

[Start Lab](#)

01:30:00

Working with Artifact Registry

 Lab  1 hour  No cost  Intermediate**GSP1076****Google Cloud Self-Paced Labs**[Lab instructions and tasks](#)

GSP1076

Overview

Objectives

Setup and requirement

Task 1. Prepare the lab environment

Task 2. Working with container images

Task 3. Integration with Cloud Code

Task 4. Working with language packages

Congratulations!



Registry

 Lab  1 hour  No cost  Intermediate**GSP1076****Google Cloud Self-Paced Labs**

Registry

 Lab  1 hour  No cost  Intermediate**GSP1076****Google Cloud Self-Paced Labs**

Registry

Lab 1 hour No cost Intermediate



GSP1076



Google Cloud Self-Paced Labs

Registry

Lab 1 hour No cost Intermediate



GSP1076



Google Cloud Self-Paced Labs

Registry

Lab 1 hour No cost Intermediate



GSP1076



Google Cloud Self-Paced Labs

Registry

 Lab  1 hour  No cost  Intermediate



GSP1076



Google Cloud Self-Paced Labs

Registry

 Lab  1 hour  No cost  Intermediate



GSP1076



Google Cloud Self-Paced Labs

Registry

 Lab  1 hour  No cost  Intermediate



GSP1076

Registry

 Lab  1 hour  No cost  Intermediate



GSP1076

 Google Cloud Self-Paced Labs

Registry

 Lab  1 hour  No cost  Intermediate



GSP1076

 Google Cloud Self-Paced Labs

Registry

 Lab  1 hour  No cost  Intermediate



GSP1076



Registry

Lab 1 hour No cost Intermediate



GSP1076



Registry

Lab 1 hour No cost Intermediate



GSP1076



Registry

Lab 1 hour No cost Intermediate



GSP1076



Cloud Registry

Lab 1 hour No cost Intermediate



GSP1076



Cloud Registry

Lab 1 hour No cost Intermediate



GSP1076



Cloud Registry

Lab 1 hour No cost Intermediate



GSP1076



Registry

Lab 1 hour No cost Intermediate



GSP1076



Registry

Lab 1 hour No cost Intermediate



GSP1076



Registry

Lab 1 hour No cost Intermediate



GSP1076



Google Cloud Self-Paced Labs

Registry

Lab 1 hour No cost Intermediate



GSP1076



Google Cloud Self-Paced Labs

Registry

Lab 1 hour No cost Intermediate



GSP1076



Google Cloud Self-Paced Labs

Registry

Lab 1 hour No cost Intermediate



GSP1076



Registry

Lab 1 hour No cost Intermediate



GSP1076



Registry

Lab 1 hour No cost Intermediate



GSP1076

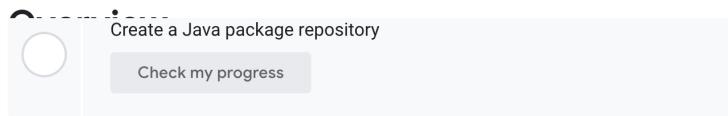


Registry

Lab 1 hour No cost Intermediate



GSP1076



Set up authentication to Artifact Registry

- Use the following command to update the well-known location for Application Default Credentials (ADC) with your user account credentials so that the Artifact Registry credential helper can authenticate using them when connecting with repositories:

```
gcloud auth login --update-adc
```



If prompted to authenticate:

- Choose **Y**.
- Paste the code into a browser window.
- Select Google Sign In and sign in using the credentials in the lab.
- Copy the authentication code from the browser back into the console to complete the authentication.

Configure Maven for Artifact Registry

1. Run the following command to print the repository configuration to add to your Java project:

```
gcloud artifacts print-settings mvn \
    --repository=container-dev-java-repo \
    --location="REGION"
```



2. Open the `pom.xml` in Cloud Shell Editor and add the returned settings to the appropriate sections in the file:

- Add the **distributionManagement** section.

```
<distributionManagement>
    <snapshotRepository>
        <id>artifact-registry</id>
        <url>artifactregistry://"REGION"-maven.pkg.dev/"PROJECT_ID"/c
    </snapshotRepository>
    <repository>
        <id>artifact-registry</id>
        <url>artifactregistry://"REGION"-maven.pkg.dev/"PROJECT_ID"/c
    </repository>
</distributionManagement>
```



- Add the **repositories** section.

```
<repositories>
```



```

<repository>
  <id>artifact-registry</id>
  <url>artifactregistry://"REGION"-maven.pkg.dev/"PROJECT_ID"/cc
  <releases>
    <enabled>true</enabled>
  </releases>
  <snapshots>
    <enabled>true</enabled>
  </snapshots>
</repository>
</repositories>

```

- Update the **extensions** in the Builds section.

```

<extensions>
  <extension>
    <groupId>com.google.cloud.artifactregistry</groupId>
    <artifactId>artifactregistry-maven-wagon</artifactId>
    <version>2.1.0</version>
  </extension>
</extensions>

