Red Hat OpenShift Developer II DO288 Practice Exam (Sample)

Study Guide

Everything you need from our exam experts!

Table of Contents

Introduction

Preparing for a certification exam can feel overwhelming, but with the right tools, it becomes an opportunity to build confidence, sharpen your skills, and move one step closer to your goals. At Examzify, we believe that effective exam preparation isn't just about memorization, it's about understanding the material, identifying knowledge gaps, and building the test-taking strategies that lead to success.

This guide was designed to help you do exactly that.

Whether you're preparing for a licensing exam, professional certification, or entry-level qualification, this book offers structured practice to reinforce key concepts. You'll find a wide range of multiple-choice questions, each followed by clear explanations to help you understand not just the right answer, but why it's correct.

The content in this guide is based on real-world exam objectives and aligned with the types of questions and topics commonly found on official tests. It's ideal for learners who want to:

- **Practice answering questions under realistic conditions,**
- **Improve accuracy and speed,**
- **Review explanations to strengthen weak areas, and**
- **Approach the exam with greater confidence.**

We recommend using this book not as a stand-alone study tool, but alongside other resources like flashcards, textbooks, or hands-on training. For best results, we recommend working through each question, reflecting on the explanation provided, and revisiting the topics that challenge you most.

Remember: successful test preparation isn't about getting every question right the first time, it's about learning from your mistakes and improving over time. Stay focused, trust the process, and know that every page you turn brings you closer to success.

Let's begin.

How to Use This Guide

This guide is designed to help you study more effectively and approach your exam with confidence. Whether you're reviewing for the first time or doing a final refresh, here’s how to get the most out of your Examzify study guide:

1. Start with a Diagnostic Review

Skim through the questions to get a sense of what you know and what you need to focus on. Your goal is to identify knowledge gaps early.

2. Study in Short, Focused Sessions

Break your study time into manageable blocks (e.g. 30 – 45 minutes). Review a handful of questions, reflect on the explanations.

3. Learn from the Explanations

After answering a question, always read the explanation, even if you got it right. It reinforces key points, corrects misunderstandings, and teaches subtle distinctions between similar answers.

4. Track Your Progress

Use bookmarks or notes (if reading digitally) to mark difficult questions. Revisit these regularly and track improvements over time.

5. Simulate the Real Exam

Once you're comfortable, try taking a full set of questions without pausing. Set a timer and simulate test-day conditions to build confidence and time management skills.

6. Repeat and Review

Don’t just study once, repetition builds retention. Re-attempt questions after a few days and revisit explanations to reinforce learning. Pair this guide with other Examzify tools like flashcards, and digital practice tests to strengthen your preparation across formats.

There’s no single right way to study, but consistent, thoughtful effort always wins. Use this guide flexibly, adapt the tips above to fit your pace and learning style. You've got this!

Questions

1. **To import an image into an image stream named myis from a private registry and confirm the import, which command is appropriate?**

   A. oc import-image myis --confirm --from registry.example.com/myorg/myimage

   B. oc import-image myis --confirm --reference-policy local --from registry.example.com/myorg/myimage

   C. oc import-image myis --from registry.example.com/myorg/myimage --confirm --reference-policy local

   D. oc import-image myis --from registry.example.com/myorg/otherimage --confirm

2. **Which command creates an image stream referencing a container image on quay.io?**

   A. oc import-image <name> --confirm --reference-policy local --from quay.io/<user>/<name>

   B. oc import-image <name> --confirm --reference-policy local --from docker.io/<user>/<name>

   C. oc import-image <name> --from quay.io/<user>/<name> --confirm --reference-policy local

   D. oc import-image <name> --confirm --reference-policy remote --from quay.io/<user>/<name>

3. **To run a Dockerfile with a random user instead of 'wildfly', which sequence of changes should be applied?**

   A. 4, 3, 2, 1

   B. 2, 4, 1, 3

   C. 1, 3, 2, 4

   D. 1, 2, 3, 4 in that order

4. **Which command tags an existing image under a new repository/name and tag in Podman?**

