

Guide of Our System

how to run “volleyball-match-record-system”

How to Register Information

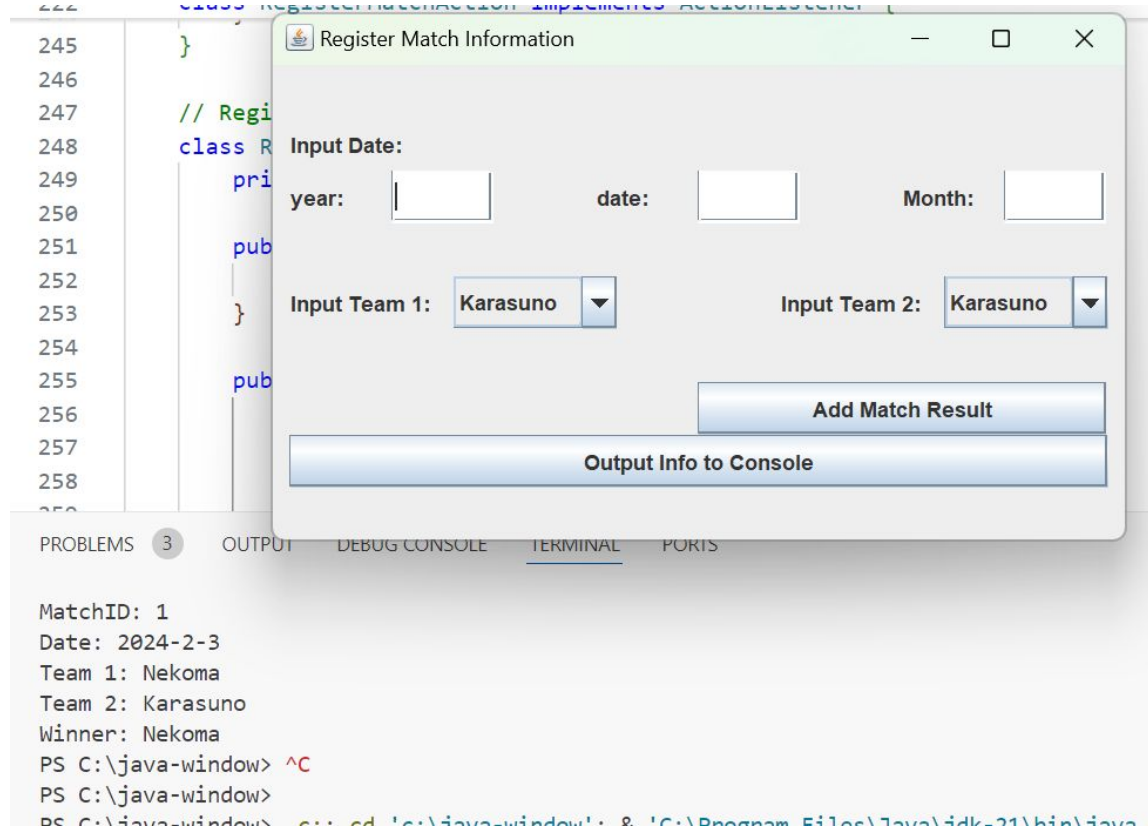
demo

demo

```
J RegisterMatchInfo.java X J TestRegisterMatchInfo.java • J TestList.java
J RegisterMatchInfo.java > RegisterMatchInfo > RegisterWinnerAction > actionPerformed(ActionEvent)
5 public class RegisterMatchInfo {
222     class RegisterMatchAction implements ActionListener {
245     }
246
247     // Register Winner by click button
248     class RegisterWinnerAction implements ActionListener {
249         private JComboBox<String> comboBox;
250
251         public RegisterWinnerAction(JComboBox<String> comboBox) {
252             this.comboBox = comboBox;
253         }
254
255         public void actionPerformed(ActionEvent e) {
256             winner = (String) comboBox.getSelectedItem();
257             createMatchListFrame();
258             winnerFrame.dispose();
259             // System.exit(0);
260         }
261     }
262
263     class DelButtonAction implements ActionListener {
264         public void actionPerformed(ActionEvent e) {
265             int index = matchList.getSelectedIndex();
266             if (index != -1) {
267                 listModel.remove(index);
268             } else {
269                 // System.out.println("No item selected");
270             }
271         }
272     }
273 }
```

