



Getting On Board with Java in the Cloud

A HANDS-ON, STEP-BY-STEP GUIDE TO TRYING OUT AN ENTERPRISE CLOUD

BY HARSHAD OAK

The Java cloud market is evolving quickly. Vendors offer myriad cloud products along with new terminology and jargon. But getting started in the cloud doesn't have to be difficult. This guide explains what Oracle Java Cloud Service is and provides step-by-step instructions for getting on board.

In their early days, cloud solutions were mostly lightweight slivers of traditional server solutions. But today, we have pretty much all the functionality of large server-side products—often split into many specialized cloud solutions. This model comes with the benefit that developers are able to pick and choose enterprise options, and use only what’s required.



Source: iStockphoto

Cloud solutions are commonly classified as follows:

- **IaaS.** Infrastructure as a service, or basic virtualized hardware and an operating system
- **PaaS.** Platform as a service, or IaaS with additional services, such as a database
- **SaaS.** Software as a service, or full applications on top of a PaaS stack

Within these categories, there are dozens of specialized cloud solutions. For example, Oracle offers many different cloud services in each of these categories. While at first this might appear overwhelming, the good news is that almost all cloud solutions are not “new” technology as such, which would require an understanding of new core technology. The challenge is more about getting accustomed to new interfaces, workflows, and terminology.

In this article, I explain what Oracle Java Cloud Service is and how to get on board.

Oracle Java Cloud Service and Variants

Oracle Java Cloud Service began life a few years ago as a shared PaaS environment that offered support for commonly used Java EE technologies. Back then, it did not offer any fine control over the environment or the ability to tweak and customize based on requirements.

My previous Java cloud articles in *Java Magazine* (“Hands On with Oracle Java Cloud Service,” September/October 2013, and “Build with NetBeans IDE, Deploy to Oracle Java Cloud Service,” May/June 2014) discussed earlier versions of Oracle Java Cloud Service. Since then, Oracle has significantly enhanced its Oracle Java Cloud Service solutions.

TODAY, ORACLE HAS THE FOLLOWING THREE ORACLE JAVA CLOUD SERVICE OFFERINGS:

› ORACLE JAVA CLOUD SERVICE - SAAS EXTENSION

This is the Oracle Java Cloud Service offering that has been available the longest. It was renamed with *SaaS Extension* appended after the other two cloud services were launched.

As the name suggests, the primary use case this solution addresses is that of an Oracle SaaS user who needs to extend the capabilities of a SaaS offering. Because Oracle Java Cloud Service - SaaS Extension is primarily designed for this purpose, it offers easy integration with Oracle’s SaaS solutions. Although the name includes *SaaS Extension*, nothing in the product restricts you from deploying a standalone Java EE application that is not an extension of a SaaS cloud.

Oracle Java Cloud Service - SaaS Extension provides a shared PaaS environment where you can easily deploy Java EE applications without having to worry about any of the underlying hardware setup, server installations, patching, management, and more. Oracle Java Cloud Service - SaaS Extension supports all the commonly used Java EE technologies such as servlets, JavaServer Pages, JavaServer Faces, and Enterprise JavaBeans. It supports JAX-WS and REST web services. It also supports Oracle Application Development Framework, which is widely used among Oracle developers. It has most of the things Java EE applications require and is certainly a solution to consider when you are looking for a no-hassle, out-of-the-box shared PaaS environment for Java EE.

Oracle Java Cloud Service - SaaS Extension does not let you configure the application server, the JVM, or the operating system to your exact requirements. This can work as an advantage in cases where the user does not want to be bothered by those things. However, in some enterprise applications, especially, greater control might be desired. Enter Oracle Java Cloud Service and Oracle Java Cloud Service - Virtual Image.

› ORACLE JAVA CLOUD SERVICE

The primary differentiator for Oracle Java Cloud Service is that you can use the self-service portal to easily provision your environment to best suit your requirements. You also have control of the underlying infrastructure and can choose Oracle WebLogic Server, memory, clustering, load balancing, virtual machines, and more. Setting up Oracle Java Cloud Service involves a lot more work and decision-making than Oracle Java Cloud Service - SaaS Extension, but you can get exactly what you require, and also have the freedom to further tweak things if needed in the future.