   A. sudo podman tag mysql-custom devops/mysql:snapshot

   B. sudo podman rmi mysql-custom

   C. oc tag mysql-custom devops/mysql:snapshot

   D. docker tag mysql-custom devops/mysql:snapshot

5. Which command creates a secret regtoken from a dockerconfigjson file with the proper type for use by Kubernetes?

A. oc create secret generic regtoken --from-file .dockerconfigjson=${XDG_RUNTIME_DIR}/containers/auth.json --type kubernetes.io/dockerconfigjson

B. oc create secret generic regtoken --from-file .dockerconfigjson=${XDG_RUNTIME_DIR}/containers/auth.json --type kubeconfigjson

C. oc create secret generic regtoken --from-file dockerconfigjson=${XDG_RUNTIME_DIR}/containers/auth.json --type kubernetes.io/dockerconfigjson

D. oc create secret generic regtoken --from-file .dockerconfigjson=${XDG_RUNTIME_DIR}/containers/auth.json --type kubernetes.io/dockersecret

6. Which command deletes all resources with label app=appname?

A. oc delete all -l app=appname

B. oc delete all -l app=appname --ignore-not-found

C. oc delete pods -l app=appname

D. oc delete all

7. Which statement correctly describes how per-environment configuration can be supplied without changing the template body?

A. Template parameters (and optionally environment-specific ConfigMaps).

B. Hard-coded values in the template.

C. Environment-specific ConfigMaps.

D. Separate templates per environment.

8. Which command retrieves logs for a deployment configuration?

A. oc logs dc/<application-name>

B. oc logs deploy/<application-name>

C. oc get logs dc/<application-name>

D. oc logs deploymentconfig/<application-name>

9. Which command connects to an external MySQL database given host, user, password, and database name?

A. mysql -h <db> -p <pass> <name> -u <user>

B. mysql -h <db> -p <pass> <name>

C. psql -h <db> -U <user> -W <pass> <name>

D. mysql -h <db> -u <user> -p <pass> <name>

10. To retrieve the webhook secret for a BuildConfig, which command should you run?

A. oc get bc <name> -o yaml

B. oc get bc <name> -o json

C. oc describe bc <name> --json

D. oc get bc <name> -o wide

Answers

1. A
2. A
3. D
4. A
5. A
6. A
7. A
8. A
9. D
10. B

Explanations

1. To import an image into an image stream named myis from a private registry and confirm the import, which command is appropriate?

A. oc import-image myis --confirm --from registry.example.com/myorg/myimage

B. oc import-image myis --confirm --reference-policy local --from registry.example.com/myorg/myimage

C. oc import-image myis --from registry.example.com/myorg/myimage --confirm --reference-policy local

D. oc import-image myis --from registry.example.com/myorg/otherimage --confirm

Importing an image into an OpenShift ImageStream is done with an import command that explicitly pairs the target stream with the source image and then executes the import. The essential pieces are the name of the image stream you want to populate, the source image in the registry, and the flag that actually performs the import. In this case, you want to populate the image stream named as myis from the image located at registry.example.com/myorg/myimage, and you want the import to run immediately. The appropriate command uses --from to specify the exact source image and --confirm to carry out the import right away. Additional flags like --reference-policy local alter how the imported image is referenced inside the stream, but they aren't required for a basic import and aren't needed here. Using the correct source image and the required confirm flag ensures the import happens into the desired image stream, which is why this option is the best match.

2. Which command creates an image stream referencing a container image on quay.io?

A. oc import-image <name> --confirm --reference-policy local --from quay.io/<user>/<name>

B. oc import-image <name> --confirm --reference-policy local --from docker.io/<user>/<name>

C. oc import-image <name> --from quay.io/<user>/<name> --confirm --reference-policy local

D. oc import-image <name> --confirm --reference-policy remote --from quay.io/<user>/<name>

Importing an image into an OpenShift image stream from a remote registry. The command uses --from to specify the exact image location, here quay.io/<user>/<name>, so OpenShift pulls the image from quay.io and creates an image stream that references that image. The --confirm flag actually performs the import, turning the plan into the actual ImageStream resource in your cluster. The --reference-policy local option controls how the image is referenced after import, keeping a local reference within the cluster. This choice correctly points to the quay.io registry, uses --confirm to execute the import, and sets a valid reference policy. The other options either point to a different registry (docker.io), use an unsupported or irrelevant policy value, or are functionally equivalent in effect but not specifically aligned with the quay.io source requirement.