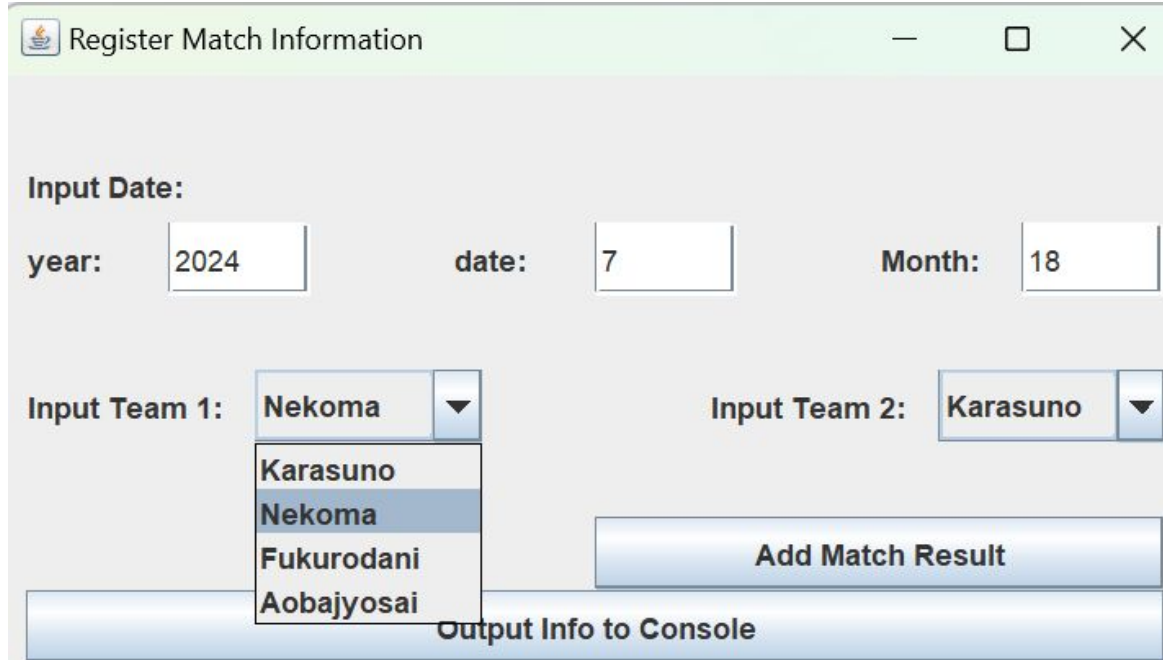
Register Match Information

When you run
"RegisterMatchInfo.java",
a window like the one
on the right will appear.



Step1 : Add Date and Team Information

Enter the date of the match and select the competing school.



The screenshot shows a window titled "Register Match Information" with a green title bar. Inside, there are input fields for the match date and team selection. The date is entered as 2024 for the year, 7 for the date, and 18 for the month. For the teams, "Input Team 1:" has a dropdown menu open showing "Nekoma" (selected), "Karasuno", "Fukurodani", and "Aobajyosai". "Input Team 2:" has a dropdown menu showing "Karasuno" (selected). There is a blue button labeled "Add Match Result" and a blue bar at the bottom labeled "Output Info to Console".

Register Match Information

Input Date:

year: 2024 date: 7 Month: 18

Input Team 1: Nekoma

Input Team 2: Karasuno

Karasuno
Nekoma
Fukurodani
Aobajyosai

Add Match Result

Output Info to Console

Step2-1 : Add Match Result

If match already finished, you can register winner team.

Click button “Add Match Result” (green color) then a new window “Register Winner” will appear.

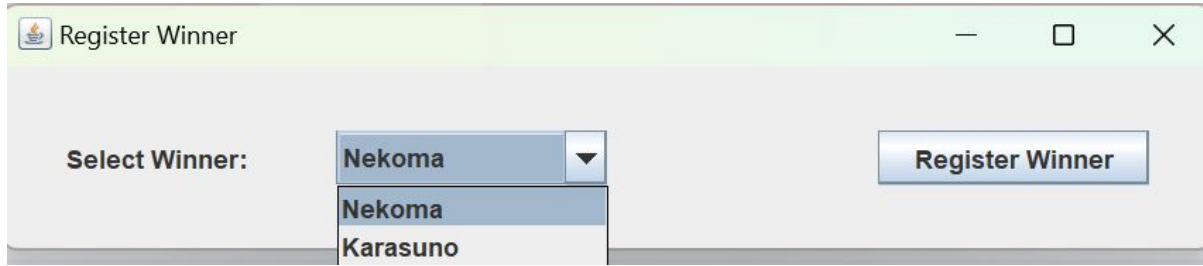
Point :

Registration not allowed until the match
Register as soon as the competing sch
(Should be able to register even before

The image shows two overlapping windows from a software application. The top window, titled 'Register Winner', has a light green header bar and contains a 'Select Winner:' label, a dropdown menu currently showing 'Nekoma', and a 'Register Winner' button. The bottom window, titled 'Register Match Information', has a grey header bar and contains several input fields: 'Input Date:' with sub-fields for 'year:' (2024), 'date:' (7), and 'Month:' (18); 'Input Team 1:' with a dropdown showing 'Nekoma'; and 'Input Team 2:' with a dropdown showing 'Karasuno'. At the bottom of this window is a large green button labeled 'Add Match Result', which is circled in green. A large green arrow points from the text 'click here' to this button. Below the 'Add Match Result' button is a button labeled 'Output info to Console'. On the left side of the 'Register Match Information' window, there is a sidebar with a 'メニュー' (Menu) icon and a list of items including 'Guide of Our System', 'Register Match Information', 'Add Match Data and Team Information', and 'Add Match Result (Winner)'.

