Magento® An Adobe Company | U

Exercise Solutions

Magento Commerce Cloud for Developers On-Demand Course

magento.com/training/overview

v. 2.2

Magento® U

training@magento.com

u.magento.com

u.magento.com/certification

Contents

NOTE: Exercises are included in the course for each section.

Unit 1. Magento Commerce Cloud Overview

1.1 Introduction to Magento Cloud

Question 1
What are 3 key benefits of the Magento Cloud production architecture? (3 of 5)

A. Guaranteed site uptime
B. Highly resilient and fault tolerant ***(correct)***
C. Allows for zero-downtime upsizing ***(correct)***
D. Scaling of the entire server stack rather than just the application layer ***(correct)***
E. Is limited to 1 Availability Zone

Question 2
What are 3 key benefits of leveraging Fastly for full page cache on Magento Cloud? (3 of 5)

A. Improved response times across the globe ***(correct)***
B. Significantly reduced load at the origin ***(correct)***
C. Improved page load performance across the globe ***(correct)***
D. Automatic translation of all content
E. Improved troubleshooting capabilities

Question 3
What are the 2 key benefits of the read-only core file system? (Pick 2)

A. Core files can be easily overwritten in the production environment
B. Significantly reduces the surfaces available to attack ***(correct)***
C. Improves website performance
D. Guarantees all changes to production environment are tracked ***(correct)***

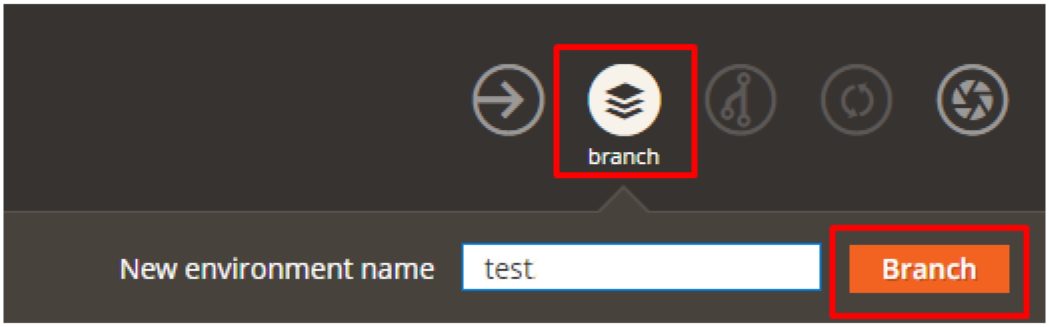
1.2 Getting Started: Installation

If you need reminders for how to install using the wizard, review the steps covered in this section.

1.3 Cloud Admin UI Overview

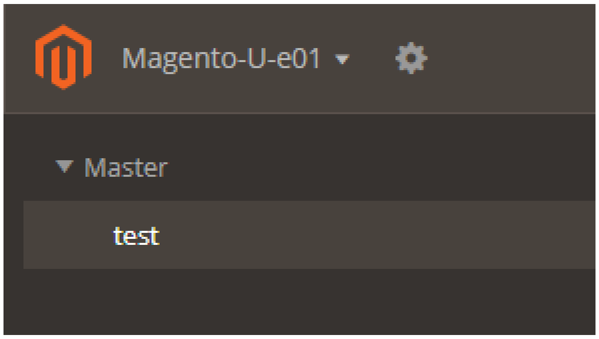
1. **Create the branch using the Cloud admin UI.** *Since this is practice, don't use the following names: master, production, staging, integration*

Press the "branch" button and provide the branch name:



New environment name: test

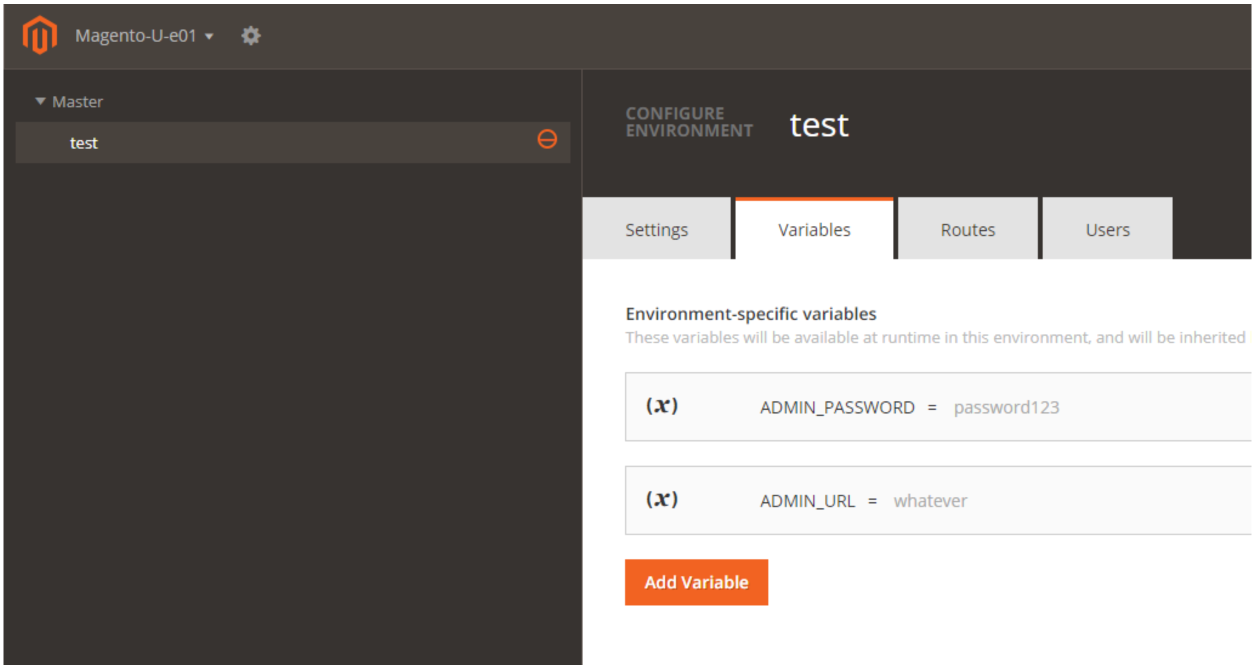
The new environment will be cloned from the Master environment:



2. On the newly created branch, add variables:

- ADMIN_URL = whatever
- ADMIN_PASSWORD = anypasswordyoulike

Navigate to “Configure environment” > Variables and add two variables. The variable add triggers a deployment process (the changes will be applied during deploy).



