

    <properties>
        <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
        <maven.compiler.source>1.8</maven.compiler.source>
        <maven.compiler.target>1.8</maven.compiler.target>
    </properties>

    <dependencies>
        <dependency>
            <groupId>junit</groupId>
            <artifactId>junit</artifactId>
            <version>3.8.1</version>
            <scope>test</scope>
        </dependency>
        <dependency>
            <groupId>org.json</groupId>
            <artifactId>json</artifactId>
            <version>20180130</version>
        </dependency>
    </dependencies>
</project>
```

Second Application

- Update dependency
  - Maven → Update Project
  - Run → Maven Clean
  - Run → Maven Install
    - Will install org.json.json!
- Create MyJSONHelloWorld
- with public static void main



Name: MyJSONHelloWorld

Modifiers: ● public ○ package ○ private ○ protected

□ abstract □ final □ static

Superclass: java.lang.Object

Interfaces:

Which method stubs would you like to create?

☑ public static void main(String[] args)

□ Constructors from superclass

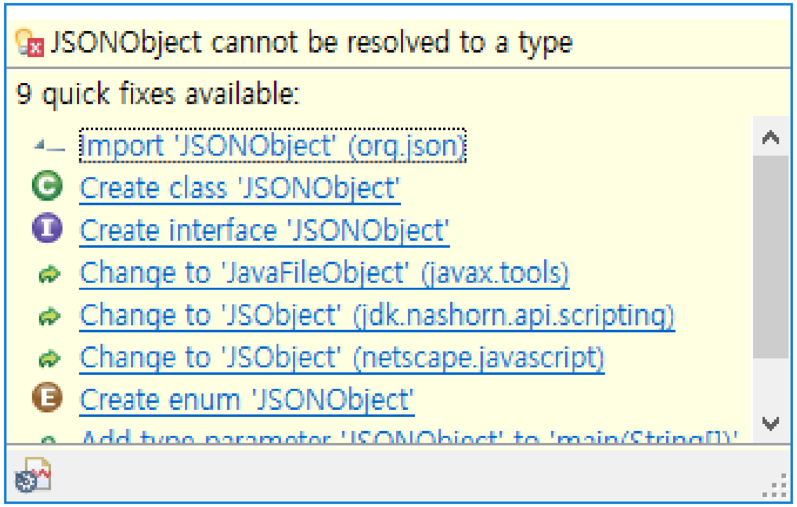
☑ Inherited abstract methods

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

□ Generate comments

Second Application

- It is not a JSON course
- Just past the following code snippet into main method
  - JSONObject jsonHello = new JSONObject();
  - jsonHello.put(“message”, “My JSON Hello World!”);
  - System.out.println(jsonHello);
- You will see error because JSONObject is not imported
- Resolve the error with IDE
  - Hover over JSONObject
  - Click import ‘JSONObject’ as shown in →



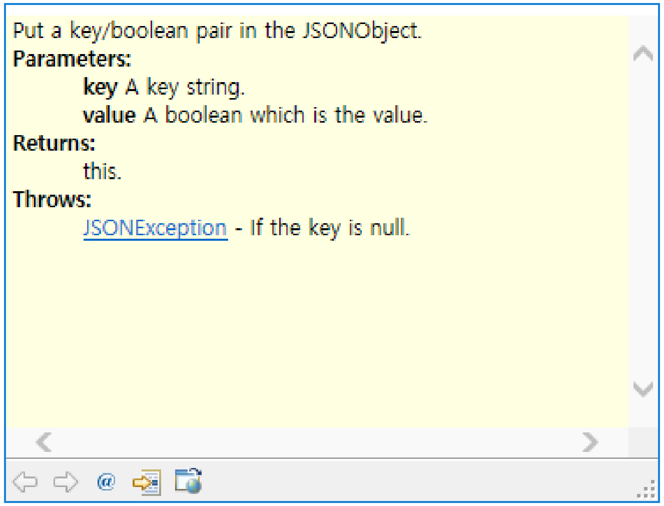
JSONObject cannot be resolved to a type

9 quick fixes available:

- Import 'JSONObject' (org.json)
- Create class 'JSONObject'
- Create interface 'JSONObject'
- Change to 'JavaFileObject' (javax.tools)
- Change to 'JSObject' (jdk.nashorn.api.scripting)
- Change to 'JSObject' (netscape.javascript)
- Create enum 'JSONObject'
- Add type parameter 'JSONObject' to 'main(String[])'

Second Application

- Maven → Download Javadoc
  - You can read their documentation
- Maven → Download Sources
  - You can step into 3rd party libraries



Put a key/boolean pair in the JSONObject.

Parameters:

key A key string.

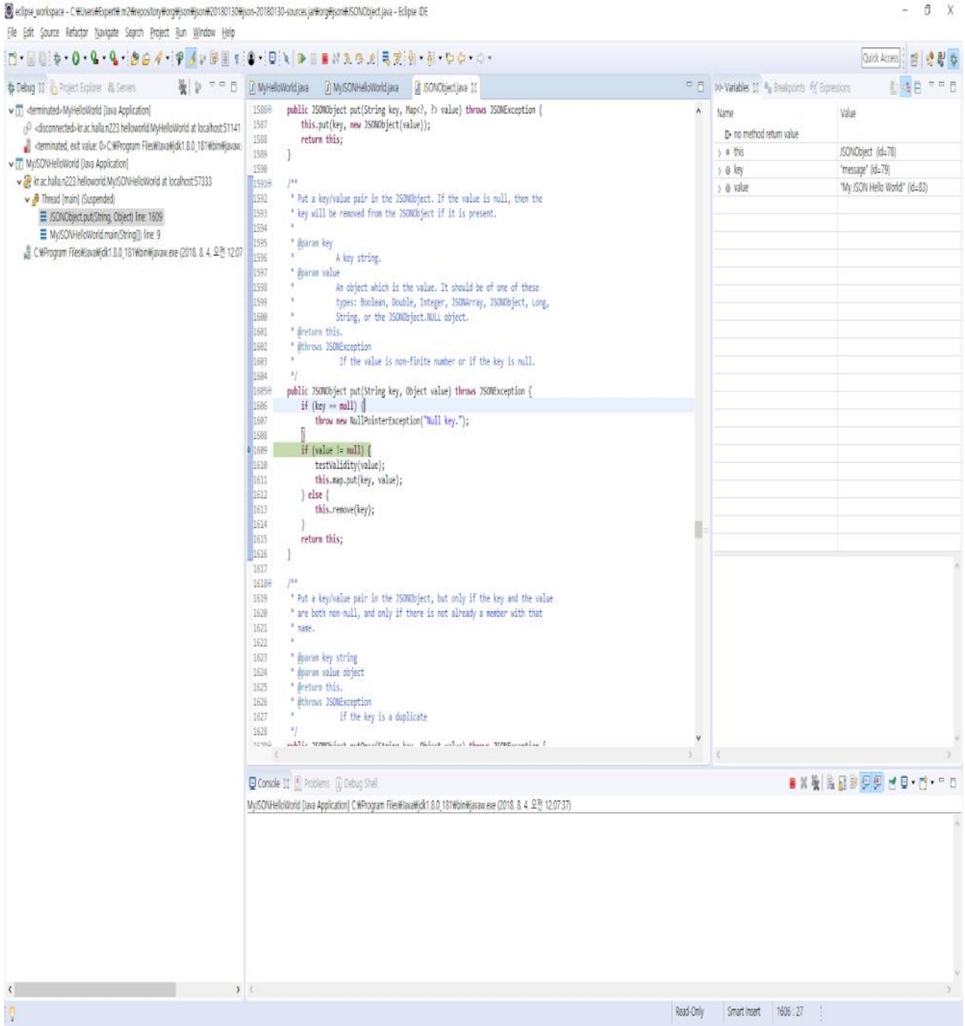
value A boolean which is the value.

Returns:

this.

Throws:

JSONException - If the key is null.



eclipse_workspace - C:₩Users₩Expert₩.m2₩repository₩org₩json₩json₩20180130₩json-20180130-sources.jar₩org₩json₩JSONObject.java - Eclipse IDE