```

Here's an example of the complete file for your reference.

```

<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd" modelVersion="4.0.0">

  <artifactId>hello-world</artifactId>
  <packaging>jar</packaging>
  <name>Cloud Code Hello World</name>
  <description>Getting started with Cloud Code</description>
  <version>1.0.0</version>
  <distributionManagement>
    <snapshotRepository>
      <id>artifact-registry</id>
      <url>artifactregistry://"REGION"-maven.pkg.dev/"PROJECT_ID"/cc
    </snapshotRepository>
    <repository>
      <id>artifact-registry</id>
      <url>artifactregistry://"REGION"-maven.pkg.dev/"PROJECT_ID"/cc
    </repository>
  </distributionManagement>

  <repositories>
    <repository>
      <id>artifact-registry</id>
      <url>artifactregistry://"REGION"-maven.pkg.dev/"PROJECT_ID"/cc
      <releases>
        <enabled>true</enabled>
      </releases>
      <snapshots>
        <enabled>true</enabled>
      </snapshots>
    </repository>
  </repositories>

  <parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>2.6.3</version>
  </parent>

  <properties>
    <java.version>1.8</java.version>
    <checkstyle.config.location>./checkstyle.xml</checkstyle.config.location>
  </properties>

  <build>
    <plugins>
      <plugin>
        <groupId>com.google.cloud.artifactregistry</groupId>
        <artifactId>artifactregistry-maven-wagon</artifactId>
        <version>2.1.0</version>
        <configuration>
          <url>artifactregistry://"REGION"-maven.pkg.dev/"PROJECT_ID"/cc</url>
        </configuration>
      </plugin>
    </plugins>
  </build>

```

```

<plugins>
    <plugin>
        <groupId>com.google.cloud.tools</groupId>
        <artifactId>jib-maven-plugin</artifactId>
        <version>3.2.0</version>
    </plugin>
    <plugin>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-maven-plugin</artifactId>
    </plugin>
    <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-checkstyle-plugin</artifactId>
        <version>3.1.2</version>
    </plugin>
</plugins>
<extensions>
    <extension>
        <groupId>com.google.cloud.artifactregistry</groupId>
        <artifactId>artifactregistry-maven-wagon</artifactId>
        <version>2.1.0</version>
    </extension>
</extensions>
</build>


<dependencyManagement>
    <dependencies>
        <dependency>
            <groupId>org.springframework.cloud</groupId>
            <artifactId>spring-cloud-gcp-dependencies</artifactId>
            <version>1.2.8.RELEASE</version>
            <type>pom</type>
            <scope>import</scope>
        </dependency>
    </dependencies>
</dependencyManagement>

<dependencies>

    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter</artifactId>
    </dependency>

    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-jetty</artifactId>
    </dependency>

    <dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-webmvc</artifactId>
    </dependency>

    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-thymeleaf</artifactId>
    </dependency>

    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-test</artifactId>
        <scope>test</scope>
    </dependency>

    <dependency>
        <groupId>org.springframework.cloud</groupId>
        <artifactId>spring-cloud-gcp-starter-logging</artifactId>
    </dependency>

</dependencies>
</project>

```

Upload your Java package to Artifact Registry

With Artifact Registry configured in Maven, you can now use Artifact Registry to store Java Jars for use by other projects in your organization.

- Enter the below command to change to `java-hello-world` folder.

```
cd ~/cloud-code-samples/java/java-hello-world
```



- Run the following command to upload your Java package to Artifact Registry:

```
mvn deploy
```



Check the Java package in Artifact Registry

In the Cloud console go to **Artifact Registry > Repositories** and click into `container-dev-java-repo` to check that the `hello-world` binary artifact is there:

qwiklabs-gcp-02-d07b64265aef > us-west1 > container-dev-java-repo

Repository Details

Format	Maven
Type	Standard

[▼ SHOW MORE](#)

[Filter](#) Enter property name or value

<input type="checkbox"/> Name ↑	Created	Updated
org.springframework.boot:hello-world	Just now	Just now

Congratulations!

In this lab you learned about some of the features available in Artifact Registry. You first created repositories for containers and language packages. You then managed container images with Artifact Registry and integrated it with Cloud Code. Finally, you configured Maven to use Artifact Registry for Java dependencies. You now have a solid understanding of features available in Artifact Registry.

Google Cloud training and certification

...helps you make the most of Google Cloud technologies. [Our classes](#) include technical skills and best practices to help you get up to speed quickly and continue your learning journey. We offer fundamental to advanced level training, with on-demand, live, and virtual options to suit your busy schedule. [Certifications](#) help you validate and prove your skill and expertise in Google Cloud technologies.

Manual Last Updated May 9, 2024

Lab Last Tested May 9, 2024

Copyright 2024 Google LLC All rights reserved. Google and the Google logo are trademarks of Google LLC. All other company and product names may be trademarks of the respective companies with which they are associated.