Environment-specific variables

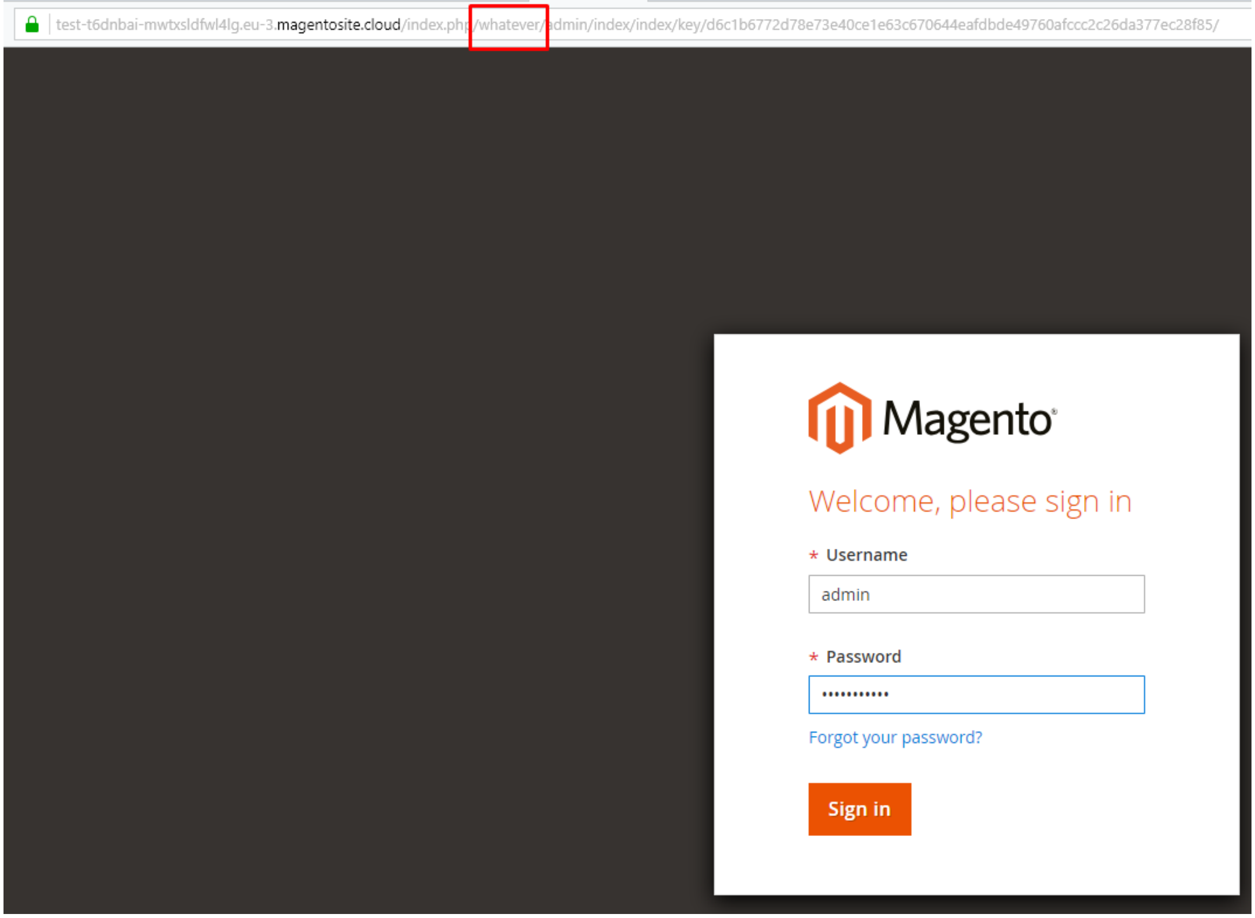
These variables will be available at runtime in this environment, and will be inherited

(x) ADMIN_PASSWORD = password123

(x) ADMIN_URL = whatever

3. Sign in to the Magento admin backend.

Navigate to the site.url/whatever and log in using the password you set in the ADMIN_PASSWORD variable:



test-t6dnbai-mwtxsldfwl4lg.eu-3.magentosite.cloud/index.php/whatever/admin/index/index/key/d6c1b6772d78e73e40ce1e63c670644eafdbde49760afccc2c26da377ec28f85/