3. To run a Dockerfile with a random user instead of 'wildfly', which sequence of changes should be applied?

A. 4, 3, 2, 1

B. 2, 4, 1, 3

C. 1, 3, 2, 4

D. 1, 2, 3, 4 in that order

To run a Dockerfile with a random user (non-root) instead of wildfly, you need a non-root user defined, proper file ownership set for that user, and you run the image with that user. The best sequence is to first create the non-root user, then adjust file ownership and permissions so the new user can access what it needs, then switch the effective user to that non-root user, and finally ensure the container's entrypoint runs under that user. This order works because you must have a user in place before you can switch to it, and you must own the application files as that user before you drop privileges. If you tried to change file ownership after already dropping to the new user, you'd lack the necessary privileges; if you tried to run as a non-root user before creating it, there'd be nothing to switch to.

4. Which command tags an existing image under a new repository/name and tag in Podman?

A. sudo podman tag mysql-custom devops/mysql:snapshot

B. sudo podman rmi mysql-custom

C. oc tag mysql-custom devops/mysql:snapshot

D. docker tag mysql-custom devops/mysql:snapshot

Tagging an existing image with a new repository/name and tag in Podman is done with the podman tag command. The correct command uses the current image name as the source (mysql-custom) and creates a new reference (devops/mysql:snapshot) as the target, effectively giving the same image another alias under a different repository and tag. This does not modify the image itself, it just adds another way to refer to it. Other options perform different actions or use different tools: removing an image (rmi) deletes it; OpenShift's oc tag works with image streams in OpenShift rather than Podman; and docker tag uses the Docker CLI, not Podman (you would use podman tag instead in Podman workflows).

5. Which command creates a secret regtoken from a dockerconfigjson file with the proper type for use by Kubernetes?

A. oc create secret generic regtoken --from-file .dockerconfigjson=${XDG_RUNTIME_DIR}/containers/auth.json --type kubernetes.io/dockerconfigjson

B. oc create secret generic regtoken --from-file .dockerconfigjson=${XDG_RUNTIME_DIR}/containers/auth.json --type kubeconfigjson

C. oc create secret generic regtoken --from-file dockerconfigjson=${XDG_RUNTIME_DIR}/containers/auth.json --type kubernetes.io/dockerconfigjson

D. oc create secret generic regtoken --from-file .dockerconfigjson=${XDG_RUNTIME_DIR}/containers/auth.json --type kubernetes.io/dockersecret

The key idea is that Kubernetes expects a secret of type kubernetes.io/dockerconfigjson to hold a Docker config in a data field named .dockerconfigjson. The correct command maps the local file to that exact key and sets the proper type, producing a secret that Kubernetes can use as an image pull secret. Specifically, it uses --from-file with .dockerconfigjson=path/to/file and --type kubernetes.io/dockerconfigjson, which creates the data entry .dockerconfigjson containing the Docker config JSON. The other options fail because they either use an incorrect secret type or omit the required leading dot in the data key, which would create a data field named dockerconfigjson instead of .dockerconfigjson and Kubernetes wouldn't recognize it as a Docker config secret.

6. Which command deletes all resources with label app=appname?

A. oc delete all -l app=appname

B. oc delete all -l app=appname --ignore-not-found

C. oc delete pods -l app=appname

D. oc delete all

The idea is to delete resources by matching a label across all resource types in the namespace. Using a label selector with the all resource type targets every kind of resource (pods, deployments, services, etc.) that carries the specified label, and removes only those matching app=appname. This approach is the correct way to delete all resources with that label in one command. Deleting only pods would miss other resource types with the same label. Omitting the -l filter would remove every resource in the namespace, not just those labeled app=appname. The --ignore-not-found flag only suppresses errors for non-existent resources and doesn't change the scope of what's deleted, so it's not needed for achieving the goal.