Step2-2 : Add Match Result

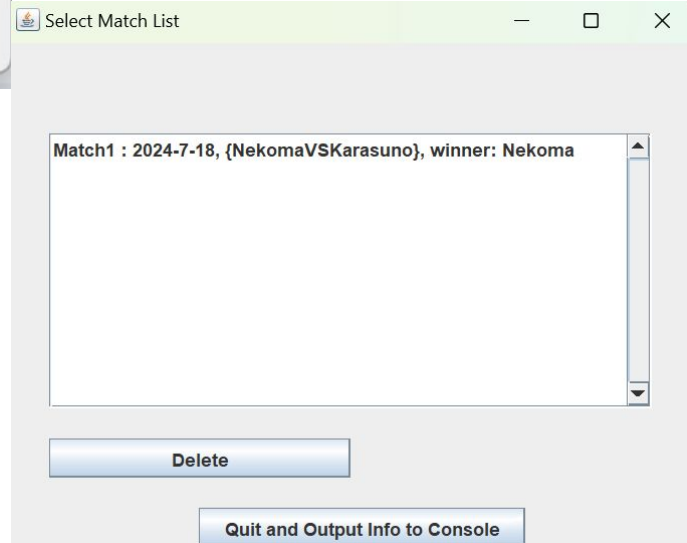
Select winner and click button “Register Winner”.



The "Register Winner" window features a title bar with a small icon, the text "Register Winner", and standard window controls (minimize, maximize, close). The main area contains a label "Select Winner:" followed by a dropdown menu. The dropdown menu is open, showing "Nekoma" as the selected option, with "Nekoma" and "Karasuno" as other visible options. To the right of the dropdown is a button labeled "Register Winner".

Then new window “Select Match List” will appear.

If you register a wrong information, you can delete it by select line in the field and click delete button.



The "Select Match List" window has a title bar with a small icon, the text "Select Match List", and standard window controls. The main area contains a text field displaying "Match1 : 2024-7-18, {NekomaVSKarasuno}, winner: Nekoma". Below the text field is a "Delete" button. At the bottom of the window is a button labeled "Quit and Output Info to Console".

Step3 : Output information

If you click the button “Quit and Output Info to Console”, you can see the output as below.

```
PS C:\java-window> c::; cd 'c:\java-window'; & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Use
rs\lydwo\AppData\Roaming\Code\User\workspaceStorage\f84cd399d19ee08df405d99fb9b2580c\redhat.java\jdt_ws\java-window_983fec43\bin' 'RegisterMatchIn
fo'
```

MatchID: 1
Date: 2024-7-18
Team 1: Nekoma
Team 2: Karasuno
Winner: Nekoma

How to Get Information

Try 1: Get information of Player 1

Command: `./gradlew run --args="GetPlayerInfo 1"`

Successful output: You can see a similar output as below

```
PS C:\volleyball-app\multi-storage-transaction-sample> ./gradlew run --args="GetPlayerInfo 1"
Starting a Gradle Daemon, 3 incompatible and 3 stopped Daemons could not be reused, use --status for details
```

```
> Task :compileJava
```

隴ヲ蜻・[options] 縵入綱シ縵ケ蛟、8纏ツ蟒・、「纏輔」| 纏ヲ纏・、縹サ雁セ後・、綱I綱I綱シ縵ケ纏ヲ蜃企勁纏輔「縵俱」亥ヨ壹千纏・

ヲ縛・縋ヲ雁セ後・綱₁綱₁綱₁縋ケ縛_ア蜚企勁縛輔₁縋俱₂亥₃壹₄千縛₅・

縛ッ縹・xlint:縹I縹励す縹ア縹ウ縹剃ッ逕イ縹励ウ縹吮・

隴ヲ蜻・蛟・

> Task :run

```
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
```

SLF4J: Defaulting to no-operation (NOP) logger implementation

SLF4J: See <http://www.slf4j.org/codes.html#StaticLoggerBinder> for further details.

```
{ "playerID": 1, "name": "Kageyama Tobio", "team": "Karasuno", "grade": 1, "position": "S", "status": 27 }
```

Deprecated Gradle features were used in this build, making it incompatible with Gradle 8.0.

You can use `--warning-mode all` to show the individual deprecation warnings and determine if they come from your own scripts or plugins.

How to run sample app

Step 0 : Download JAVA

Recommend to download Java 8, 11, 17, 21

We use version 21

Prerequisites

- One of the following Java Development Kits (JDKs):
 - [Oracle JDK](#) LTS version (8, 11, 17 or 21)
 - OpenJDK LTS version (8, 11, 17, or 21) from [Eclipse Temurin](#), [Amazon Corretto](#), or [Microsoft](#)
- [Docker](#) 20.10 or later with [Docker Compose](#) V2 or later