Welcome, please sign in

* Username: admin

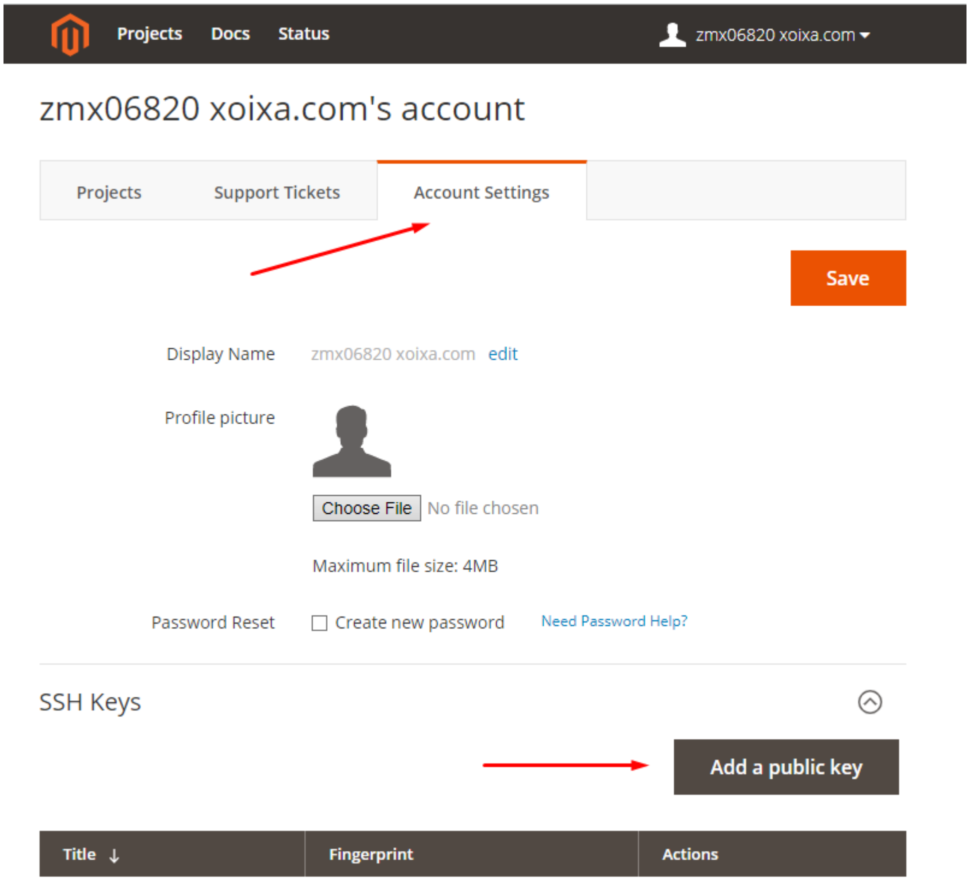
* Password: •••••••••••

For more information, see the dev docs:
https://devdocs.magento.com/guides/v2.2/cloud/env/environment-vars_magento.html

1.4 User Management

1. Go the project settings and add your SSH key

Sign in into https://magento.cloud, go to the **Account settings** tab, and add your public SSH key.



Display Name zmx06820 xoixa.com

Profile picture

No file chosen

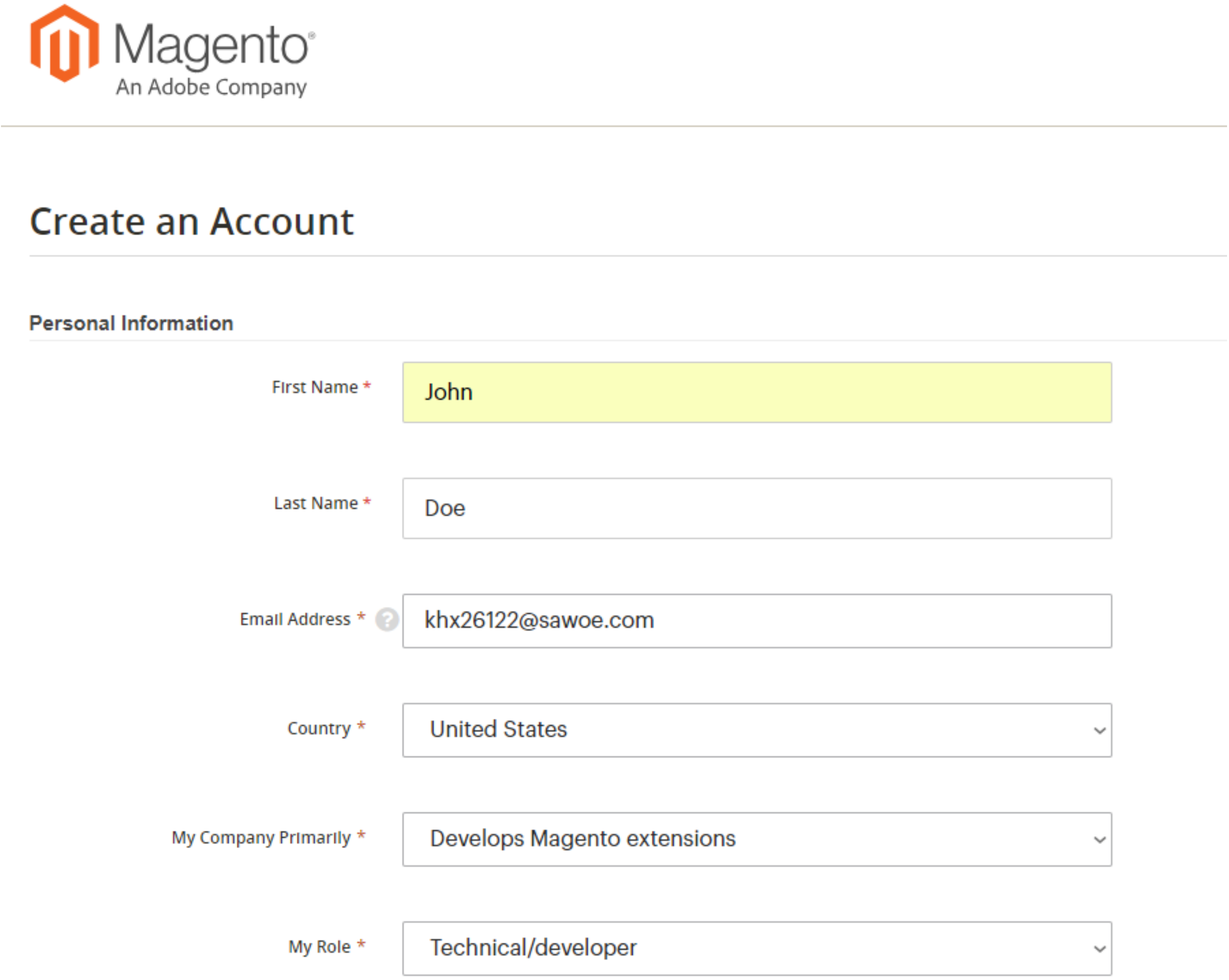
Maximum file size: 4MB

Password Reset □ Create new password Need Password Help?