› ORACLE JAVA CLOUD SERVICE - VIRTUAL IMAGE

Oracle Java Cloud Service - Virtual Image is a similar environment to Oracle Java Cloud Service and also offers control over many aspects of the underlying environment. Oracle Java Cloud Service - Virtual Image is designed for use in development and testing, so it does not support backup and restoration, patching, or scaling. Setting it up is a little simpler because it has fewer prerequisites than does Oracle Java Cloud Service.

Note: The similar names of these cloud solutions can be somewhat confusing. So in the rest of the article, I treat these as three distinct products; notice carefully which one I am referring to in a particular context.

The primary differentiator for Oracle Java Cloud Service is that you can use the self-service portal to easily provision your environment to best suit your requirements.

Getting Started with Oracle Java Cloud Service

Log in to Oracle Cloud by entering your identity domain and login credentials. You will see a dashboard listing all services. As shown in Figure 1, you can use the drop-downs to show only particular services in a particular identity domain. In this case, I have marked a few services as favorites by clicking on the star icon and then only displayed those favorite services.

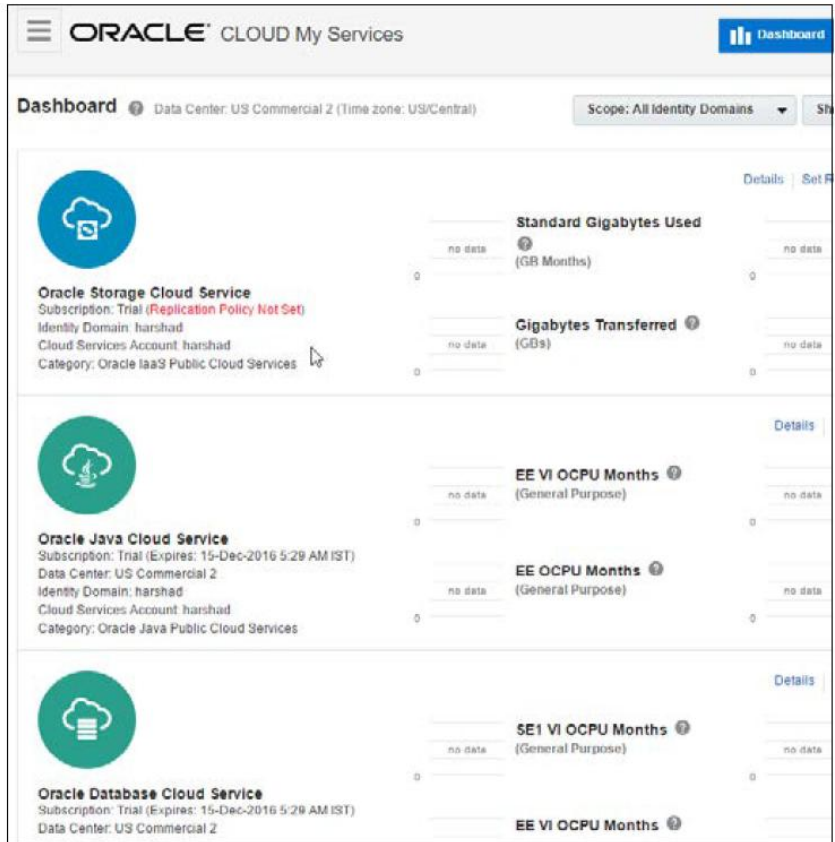


Figure 1. Configuration dashboard

Click the **Service Console** link for Oracle Java Cloud Service, and you will get to a welcome page for Oracle Java Cloud Service. Click the **Services** link, as shown in Figure 2, on that page to set up **prerequisites**.