```
<terminated>MyHelloWorld [Java Application]
    <disconnected>kr.ac.halla.n223.helloworld.MyHelloWorld at localhost:51141
    <terminated, exit value: 0>C:₩Program Files₩Java₩jdk1.8.0_181₩bin₩javaw.
MyJSONHelloWorld [Java Application]
    kr.ac.halla.n223.helloworld.MyJSONHelloWorld at localhost:57333
        Thread [main] (Suspended)
            JSONObject.put(String, Object) line: 1609
            MyJSONHelloWorld.main(String[]) line: 9
    C:₩Program Files₩Java₩jdk1.8.0_181₩bin₩javaw.exe (2018. 8. 4. 오전 12:07
```

```java
    public JSONObject put(String key, Map<?, ?> value) throws JSONException {
        this.put(key, new JSONObject(value));
        return this;
    }

    /**
     * Put a key/value pair in the JSONObject. If the value is null, then the
     * key will be removed from the JSONObject if it is present.
     *
     * @param key
     *            A key string.
     * @param value
     *            An object which is the value. It should be of one of these
     *            types: Boolean, Double, Integer, JSONArray, JSONObject, Long,
     *            String, or the JSONObject.NULL object.
     * @return this.
     * @throws JSONException
     *             If the value is non-finite number or if the key is null.
     */
    public JSONObject put(String key, Object value) throws JSONException {
        if (key == null) {
            throw new NullPointerException("Null key.");
        }
        if (value != null) {
            testValidity(value);
            this.map.put(key, value);
        } else {
            this.remove(key);
        }
        return this;
    }

    /**
     * Put a key/value pair in the JSONObject, but only if the key and the value
     * are both non-null, and only if there is not already a member with that
     * name.
     *
     * @param key string
     * @param value object
     * @return this.
     * @throws JSONException
     *             if the key is a duplicate
     */
```

Name	Value
no method return value	
this	JSONObject (id=78)
key	"message" (id=79)
value	"My JSON Hello World" (id=83)

```
MyJSONHelloWorld [Java Application] C:₩Program Files₩Java₩jdk1.8.0_181₩bin₩javaw.exe (2018. 8. 4. 오전 12:07:37)
```

Summary

- Preparation for Java Review
    - Open JDK 14
    - Eclipse IDE
    - Maven
    - Run and Debug first application
    - Run and Debug second application using 3rd party library

- Next Week
    - Review Java