If you don't have SSH keys, you can generate a new key pair using the following command (Linux and Mac OS X): ssh-keygen

The new public key is saved in the **~/.ssh/id_rsa/pub** file.

2. **Register an additional (buddy) user on the Magento Cloud portal. (You can use a private tab and the "10 minutes mail" service for an email confirmation.)**



Create an Account

Personal Information

First Name * John

Last Name * Doe

Email Address * khx26122@sawoe.com

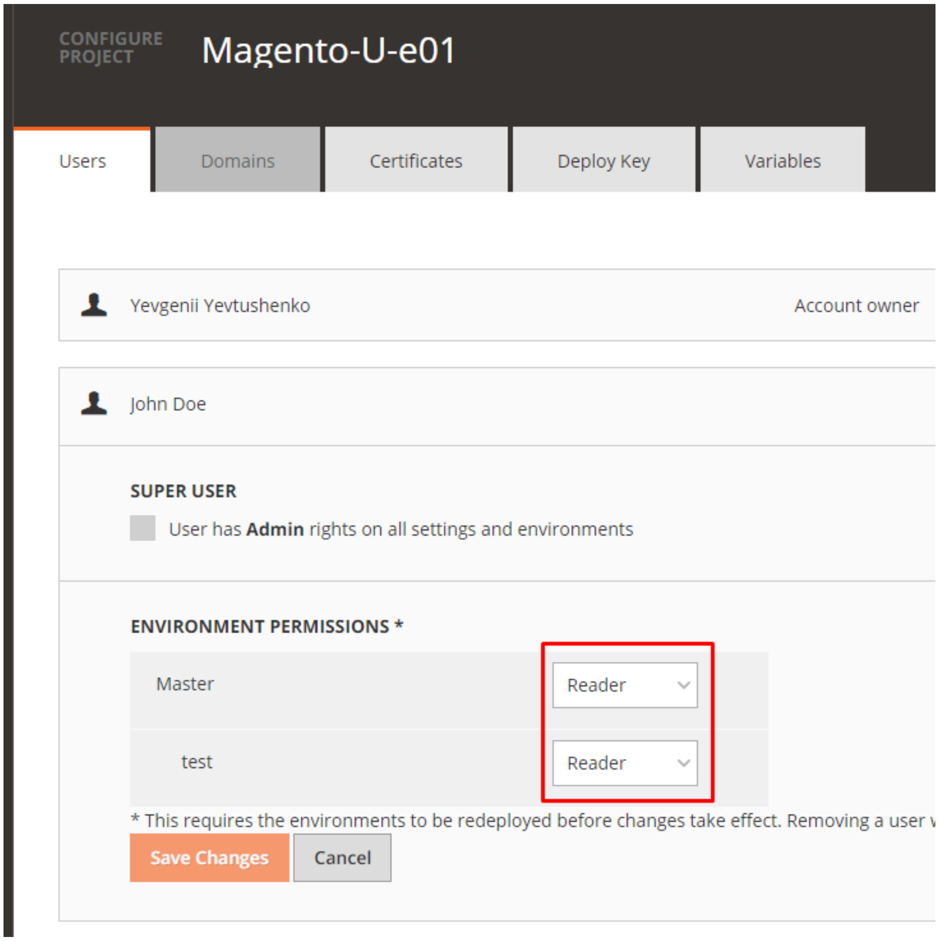
Country * United States

My Company Primarily * Develops Magento extensions

My Role * Technical/developer

3. Add the newly created user to the Master environment with "reader" role

Go to the project level settings and add user by email. Set the "reader" role in the environment permissions section:



Yevgenii Yevtushenko Account owner

John Doe

SUPER USER

□ User has **Admin** rights on all settings and environments

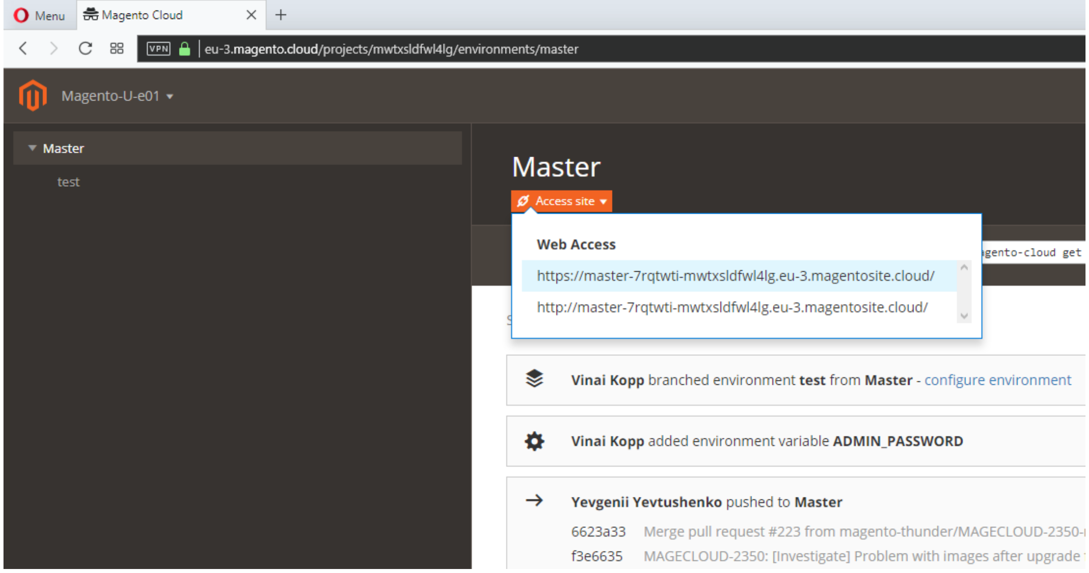
ENVIRONMENT PERMISSIONS *

Master	Reader
test	Reader

* This requires the environments to be redeployed before changes take effect. Removing a user w