NOTE

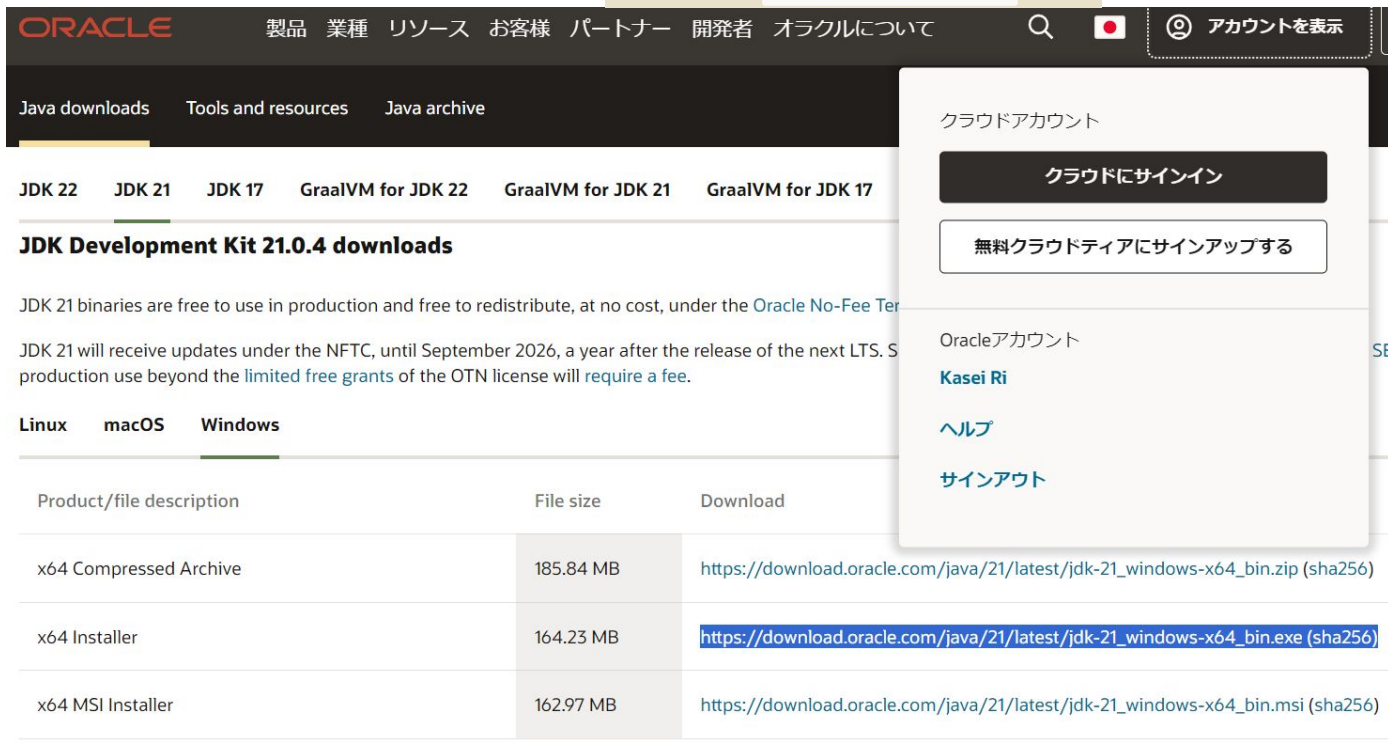
We recommend using the LTS versions mentioned above, but other non-LTS versions may work.

In addition, other JDKs should work with ScalarDB, but we haven't tested them.

Step 0: Download Java

You might be required to create oracle account

Step for Windows →



The screenshot shows the Oracle website's Java download page. The top navigation bar includes the Oracle logo and links for products, resources, customers, partners, developers, and Oracle information. Below this, the 'Java downloads' section is active, showing links for JDK 22, JDK 21, JDK 17, and GraalVM for JDK 22, 21, and 17. The 'JDK 21' link is selected, leading to the 'JDK Development Kit 21.0.4 downloads' page. This page explains that JDK 21 binaries are free for production and redistribution under the Oracle No-Fee Terms. It also mentions that updates will be provided under the NPTC until September 2026. The 'Windows' tab is selected under the operating system filters. A table lists the available download options for Windows, including a compressed archive, an installer, and an MSI installer, each with its file size and a direct download link. A sign-in modal is overlaid on the right side of the page, prompting the user to sign in with a cloud account or create a new one.

ORACLE 製品 業種 リソース お客様 パートナー 開発者 オラクルについて

Java downloads Tools and resources Java archive

JDK 22 JDK 21 JDK 17 GraalVM for JDK 22 GraalVM for JDK 21 GraalVM for JDK 17

JDK Development Kit 21.0.4 downloads

JDK 21 binaries are free to use in production and free to redistribute, at no cost, under the [Oracle No-Fee Terms](#).

JDK 21 will receive updates under the NPTC, until September 2026, a year after the release of the next LTS. Such production use beyond the [limited free grants](#) of the OTN license will [require a fee](#).

Linux macOS **Windows**

Product/file description	File size	Download
x64 Compressed Archive	185.84 MB	https://download.oracle.com/java/21/latest/jdk-21_windows-x64_bin.zip (sha256)
x64 Installer	164.23 MB	https://download.oracle.com/java/21/latest/jdk-21_windows-x64_bin.exe (sha256)
x64 MSI Installer	162.97 MB	https://download.oracle.com/java/21/latest/jdk-21_windows-x64_bin.msi (sha256)

サイン・イン

Username or email

次

[Forgot username?](#)

Oracleアカウントをお持ちではありませんか。

アカウントの作成

© Oracle | 使用条件 | プライバシー | ポリシー

アカウントを表示

クラウドアカウント

クラウドにサインイン

無料クラウドティアにサインアップする

Oracleアカウント

[Kasei Ri](#)

[ヘルプ](#)

[サインアウト](#)






Step 0 : Download schema loader

Schema loader [Download](#)