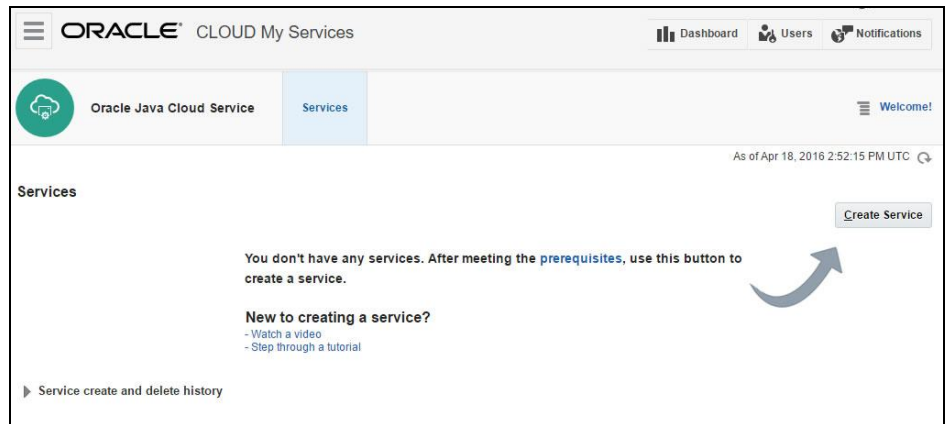


Figure 2. Create Java Cloud Service

The prerequisites are a Secure Shell (SSH) public/private key, an active Oracle Storage Cloud Service, and an active Oracle Database Cloud Service. The Oracle Java Cloud Service trial includes the trial versions of the other cloud services on which it depends, so you don't need to request any additional trials. Let's look at these prerequisites in more detail.

SSH Public/Private Key

Oracle Java Cloud Service requires an SSH public/private key pair for authenticating, so you need to generate one. I used the **PuTTYgen tool** (Windows .exe) to generate the key pair, but there are alternative ways as well. The public key is also required when provisioning Oracle Database Cloud Service and Oracle Java Cloud Service.

Oracle Storage Cloud Service

Oracle Storage Cloud Service offers a secure and scalable storage capability. Oracle Java Cloud Service requires Oracle Storage Cloud Service as it stores backups of service instances to a container in Oracle Storage Cloud Service.

You can see in Figure 1 that **Replication Policy Not Set** is highlighted against Oracle Storage Cloud Service. So first, you need to set a replication policy for Oracle Storage Cloud Service by clicking the **Set Replication Policy** link. For faster data transfers during replication, I recommend that you select the same primary data center to host the Oracle cloud services and Oracle Storage Cloud Service. Legal and security requirements also need to be considered.

As shown in Figure 3, I selected the same primary data center for Oracle Java Cloud Service.

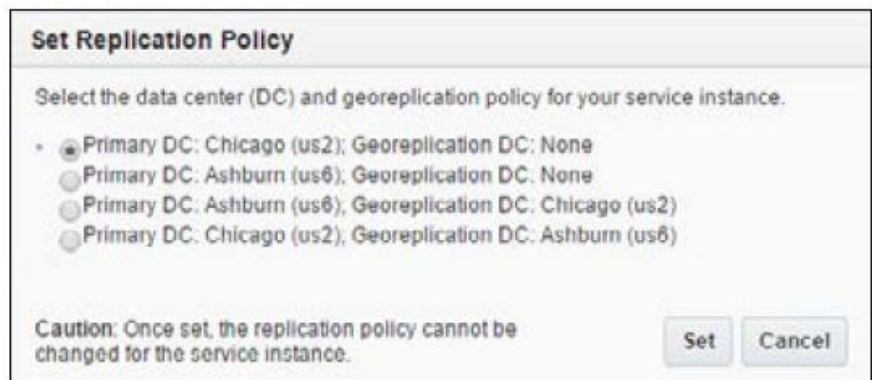


Figure 3. Storage replication policy

Next, you need to create the required Oracle Storage Cloud Service containers for Oracle Java Cloud Service and Oracle Database Cloud Service. These containers **can be created** using the REST API or a Java library.

Note: If you are using the Virtual Image option of both Oracle Java Cloud Service and Oracle Database Cloud Service, you do not need to create the Oracle Storage Cloud Service containers. Because the Virtual Image is a development and testing environment, you have the option of not using Oracle Storage Cloud Service containers for backup and recovery.

Oracle Database Cloud Service

Oracle Java Cloud Service needs Oracle Database Cloud Service to be working. So, before you can create the Oracle Java Cloud Service instance, you need to first create the Oracle Database Cloud Service instance. Click the **Service Console** link for Oracle Database Cloud Service, as shown in Figure 1. On the following welcome page, click the **Services** link. You now get to the page shown in Figure 4.