4. **Sign in to the Cloud admin UI as the newly created user and observe which functions are available to you.**

You can see the configuration menus. The SSH link as well as some other functions like a snapshot aren't available for the restricted user. Such permission level can be used, for example, for QA engineers.



eu-3.magento.cloud/projects/mwtxsldfwl4lg/environments/master

Master

Web Access

https://master-7rqtwti-mwtxsldfwl4lg.eu-3.magentosite.cloud/

http://master-7rqtwti-mwtxsldfwl4lg.eu-3.magentosite.cloud/

Vinai Kopp branched environment **test** from **Master** - configure environment

Vinai Kopp added environment variable **ADMIN_PASSWORD**

Yevgenii Yevtushenko pushed to **Master**

```
6623a33    Merge pull request #223 from magento-thunder/MAGECLOUD-2350-
f3e6635    MAGECLOUD-2350: [Investigate] Problem with images after upgrade
```

1.5 Onboarding UI

1. What type of users are you able to add through the onboarding UI?
   A. Environment admin ***(correct)***
   B. Magento Admin
   C. Content Manager

2. Which of these resources can you access from the UI?
   A. Environment link ***(correct)***
   B. Help Center ***(correct)***
   C. DNS
   D. Dev docs links ***(correct)***
   E. Google

3. Can users access the Cloud project interface from the onboarding UI?
   A. Yes ***(correct)***
   B. No
   C. Partially

4. Who should be provided shared access under the Magento account?
   A. Any person
   B. Dev team
   C. Project manager
   D. Person who is supposed to file tickets for the Magento account ***(correct)***

Unit 2. Plans & Environments

2.1 Magento Cloud Plans

What are three main advantages of the Pro account over the Starter account?

A. It runs on a shared infrastructure

B. It includes a dedicated Launch Manager ***(correct)***

C. It combines the powerful benefits of Magento Commerce with a cloud hosting agreement for smaller merchants

D. It runs on an isolated and dedicated hardware setup in the underlying Infrastructure-as-a-Service ***(correct)***

E. It has extended core features ***(correct)***

2.2 Environment Types

Which two actions are the most common use cases for the Integration environment?

A. To develop and fix code changes

B. To test functionality of new complete modules ***(correct)***

C. To test environment and Magento settings ***(correct)***

D. To merge to production environment

Unit 3. Local Environment

3.2 Working with Composer

Question:
When using this command:

```
php composer.phar install
```

Which action will occur?

- The install command uses composer.lock file
- The install command uses composer.json file
- The update command uses composer.json file
- The update command uses composer.lock file

Solution:
If a composer.lock file is present, the command will read it and install the packages in the versions listed in the composer.lock file.

If there is no composer.lock file, the command will build the package dependency graph from the requirements listed in the composer.json file, then install an appropriate combination of packages and record the installed versions in a composer.lock file.

The composer update command will build the dependency graph based on the requirements listed in composer.json, install the latest package versions matching the version constraints, and then update the composer.lock file with the new package versions.

So in a nutshell, composer install uses either only composer.lock OR composer.json and composer.lock, and composer update always uses composer.json and composer.lock

3.3 Magento-cloud CLI

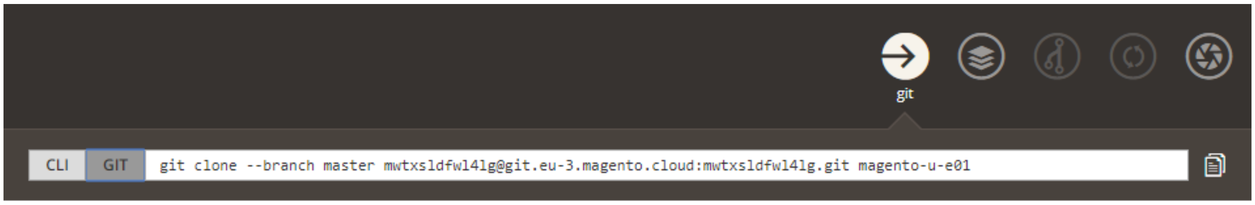
Exercises

1. List environments, users, variables
2. Connect to a database from the local environment
3. Open an SSH tunnel and connect to a MySQL service
4. Build a project on local environment

Solutions

1. List environments, users, variables

1. Clone your project from git onto your local environment using the git link from admin UI:



```
git clone --branch master mwtxsldfwl4lg@git.eu-3.magento.cloud:mwtxsldfwl4lg.git magento-u-e01
```

```
zz@ubuntu:~/cloud$ git clone --branch master mwtxsldfwl4lg@git.eu-
3.magento.cloud:mwtxsldfwl4lg.git magento-u-e01
Cloning into 'magento-u-e01'...
remote: counting objects: 44692, done.
Receiving objects: 100% (44692/44692), 25.74 MiB | 6.81 MiB/s, done.
Resolving deltas: 100% (19505/19505), done.
Checking connectivity... done.
zz@ubuntu:~/cloud$
```