Please select the highlighted section below

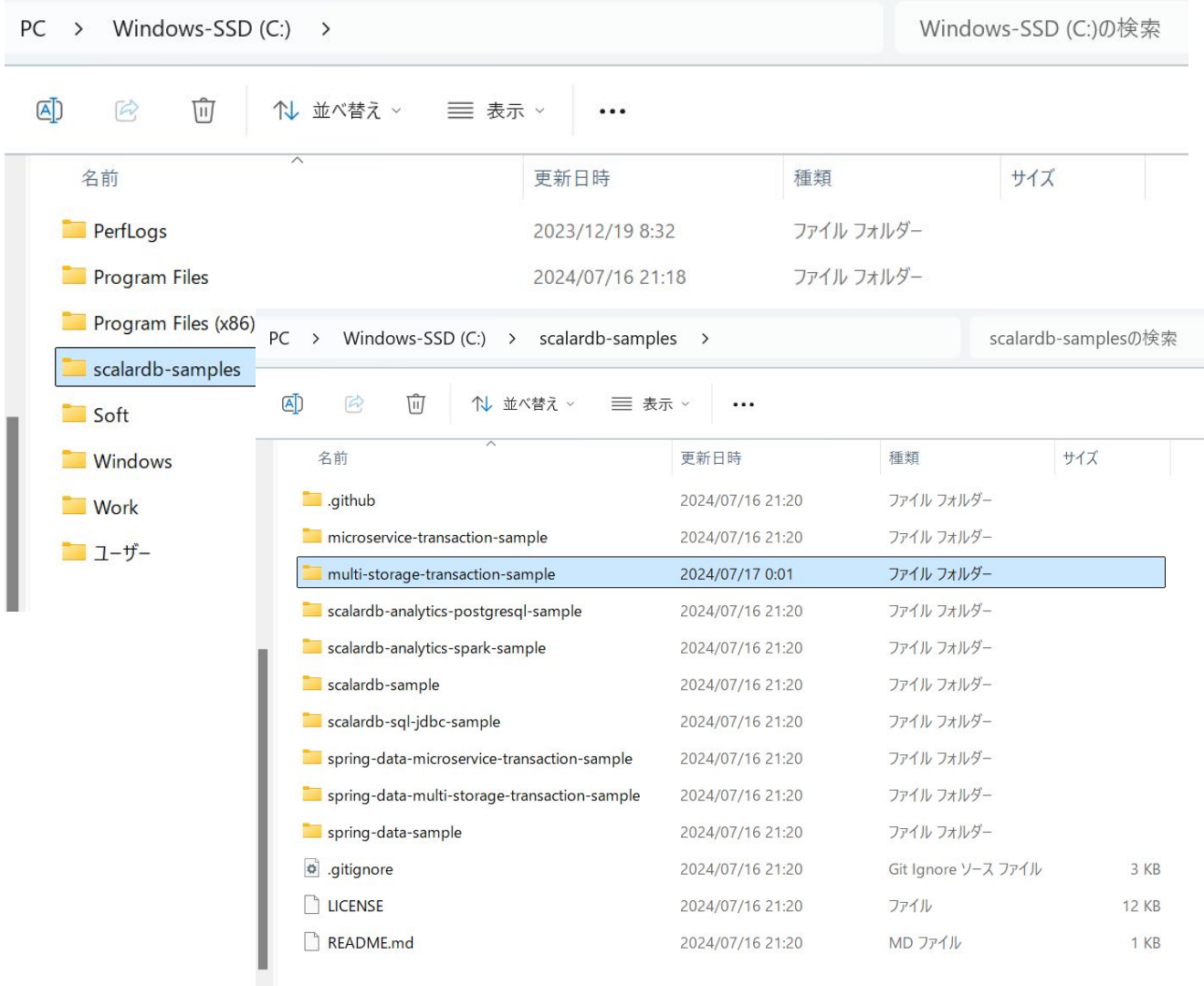
▼ Assets

6

 scalardb-cluster-schema-loader-3.12.3-all.jar	104 MB	2 weeks ago
 scalardb-cluster-sql-cli-3.12.3-all.jar	107 MB	2 weeks ago
 scalardb-schema-loader-3.12.3.jar	65.7 MB	2 weeks ago
 scalardb-server-3.12.3.zip	67.3 MB	2 weeks ago
 Source code (zip)		2 weeks ago
 Source code (tar.gz)		2 weeks ago