7. Which statement correctly describes how per-environment configuration can be supplied without changing the template body?

A. Template parameters (and optionally environment-specific ConfigMaps).

B. Hard-coded values in the template.

C. Environment-specific ConfigMaps.

D. Separate templates per environment.

Per-environment configuration can be supplied by parameterizing the template and passing different values at instantiation time. Template parameters let you customize things like database endpoints, feature flags, replica counts, or image tags without touching the template's body. You can enhance this by using environment-specific ConfigMaps that the deployed resources reference (for example, as environment variables or mounted files); you would select or supply the appropriate ConfigMap for the target environment, again without editing the template itself. This combination keeps a single template reusable across environments while delivering the necessary environment-specific data. Hard-coded values in the template require changing the template for each environment, which isn't desired. Relying on environment-specific ConfigMaps alone can work, but without parameterizing the values you still limit how flexibly the same template can be reused. Using template parameters (with optional ConfigMaps) provides the most flexible, maintainable approach. Separate templates per environment achieves the goal but adds duplication and maintenance overhead.

8. Which command retrieves logs for a deployment configuration?

A. oc logs dc/<application-name>

B. oc logs deploy/<application-name>

C. oc get logs dc/<application-name>

D. oc logs deploymentconfig/<application-name>

In OpenShift, to view the logs associated with a DeploymentConfig you use the logs command with the DeploymentConfig resource type, abbreviated as dc. So you run oc logs dc/<application-name>. This fetches the logs from the pods created by the latest rollout of that deployment configuration. Using the other forms either targets a different resource type (a Deployment) or uses a syntax that isn't for retrieving logs, so they won't return the DC's logs. The short alias dc is the standard and expected way to reference a DeploymentConfig in this context.

9. Which command connects to an external MySQL database given host, user, password, and database name?

A. mysql -h <db> -p <pass> <name> -u <user>

B. mysql -h <db> -p <pass> <name>

C. psql -h <db> -U <user> -W <pass> <name>

D. mysql -h <db> -u <user> -p <pass> <name>

Connecting to a MySQL database from the client requires specifying where to connect, which user to authenticate as, the password, and which database to use. The host is given with -h, the user with -u, the password with -p, and the database name is supplied as the final argument. The command that follows this exact pattern is the clear, conventional way to connect: it sets the host, the user, the password, and then selects the database to use. The other options either point to a different database client (PostgreSQL) or arrange the flags in a nonstandard way, making them unsuitable for connecting to a MySQL database.

10. To retrieve the webhook secret for a BuildConfig, which command should you run?

A. oc get bc <name> -o yaml

B. oc get bc <name> -o json

C. oc describe bc <name> --json

D. oc get bc <name> -o wide

Webhooks secret is stored inside the BuildConfig's trigger configuration, so you need to inspect the full BuildConfig to see it. Outputting the BuildConfig in JSON reveals the entire object, including the nested fields under the triggers where the secret is defined. This makes the secret easy to locate and extract. The wide view shows only a subset of fields and often won't display the webhook details. Describing the BuildConfig gives a human-friendly summary and isn't guaranteed to expose the actual secret in a straightforward way.

Next Steps

Congratulations on reaching the final section of this guide. You've taken a meaningful step toward passing your certification exam and advancing your career.

As you continue preparing, remember that consistent practice, review, and self-reflection are key to success. Make time to revisit difficult topics, simulate exam conditions, and track your progress along the way.

If you need help, have suggestions, or want to share feedback, we'd love to hear from you. Reach out to our team at hello@examzify.com.

Or visit your dedicated course page for more study tools and resources:

https://redhatopenshiftdev2do288.examzify.com

We wish you the very best on your exam journey. You've got this!