2. Go to the project directory and list environments:

```
zz@ubuntu:~/cloud$ cd magento-u-e01
zz@ubuntu:~/cloud/magento-u-e01$ magento-cloud environment:list
Your environments are:
+---------+--------+--------+
| ID      | Name   | Status |
+---------+--------+--------+
| master* | Master | Active |
| test    | test   | Active |
+---------+--------+--------+
* - Indicates the current environment

Check out a different environment by running magento-cloud checkout [id]
Branch a new environment by running magento-cloud environment:branch [new-name]
Make a snapshot of the current environment by running magento-cloud snapshot:create
```

3. List users:

```
zz@ubuntu:~/cloud/magento-u-e01$ magento-cloud user:list
+--------------------------+---------------------+---------------+------------------------
-------------+
| Email address            | Name                | Project role  | ID
|
+--------------------------+---------------------+---------------+------------------------
-------------+
| yyevtushenko@magento.com | Yevgenii Yevtushenko | admin (owner) | e99f9f6b-e3b5-4623-9c81-
4c8346b96ac6 |
| khx26122@sawoe.com       | John Doe            | admin         | 20d01e3d-85bd-475c-8aa8-
53634118d4f2 |
| vinai@netzarbeiter.com   | Vinai Kopp          | viewer        | 63b9e01e-5c5b-4100-a695-
5b754ec991ad |
+--------------------------+---------------------+---------------+------------------------
-------------+
To view a user's role(s), run: magento-cloud user:get [email]
To change a user's role(s), run: magento-cloud user:add [email]
```

4. List variables:

```
zz@ubuntu:~/cloud/magento-u-e01$ magento-cloud variable:list --level environment
Environment-level variables on the environment master of project Magento-U-e01
(mwtxsldfwl4lg):
+----------------+-------------+-------------+
| Name           | Level       | Value       |
+----------------+-------------+-------------+
| ADMIN_PASSWORD | environment | password123 |
+----------------+-------------+-------------+

To view variable details, run: magento-cloud variable:get [name]
To create a new variable, run: magento-cloud variable:create
To update a variable, run: magento-cloud variable:update [name]
To delete a variable, run: magento-cloud variable:delete [name]
```

Note: This listing shows only environment level variables instead of all of them (environment and project levels) because the composer credentials (sensitive information) are set on the project level so that information is not disclosed here.

2. Connect to a database from the local environment

Connect to a remote database and check the base_url:

```
zz@ubuntu:~/cloud/magento-u-e01$ magento-cloud db:sql
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 49846
Server version: 10.0.35-MariaDB-1~jessie mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [main]> SELECT value FROM core_config_data WHERE path LIKE '%base_url';
+---------------------------------------------------------------+
| value                                                         |
+---------------------------------------------------------------+
| http://master-7rqtwti-mwtxsldfwl4lg.eu-3.magentosite.cloud/  |
| https://master-7rqtwti-mwtxsldfwl4lg.eu-3.magentosite.cloud/ |
+---------------------------------------------------------------+
2 rows in set (0.00 sec)

MariaDB [main]> exit
Bye
Connection to ssh.eu-3.magento.cloud closed.
zz@ubuntu:~/cloud/magento-u-e01$
```

3. Open an SSH tunnel and connect to a MySQL service

```
zz@ubuntu:~/cloud/magento-u-e01$ magento-cloud tunnel:open
Are you sure you want to open SSH tunnel(s) to the master (production) environment? [y/N] y
SSH tunnel opened on port 30000 to relationship: redis
SSH tunnel opened on port 30001 to relationship: database
Logs are written to: /home/zz/.magento-cloud/tunnels.log

List tunnels with: magento-cloud tunnels
View tunnel details with: magento-cloud tunnel:info
Close tunnels with: magento-cloud tunnel:close

Save encoded tunnel details to the MAGENTO_CLOUD_RELATIONSHIPS variable using:
  export MAGENTO_CLOUD_RELATIONSHIPS="$(magento-cloud tunnel:info --encode)"
zz@ubuntu:~/cloud/magento-u-e01$
zz@ubuntu:~/cloud/magento-u-e01$ mysql -h 127.0.0.1 -P 30001
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 49985
Server version: 5.5.5-10.0.35-MariaDB-1~jessie mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| main               |
+--------------------+
2 rows in set (0.04 sec)

mysql> use main

Database changed
mysql> SELECT value FROM core_config_data WHERE path LIKE '%base_url';
+--------------------------------------------------------------+
| value                                                        |
+--------------------------------------------------------------+
| http://master-7rqtwti-mwtxsldfwl4lg.eu-3.magentosite.cloud/  |
| https://master-7rqtwti-mwtxsldfwl4lg.eu-3.magentosite.cloud/ |
+--------------------------------------------------------------+
2 rows in set (0.05 sec)
```

Note: The connection to the database via magento-cloud db:sql command differs from the connection through the SSH tunnel. In the first case the magento-cloud command connects to the cloud environment through SSH and runs the MySQL client remotely. The SSH tunnel makes the remote service available on your local environment as thought the MySQL server is running locally. Then you use your local MySQL client (it should be installed on your computer) for a database connection. Using an SSH tunnel you're able to use your favorite SQL GUI instead of the MySQL CLI client.