Step 1 : Git clone

Clone succesfully



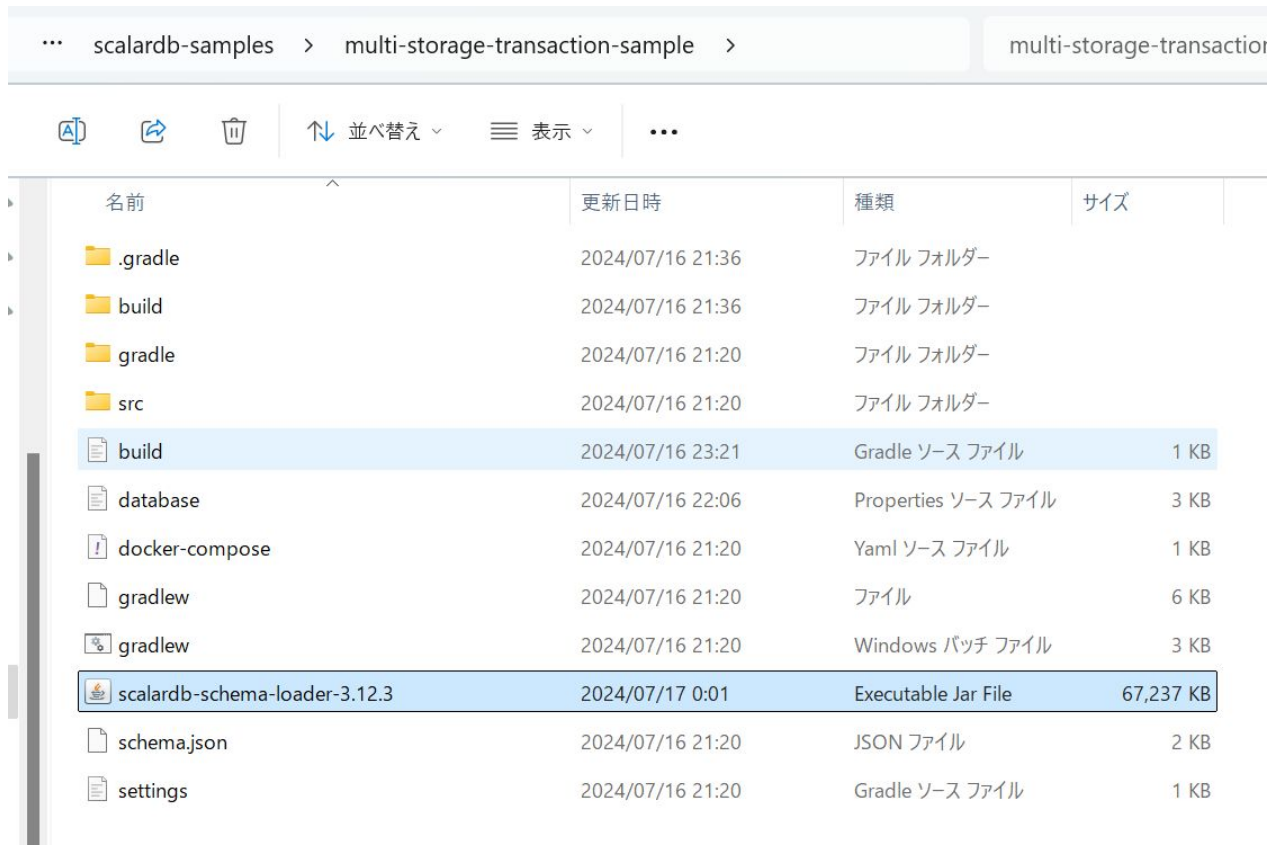
Step 3 : Check and Move schema loader

As shown in the figure
on the right

<Check the version>

We use 3.12.3

it will be used at next step



... scalardb-samples > multi-storage-transaction-sample > multi-storage-transaction

📁 📄 🗑️ ⬆️ 並べ替え ▾ ≡ 表示 ▾ ⋮

名前	更新日時	種類	サイズ
📁 .gradle	2024/07/16 21:36	ファイル フォルダー	
📁 build	2024/07/16 21:36	ファイル フォルダー	
📁 gradle	2024/07/16 21:20	ファイル フォルダー	
📁 src	2024/07/16 21:20	ファイル フォルダー	
📄 build	2024/07/16 23:21	Gradle ソース ファイル	1 KB
📄 database	2024/07/16 22:06	Properties ソース ファイル	3 KB
📄 docker-compose	2024/07/16 21:20	Yaml ソース ファイル	1 KB
📄 gradlew	2024/07/16 21:20	ファイル	6 KB
📄 gradlew	2024/07/16 21:20	Windows パッチ ファイル	3 KB
📄 scalardb-schema-loader-3.12.3	2024/07/17 0:01	Executable Jar File	67,237 KB
📄 schema.json	2024/07/16 21:20	JSON ファイル	2 KB
📄 settings	2024/07/16 21:20	Gradle ソース ファイル	1 KB

Step 4 : Start cassandra and mysql

Use Docker

Open “Docker Desktop” first



and input Command: `docker-compose up -d`

The screenshot shows the Docker Desktop application window. On the left, the "Containers" tab is active, displaying a table with columns: Name, Image, Status, and Port(s). The table shows a container named "multi-storage" with status "Running (2/2)". On the right, a detailed view of the "multi-storage-transaction-sample" project is shown, listing two containers: "mysql-1" (Running, Image: mysql:8.0) and "cassandra-1" (Running, Image: cassandra:3.11). Below the Docker Desktop window, a Windows PowerShell terminal window is open, showing the command `docker-compose up -d` being executed. The terminal output indicates that the containers were successfully created and started. The status bar at the bottom of the Docker Desktop window shows "Engine running" and system resources like RAM and CPU usage.

docker desktop

Containers

Container CPU usage 0.00% / 1200% (12 CPUs available)

Containers

multi-storage-transaction-sample

mysql-1

mysql:8.0

Running

cassandra-1

cassandra:3.11

Running

multi-storage

Running (2/2)

Windows PowerShell

```
PS C:\scalardb-samples\multi-storage-transaction-sample> docker-compose up -d
time="2024-07-17T09:24:12+09:00" level=warning msg="C:\\scalardb-samples\\multi-storage-transaction-sample\\docker-compo
se.yml: 'version' is obsolete"
[+] Running 3/3
✔ Network multi-storage-transaction-sample_default Created 0.0s
✔ Container cassandra-1 Started 0.9s
✔ Container mysql-1 Started 0.9s
```

Engine running RAM 3.70 GB CPU 0.08%

Engine running RAM 3.72 GB CPU 0.00%

New version available

Step 5 : Load schema

Command: java -jar scalardb-schema-loader-**<VERSION>**.jar --config database.properties
--schema-file schema.json --coordinator

