MAGENTO-CERTIFIED-PROFESSIONAL-CLOUD-DEVELOPER Q&As

Magento Certified Professional Cloud Developer

Pass Magento MAGENTO-CERTIFIED-PROFESSIONAL-CLOUD-DEVELOPER Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.leads4pass.com/magento-certified-professional-cloud-developer.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Magento Official Exam Center

QUESTION 1

You are upgrading a project to the latest version of Magento Commerce Cloud and part of the process involves a PHP version upgrade. This is now ready to be tested by the QA Team on the Integration environment.

How do you apply the PHP version upgrade?

A. Change the PHP version in the .magento.app.yaml file and re-deploy

B. SSH into the Integration environment and upgrade PHP manually

C. Change the PHP_VERSION configuration in the .magento.env.yaml file and re-deploy

D. Use the magento-cloud CLI tool to update the PHP_VERSION variable

Correct Answer: A

Reference: https://devdocs.magento.com/cloud/project/project-upgrade.html

QUESTION 2

You want to move static content deploy to the build phase of deployments.

Which two actions do you take? (Choose two.)

A. Download and commit app/etc/config.php from production

B. Run ece-tools scd-deploy:set build on production

C. Run ece-tools config:dump on production

D. Use scp to copy app/etc/config.php from local to production

Correct Answer: BD

QUESTION 3

You add a new Composer dependency utilizing composer require. After testing composer install locally, you add the Composer lock and json files and perform a deployment. The build phase fails as Composer is unable to resolve the dependencies.

Why did this happen?

A. The versions of PHP on your local environment and integration environment differ

B. Your commit needs to contain the updated vendor folder

C. The project cache should have been cleared with the magento-cloud:project-build-cache command

D. You did not execute the install locally with --require-dev

Correct Answer: A

QUESTION 4

You added the env:ADMIN_PASSWORD variable in the Project Web UI to change a Magento admin user\\'s password. After deployment you are unable to login using the new password.

What causes this?

A. When you add a variable, the build stage is being skipped, because the codebase has not been changed. You must push a commit to trigger a full deploy

B. Deploy scripts read configuration from the environment variable called $MAGENTO_CLOUD_VARIABLES, which contains an array of variables which were set without the env: prefix

C. Variables which are set using the Project Web UI are not available on the build phase, the admin password variable should be set in the .magento.env.yaml file

D. The sensitive option is required for env:ADMIN_PASSWORD variable

Correct Answer: C

QUESTION 5

After deploying to Staging for the first time with the Fastly module installed, you notice that Fastly is not caching pages. Page caching works properly in a local development environment.

What two steps are required to make Fastly cache pages? (Choose two.)

A. Activate the module by setting Caching Application to Fastly CDN.

B. Connect the Fastly shield using the Shield setting in the Magento admin.

C. Enable the Fastly connection by running magento-cloud fastly:setup.

D. Populate VCL in Fastly by clicking on Upload VCL to Fastly in the Magento admin.

Correct Answer: AD

Reference: https://devdocs.magento.com/cloud/cdn/configure-fastly.html