4. Build a project on the local environment

Run the build stage on the local environment

```
zz@ubuntu:~/cloud/magento-u-e01$ magento-cloud local:build
Building application mymagento (runtime type: php:7.1)
Found a composer.json file; installing dependencies
  Loading composer repositories with package information
  Installing dependencies (including require-dev) from lock file
    - Installing magento/magento-composer-installer (0.1.13)
      Loading from cache

    - Installing amzn/amazon-pay-sdk-php (3.3.1)
      Loading from cache

    - Installing composer/ca-bundle (1.1.1)
      Loading from cache

    - Installing composer/semver (1.4.2)
      Loading from cache

    - Installing composer/spdx-licenses (1.4.0)
      Loading from cache
…

  Generating optimized autoload files
Running post-build hooks
[2018-07-10 12:25:11] INFO: Starting build.
[2018-07-10 12:25:11] INFO: Verbosity level is not set
[2018-07-10 12:25:11] INFO: Flag .static_content_deploy has already been deleted.
[2018-07-10 12:25:11] INFO: Starting build. (magento/ece-tools version: 2002.0.12,
magento/magento2-base version: 2.2.4)
[2018-07-10 12:25:11] INFO: Validating configuration
[2018-07-10 12:25:11] INFO: End of validation
[2018-07-10 12:25:11] WARNING: Fix configuration with given suggestions:
- File app/etc/config.php not exists (Please run the following commands:
1. bin/magento module:enable --all
2. git add -f app/etc/config.php
3. git commit -m 'Adding config.php'
4. git push)
- No stores/website/locales found in config.php (To speed up the deploy process, please run
the following commands:
1. php ./vendor/bin/ece-tools config:dump
2. git add -f app/etc/config.php
3. git commit -m 'Updating config.php'
4. git push)
- Shared config file is missing module section.
[2018-07-10 12:25:11] INFO: Reconciling installed modules with shared config.
[2018-07-10 12:25:11] INFO: Command: php ./bin/magento module:enable --all
[2018-07-10 12:25:12] INFO: Status: 0
[2018-07-10 12:25:12] INFO: Output: array (
  0 => 'The following modules have been enabled:',
  1 => '- Magento_Store',
  2 => '- Magento_Directory',
  3 => '- Magento_Eav',
  4 => '- Magento_Backend',
  5 => '- Magento_CacheInvalidate',

…

  207 => 'Cache cleared successfully.',
  208 => 'Generated classes cleared successfully. Please run the \'setup:di:compile\'
command to generate classes.',
  209 => 'Info: Some modules might require static view files to be cleared. To do this, run
\'module:enable\' with the --clear-static-content option to clear them.',
)
[2018-07-10 12:25:12] INFO: Applying patches.
[2018-07-10 12:25:12] INFO: File static.php was copied.

…

[2018-07-10 12:26:40] INFO: Command: composer dump-autoload -o
[2018-07-10 12:26:58] INFO: Status: 0
[2018-07-10 12:26:58] INFO: Output: array (
  0 => 'Generating optimized autoload files',
)
[2018-07-10 12:26:58] INFO: Flag .static_content_deploy has already been deleted.
[2018-07-10 12:26:58] NOTICE: Skipping static content deploy: No stores/website/locales
found in config.php
[2018-07-10 12:26:58] INFO: Skipping build-time static content compression because static
content deployment hasn't happened.
[2018-07-10 12:26:58] INFO: Clearing temporary directory.
[2018-07-10 12:26:58] INFO: Copying data to the ./init directory
[2018-07-10 12:26:58] INFO: Flag var/.regenerate has already been deleted.
[2018-07-10 12:26:58] INFO: SCD not performed during build
[2018-07-10 12:26:58] INFO: Copying writable directories to /home/zz/cloud/magento-u-
e01/init/ directory.
[2018-07-10 12:26:58] INFO: Copying /home/zz/cloud/magento-u-e01/app/etc-
>/home/zz/cloud/magento-u-e01/init/app/etc
[2018-07-10 12:26:58] INFO: Copying /home/zz/cloud/magento-u-e01/pub/media-
>/home/zz/cloud/magento-u-e01/init/pub/media
[2018-07-10 12:26:58] NOTICE: Directory /home/zz/cloud/magento-u-e01/var/view_preprocessed
does not exist.
[2018-07-10 12:26:58] INFO: Copying /home/zz/cloud/magento-u-e01/var/log-
>/home/zz/cloud/magento-u-e01/init/var/log
[2018-07-10 12:26:58] INFO: Building completed.
```

```
Build complete for application mymagento
Web root: /home/zz/cloud/magento-u-e01/_www

Cleaning up...
zz@ubuntu:~/cloud/magento-u-e01$
```

When it’s done list files in directory. You can see dev, lib, setup, vendor and some other directories and files which were installed by composer:

```
zz@ubuntu:~/cloud/magento-u-e01$ ls -1
app
auth.json.sample
bin
CHANGELOG.md
composer.json
composer.lock
COPYING.txt
dev
docker
generated
grunt-config.json.sample
Gruntfile.js.sample
index.php
init
lib
LICENSE_AFL.txt
LICENSE_EE.txt
LICENSE.txt
m2-hotfixes
magento-vars.php
nginx.conf.sample
package.json.sample
php.ini
php.ini.sample
phpserver
pub
README_EE.md
README.md
scenario.bkf
setup
update
var
vendor
_www
zz@ubuntu:~/cloud/magento-u-e01$
```

You can build the project locally to ensure that all composer repositories are available and packages can be successfully downloaded and installed, and that patches were successfully applied and setup:di:compile / setup:static-content:deploy didn’t throw errors.

3.3 Exercise Question

What are two key features of the Magento Cloud CLI tool?

A. It helps automatically build and deploy your Cloud projects
B. It helps with Magento management and Git interactions ***(correct)***
C. It allows you to use the hooks to run shell commands during the build, deploy, and post-deploy phases
D. It duplicates the Magento Cloud Admin UI ***(correct)***

3.4 Connecting with GitHub / Bitbucket

Exercise Questions

1. What are the types of integrations covered in this section?
2. What are the benefits of using an integration?
3. How does the GitHub webhook affect the integration?
4. Can you use regular git commands?