Replace **<VERSION>**

java -jar scalardb-schema-loader-**3.12.3**.jar --config database.properties --schema-file schema.json
--coordinator

```
PS C:\scalardb-samples\multi-storage-transaction-sample> ls
```

ディレクトリ : C:\scalardb-samples\multi-storage-transaction-sample

Mode	LastWriteTime		Length	Name
----	-----	-----	-----	----
d-----	2024/07/16	21:36		.gradle
d-----	2024/07/16	21:36		build
d-----	2024/07/16	21:20		gradle
d-----	2024/07/16	21:20		src
-a-----	2024/07/16	23:21	420	build.gradle
-a-----	2024/07/16	22:06	2298	database.properties
-a-----	2024/07/16	21:20	284	docker-compose.yml
-a-----	2024/07/16	21:20	5959	gradlew
-a-----	2024/07/16	21:20	2763	gradlew.bat
-a-----	2024/07/17	0:01	68849812	scalardb-schema-loader-3.12.3.jar
-a-----	2024/07/16	21:20	1025	schema.json
-a-----	2024/07/16	21:20	55	settings.gradle

Step 5

The successful result is shown below

```
PS C:\scalardb-samples\multi-storage-transaction-sample> java -jar scalardb-schema-loader-3.12.3.jar --config database.p
roperties --schema-file schema.json --coordinator
[main] INFO com.scalar.db.schemaloader.command.SchemaLoaderCommand - Config path: database.properties
[main] INFO com.scalar.db.schemaloader.command.SchemaLoaderCommand - Schema path: schema.json
[main] INFO com.datastax.driver.core - DataStax Java driver 3.11.5 for Apache Cassandra
[main] INFO com.datastax.driver.core.GuavaCompatibility - Detected Guava >= 19 in the classpath, using modern compatibil
ity layer
[main] INFO com.datastax.driver.core.Native - Could not load JNR C Library, native system calls through this library wil
l not be available (set this logger level to DEBUG to see the full stack trace).
[main] INFO com.datastax.driver.core.ClockFactory - Using java.lang.System clock to generate timestamps.
[main] WARN com.datastax.driver.core.NettyUtil - Found Netty's native epoll transport, but not running on linux-based op
erating system. Using NIO instead.
[main] INFO com.datastax.driver.core.Cluster - Cannot connect with protocol version V5, trying with V4
[main] INFO com.datastax.driver.core.policies.DCAwareRoundRobinPolicy - Using data-center name 'datacenter1' for DCAware
RoundRobinPolicy (if this is incorrect, please provide the correct datacenter name with DCAwareRoundRobinPolicy construc
tor)
[main] INFO com.datastax.driver.core.Cluster - New Cassandra host localhost/[0:0:0:0:0:0:0:1]:9042 added
[main] INFO com.scalar.db.storage.cassandra.ClusterManager - Session to the cluster is created
[main] INFO com.scalar.db.schemaloader.SchemaOperator - Creating the table customers in the namespace customer succeeded
[main] INFO com.scalar.db.schemaloader.SchemaOperator - Creating the table orders in the namespace order succeeded
[main] INFO com.scalar.db.schemaloader.SchemaOperator - Creating the table statements in the namespace order succeeded
[main] INFO com.scalar.db.schemaloader.SchemaOperator - Creating the table items in the namespace order succeeded
[main] INFO com.scalar.db.schemaloader.SchemaOperator - Creating the coordinator tables succeeded
```

Step 6 : Load initial data

Command:

```
./gradlew run --args="LoadInitialData"
```

<Attention> If you use Command prompt then use

```
gradlew run --args="LoadInitialData"
```

or

```
gradlew.bat run --args="LoadInitialData"
```

Step 6: Expected reaction

<Notice> Use java 21: Garbled text will appear → (you can use java 8)

```
PS C:\scalardb-samples\multi-storage-transaction-sample> ./gradlew run --args="LoadInitialData"
Starting a Gradle Daemon, 2 incompatible Daemons could not be reused, use --status for details

> Task :compileJava
隴ヲ蜻・ [options] 縵ス綱シ縵ケ蛟、8縵ッ蜻・ ヌ「縵輔 | 縵ヲ縵・ ※縵ヲ雁セ後・ 綱ニ綱ニ綱シ縵ケ縵ヲ蜚企勁縵輔 | 縵俱コ亥ヨ壹ニ縵・
隴ヲ蜻・ [opt
ions] 縵ヲ綱シ縵イ綱・ ヨ 蛟、8縵ッ蜻・ ヌ「縵輔 | 縵ヲ縵・ ※縵ヲ雁セ後・ 綱ニ綱ニ綱シ縵ケ縵ヲ蜚企勁縵輔 | 縵俱コ亥ヨ壹ニ縵・
隴ヲ蜻・ [options]
蜻・ ヌ「縵輔 | 縵蓋が綱励す 綱ヲ綱ヲ縵ヲ縵、縵・ ※縵ヨ隴ヲ蜻翫 r 隴イ追コ縵励↑縵・ h 縵・ ↓縵呐 k 縵ヲ縵ヲ縵・ Xlint:縵ニ綱励す 綱ヲ綱ヲ縵剃ソ逕
イ縵励 ム縵呐・
隴ヲ蜻・ 蛟・

> Task :run
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.

Deprecated Gradle features were used in this build, making it incompatible with Gradle 8.0.

You can use '--warning-mode all' to show the individual deprecation warnings and determine if they come from your own sc
ripts or plugins.

See https://docs.gradle.org/7.6/userguide/command_line_interface.html#sec:command_line_warnings

BUILD SUCCESSFUL in 14s
2 actionable tasks: 2 executed
```