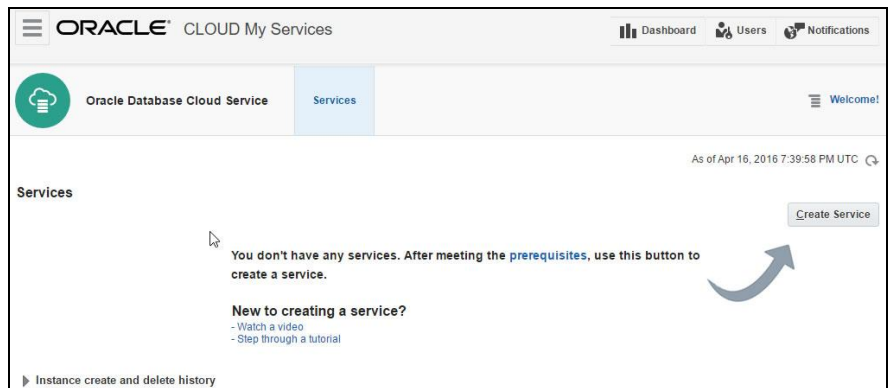


Figure 4. Create Oracle Database Cloud Service

Click **Create Service**. Next create Oracle Database Cloud Service by selecting the options for monthly billing and Oracle Database 12c Enterprise Edition on the **Service Details** page, as shown in Figure 5. I provided the service name `javamagDBWithStorage`, the description, and passwords. I also provided the SSH public key that I created earlier.

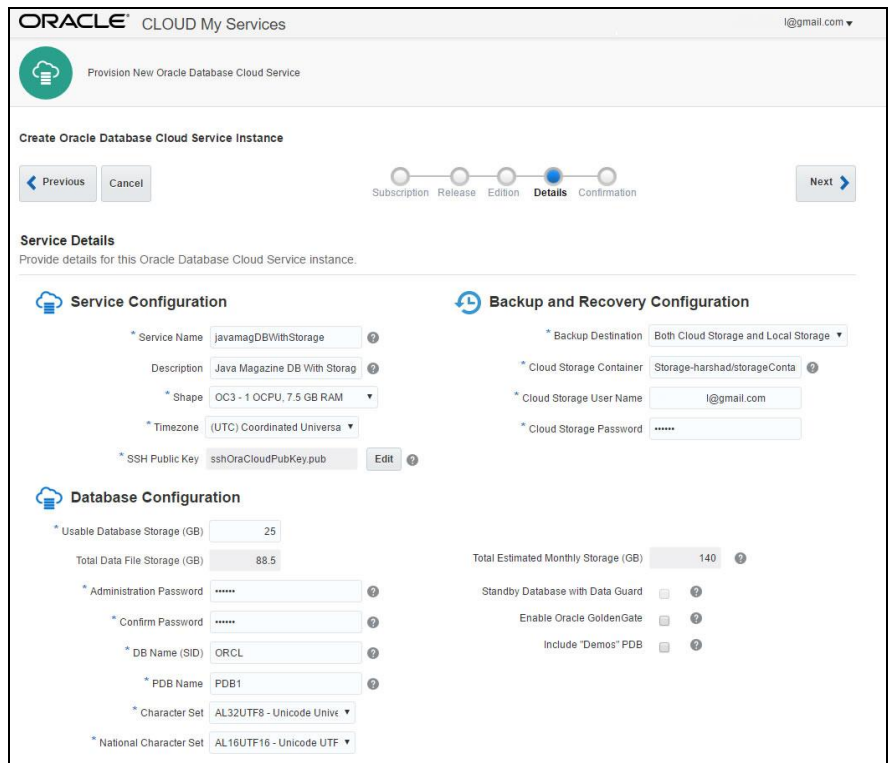


Figure 5. `javamagDBWithStorage`

Note: If you are creating Oracle Database Cloud Service - Virtual Image, you can select the backup destination as none, so that you don't need to also set up an Oracle Storage Cloud Service container for Oracle Database Cloud Service backup and restore.

Select the basic shape with 1 OCPU and 7.5 GB RAM. The configuration can go up to 16 OCPUs and 240 GB RAM. (OCPU here stands for CPU capacity equivalent to one physical core of an indeterminate Intel Xeon processor with hyperthreading enabled.)

Click **Next** and confirm the details. In a few minutes, the `javamagDBWithStorage` database is provisioned and running, as shown in Figure 6.