Let's use sample app

how to get information from database

Try 1 : Get player information from database

Successful output: You can see a similar output as below

```
{"id": 1, "name": "Yamada Taro", "credit_limit": 10000, "credit_total": 0}
```

```
PS C:\scalardb-samples\multi-storage-transaction-sample> ./gradlew run --args="GetCustomerInfo 1"
```

```
> Task :run
```

```
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
```

```
SLF4J: Defaulting to no-operation (NOP) logger implementation
```

```
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
```

```
{"id": 1, "name": "Yamada Taro", "credit_limit": 10000, "credit_total": 0}
```

```
Deprecated Gradle features were used in this build, making it incompatible with Gradle 8.0.
```

```
You can use '--warning-mode all' to show the individual deprecation warnings and determine if they come from your own scripts or plugins.
```

```
See https://docs.gradle.org/7.6/userguide/command\_line\_interface.html#sec:command\_line\_warnings
```

```
BUILD SUCCESSFUL in 8s
```

```
2 actionable tasks: 1 executed, 1 up-to-date
```


Try 2 : customer order

Then, customer (ID 1) place an order for three apples and two oranges by running the following command:

```
PS C:\scalardb-samples\multi-storage-transaction-sample> ./gradlew run --args="PlaceOrder 1 1:3,2:2"
```

```
> Task :run
```

```
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
```

```
SLF4J: Defaulting to no-operation (NOP) logger implementation
```

```
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
```

```
{"order_id": "4913664c-06db-4ca4-a010-b168ec641ae5"}
```

```
Deprecated Gradle features were used in this build, making it incompatible with Gradle 8.0.
```

```
You can use '--warning-mode all' to show the individual deprecation warnings and determine if they come from your own scripts or plugins.
```

```
See https://docs.gradle.org/7.6/userguide/command\_line\_interface.html#sec:command\_line\_warnings
```

```
BUILD SUCCESSFUL in 8s
```

```
2 actionable tasks: 1 executed, 1 up-to-date
```


Try 3 :

This is error output

```
PS C:\scalardb-samples\multi-storage-transaction-sample> ./gradlew run --args="GetOrder <1>"
```

```
> Task :run
```

```
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
```

```
SLF4J: Defaulting to no-operation (NOP) logger implementation
```

```
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
```

```
java.lang.RuntimeException: Order not found
```

```
    at sample.Sample.getOrderJson(Sample.java:243)
```

```
    at sample.Sample.getOrderById(Sample.java:315)
```

```
    at sample.command.GetOrderCommand.call(GetOrderCommand.java:17)
```

```
    at sample.command.GetOrderCommand.call(GetOrderCommand.java:8)
```

```
    at picocli.CommandLine.executeUserObject(CommandLine.java:2041)
```

```
    at picocli.CommandLine.access$1500(CommandLine.java:148)
```

```
    at picocli.CommandLine$RunLast.executeUserObjectOfLastSubcommandWithSameParent(CommandLine.java:2461)
```

```
    at picocli.CommandLine$RunLast.handle(CommandLine.java:2453)
```

```
    at picocli.CommandLine$RunLast.handle(CommandLine.java:2415)
```

```
    at picocli.CommandLine$AbstractParseResultHandler.execute(CommandLine.java:2273)
```

```
    at picocli.CommandLine$RunLast.execute(CommandLine.java:2417)
```

```
    at picocli.CommandLine.execute(CommandLine.java:2170)
```

```
    at sample.command.SampleCommand.main(SampleCommand.java:35)
```

```
Deprecated Gradle features were used in this build, making it incompatible with Gradle 8.0.
```

```
You can use '--warning-mode all' to show the individual deprecation warnings and determine if they come from your own scripts or plugins.
```

```
See https://docs.gradle.org/7.6/userguide/command\_line\_interface.html#sec:command\_line\_warnings
```

```
BUILD SUCCESSFUL in 6s
```

```
2 actionable tasks: 1 executed, 1 up-to-date
```

The last

docker-compose down

exit

```
PS C:\scalardb-samples\multi-storage-transaction-sample> docker-compose down
time="2024-07-17T10:21:07+09:00" level=warning msg="C:\\scalardb-samples\\multi-storage-transaction-sample\\docker-compo
se.yml: 'version' is obsolete"
[+] Running 3/3
  ✓ Container cassandra-1      Removed                2.9s
  ✓ Container mysql-1          Removed                1.6s
  ✓ Network multi-storage-transaction-sample_default Removed            0.2s
PS C:\scalardb-samples\multi-storage-transaction-sample> exit
```

Appendix

Cannot load schema

`./gradlew run --args="LoadInitialData" → error`

Check java version and reload schema (step 5)

```
PS C:\scalardb-samples\multi-storage-transaction-sample> java -version
java version "21.0.4" 2024-07-16 LTS
Java(TM) SE Runtime Environment (build 21.0.4+8-LTS-274)
Java HotSpot(TM) 64-Bit Server VM (build 21.0.4+8-LTS-274, mixed mode, sharing)
```

Check your environment path “JAVA_HOME”