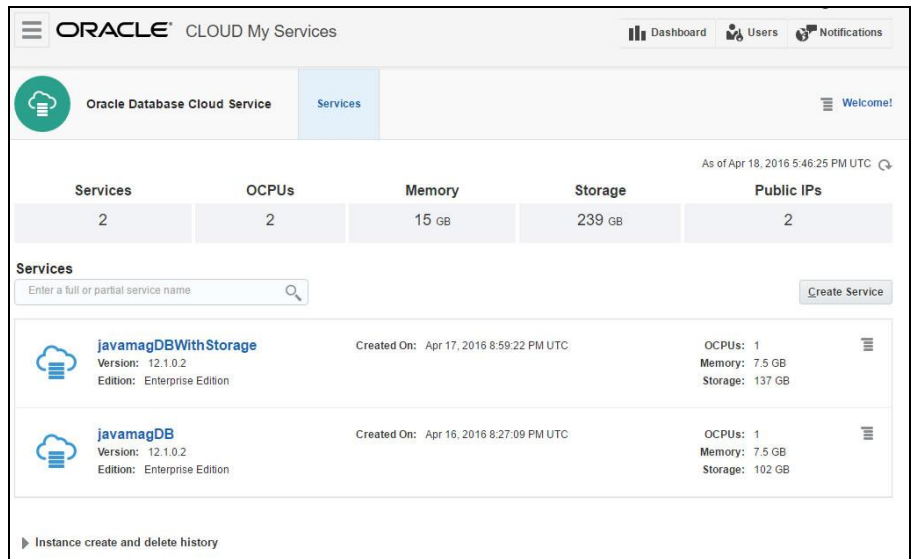


Figure 6. DB cloud instances

Oracle Java Cloud Service Details

Once the `javamagDBWithStorage` database is up and running, head back to the Oracle Java Cloud Service console, as shown in Figure 2, and click **Create Service**. As shown in Figure 7, select Oracle Java Cloud Service. Then select the enterprise edition of the latest available version of Oracle WebLogic Server.

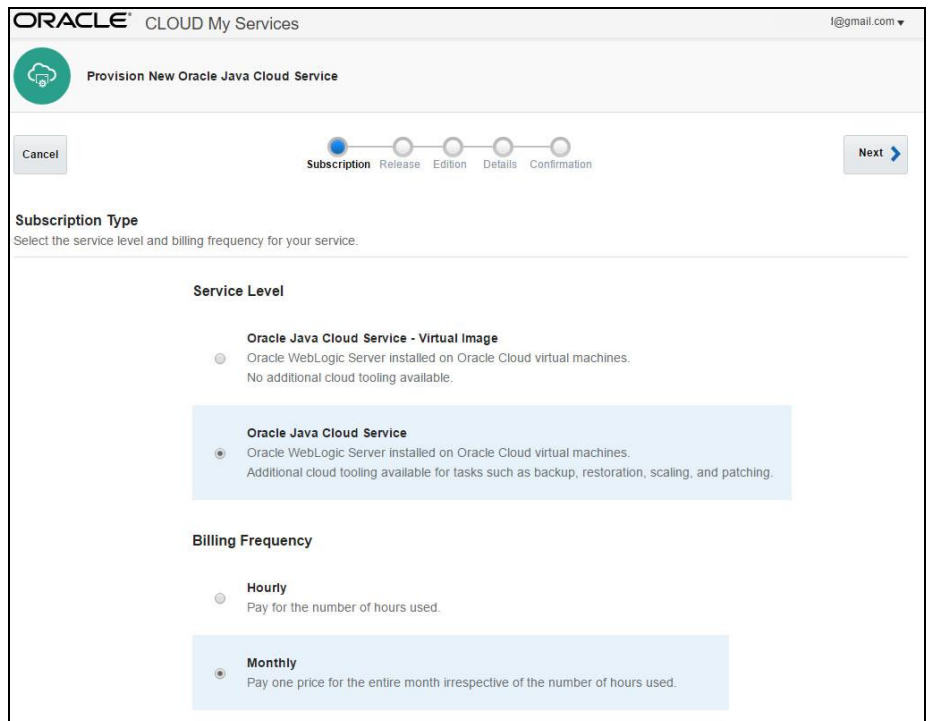


Figure 7. Service level

On the Service Details page, as shown in Figure 8, select the basic shape with 1 OCPU and 7.5 GB RAM, and specify the Oracle Database Cloud Service configuration and the Oracle Storage Cloud Service configuration for **Backup and Recovery Configuration**. Also specify the Oracle WebLogic Server username and password and choose to deploy a sample application.

Using Oracle Java Cloud Service is simple enough and gets you scale and the other benefits that make the cloud a compelling solution.

The screenshot shows the 'Provision New Oracle Java Cloud Service' page in the Oracle Cloud My Services console. The 'Details' step is active, indicated by a blue dot in the progress bar. The page is divided into four main configuration sections:

- Service Configuration:**
 - Service Name: javamagJCS
 - Description: Java Cloud Service For Java Magazine
 - Domain Partitions: 0 (No Partitions)
 - Shape: OC3 - 1 OCPU, 7.5 GB RAM
 - SSH Public Key: sshOraCloudPubKey.pub
- Database Configuration:**
 - Name: javamagDBWithStorage
 - PDB Name: <use default>
 - Administrator User Name: SYSTEM
 - Password: [masked]
- WebLogic:**
 - Username: javamagWL
 - Password: [masked]
 - Confirm Password: [masked]
- Backup and Recovery Configuration:**
 - Cloud Storage Container: Storage-harshad/storageContainerForJCS
 - Cloud Storage User Name: i@gmail.com
 - Password: [masked]

At the bottom, there is a checkbox for 'Deploy Sample Application' which is checked.

Figure 8. Service details

Confirm the service information as shown in the summary in Figure 9. In a few minutes, the Oracle Java Cloud Service instance is provisioned and ready for use. Once that's done, you can use the instance of Oracle WebLogic Server similarly to an on-premises Oracle WebLogic Server. You can also log in to the Oracle WebLogic Administration Console to deploy applications to Oracle Java Cloud Service.

The screenshot shows the 'Confirmation' step of the provisioning process. The progress bar indicates that the 'Confirmation' step is complete. The page displays a summary of the service details:

- Subscription Details:**
 - Service Level: Oracle Java Cloud Service
 - Billing Frequency: Monthly
 - Release: Oracle WebLogic Server 12c (12.2.1.0)
 - Edition: Enterprise Edition
- Service Details:**
 - Service Name: javamagJCS
 - Description: Java Cloud Service For Java Magazine
 - Cluster Size: 1
 - Domain Partitions: 0
 - Shape: OC3 - 1 OCPU, 7.5 GB RAM
 - Key: sshOraCloudPubKey.pub
 - WebLogic Administrator: javamagWL
 - Deploy Sample Application: Yes
- Database Details:**
 - Database: javamagDBWithStorage
 - PDB Name: PDB1
 - Administrator User Name: SYSTEM
- Backup and Recovery Details:**
 - Cloud Storage Container: Storage-harshad/storageContainerForJCS
 - Cloud Storage User Name: i@gmail.com

Figure 9. Confirmation

You now have Oracle Java Cloud Service set up, with an enterprise Oracle WebLogic Server and a database on the cloud. You also have backups and recovery set up on Oracle Storage Cloud Service.

Conclusion

As this article has shown, using Oracle Java Cloud Service is simple enough and gets you scale and the other benefits that make the cloud a compelling proposition, especially for enterprise applications.

The Java cloud space has matured rapidly over the past few years. In its early days, many developers had concerns: “Can the cloud be tweaked to get exactly what I want? Will the cloud bring all the power and functionality that I am used to getting from my on-premises server? Will it be flexible enough for my business?” And so on. In my experience, the newer Oracle Java Cloud Service solutions enable you to do all that and more.

NEXT STEPS

- [Learn more: Oracle Java Cloud](#)
- [Try it: Oracle Managed Cloud Services](#)



HARSHAD OAK IS A JAVA CHAMPION AND THE FOUNDER OF INDICHTHREADS AND RIGHTRIX SOLUTIONS. HE IS THE AUTHOR OF PRO JAKARTA COMMONS (APRESS, 2004) AND HAS WRITTEN SEVERAL BOOKS ON JAVA EE. OAK HAS SPOKEN AT CONFERENCES IN INDIA, THE UNITED STATES, SRI LANKA, THAILAND, AND CHINA.



Source: iStockphoto

This story originally appeared in Oracle's *Java Magazine*, May/June 2016.

CONNECT WITH US

-  blogs.oracle.com/java
-  twitter.com/oraclejavamag
-  oracle.com/technetwork/java/javamagazine

ORACLE

Copyright © 2017, Oracle and/or its affiliates. All rights reserved. Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.