Solutions

1. Bitbucket, GitHub, GitLab
2. Integrations provide additional tools that help the development workflow, such as code reviewing (GitHub) or referencing JIRA issues (Atlassian).
3. The GitHub webhook allows a more granular configuration to control the communication between repositories. For example, when you create a branch in GitHub, it becomes an active branch in the Cloud project.
4. Yes. Once the integration is complete and verified, you can use regular git commands. There are some caveats, such as creating branches with Bitbucket. You can always check if the branch is active using magento-cloud environment:list

Unit 4. Cloud Project Files & Structure

4.1 Project Structure

1. When updating Magento Cloud specific configuration files, what folder do you find them under?
   A. .magento **_(correct)_**
   B. update
   C. m2-hoftixes
   D. /etc

2. What are the **only** writeable folders in the application root on a remote cloud project?
   A. m2-hotfixes, var, pub/static, app/etc
   B. update, var, app/etc, m2-hotfixes
   C. var, app/etc, pub/media, pub/static, /tmp **_(correct)_**
   D. .magento, pub/media, app/etc, m2-hotfixes

3. What is the name of the flexible toolset that is used to deploy Magento to Magento Cloud?
   A. magento-cloud-metapackage
   B. ece-tools **_(correct)_**
   C. magento-cloud
   D. Jenkins

4. How would you apply a patch to a package under vendor on Magento Cloud?

   Solution: Place the .patch file into the m2-hotfixes folder in the project root. It will be applied automatically during the build phase (before the code is moved to a read-only mount).

4.2 .magento/routes.yaml

Question
What would you add to the following default routes.yaml file to make it redirect all HTTP traffic with a www prefix to HTTPS?

Solution

1. In your local repository, locate the routes.yaml file in the .magento folder of your cloud project root.

2. Open the file in your preferred text editor and modify the file as shown here:

```
"https://{default}":
  type: upstream
  Upstream: "mymagento:php"

"http://www.{default}/:
  type: redirect
  to: https://{default}
```

4.3 .magento/services.yaml

Exercise
Modify the default services.yaml to increase disk space for MySQL to 8 gigs

Solution

1. In your local repository, locate the services.yaml file in the .magento folder of your cloud project root.
2. Open the file in your preferred text editor and modify the file.
3. Locate the mysql block.
4. Modify the disk directive to read 8192 instead of 2048.

4.4 .magento.app.yaml

Question 1
How would you update your version of PHP?

Solution

1. In your local repository, locate the .magento.app.yaml file in your cloud project root.
2. Open the file in your preferred text editor and modify the file.
3. Locate the type line and change it from php:X.X to php:Y.Y

Question 2
What is the FULL process for adding Elasticsearch 2.5 with 2 gigs of space to an integration environment?
Hint: You need to modify two configuration files

Solution

1. In your local repository, locate the servies.yaml file in the .magento folder of your cloud project root.
2. Open up the file in your preferred text editor and modify the file.
3. Add the new service to .magento/services.yaml

   ```
   newservice:
       type: elasticsearch:2.4
       disk: 2048 # if required by the service
   ```
4. In your local repository, locate the .magento.app.yaml file in your cloud project root.
5. Open the file in your preferred text editor and modify the file.
6. Add the new service under relationships:
   ```
    elasticsearch: elasticsearch
   ```

4.5 .magento.env.yaml

Question 1

Where is the latest sample .magento.env.yaml file stored?

A. .magento

B. application root

C. vendor

D. update

Solution: It's in the root of the repository with a .dist

Question 2

How would you configure an email log with a min_level of info?

Solution

1. In your local repository, locate the .magento.env.yaml file in your Cloud project root.

2. Open the file in your preferred text editor and modify the file.

3. Add the following lines:

```
log:
   email:
     to: email@example.com
     from: email@example.com
     subject: "Log notification from Magento Cloud"
     min_level: "info"
```

Unit 5. Service Configurations

5.1 Service Configurations

Question 1
What would this configuration result in?

```
stage:
      deploy:
        SEARCH_CONFIGURATION:
             engine: elasticsearch
             elasticsearch_server_hostname: custom.host
             elasticsearch_server_port: '9300'
             elasticsearch_index_prefix: magento
             elasticsearch_server_timeout: '15'
```

Solution
This would **REPLACE** the default Magento Cloud Search configuration with the one noted above. If an Elasticsearch host cannot be reached at the noted host, search, catalog browsing, etc. will not work properly.

Question 2
How would you change ***only*** the max_concurrency directive to 30 for the default Magento Cloud Redis instance?

Solution

1. Locate the .magento.env.yaml file in the root of your Magento Cloud project
2. Open the file in your preferred text editor
3. Locate the stage > deploy section of the file (or create if it does not exist).
4. Add the following lines under deploy, indented appropriately:

```
SESSION_CONFIGURATION:
  redis:
    _merge: true
    max_concurrency: 30
```

5.2 Slave Connections

Why are you unable to read from slave connections on Magento Cloud Starter projects?

A. They are not exposed in the relationships section of .magento.app.yaml
B. It is an add-on feature to Magento Cloud Starter
C. Magento Cloud starter has only one node ***(correct)***
D. You need to perform additional configurations in the services.yaml

Unit 6. Deployment

6.1 Deployment Phases

1. What is the limitation for a build script?

   A. A script cannot write to files
   B. No connections to any service ***(correct)***
   C. Must not take more than 30 seconds systime
   D. Only shell scripts can be run

2. Which two actions can be used to retrieve connection data for services installed on the environment?

   A. $_ENV[‘MAGENTO_CLOUD_RELATIONSHIPS’] ***(correct)***
   B. CLI command magento-cloud relationships ***(correct)***
   C. ece-tools command relationships
   D. From the Cloud UI

3. You have installed a new composer package, committed the changed composer.json file but the package is not present in the successful integration build. What is the possible cause?

   A. The updated composer.lock file wasn’t committed ***(correct)***
   B. The package repository was offline
   C. A newer version of the package has been marked as stable and needs to be used
   D. A conflict with another package in the version installed on integration stopped composer from installing the new package

4. During which phase is the site in downtime?

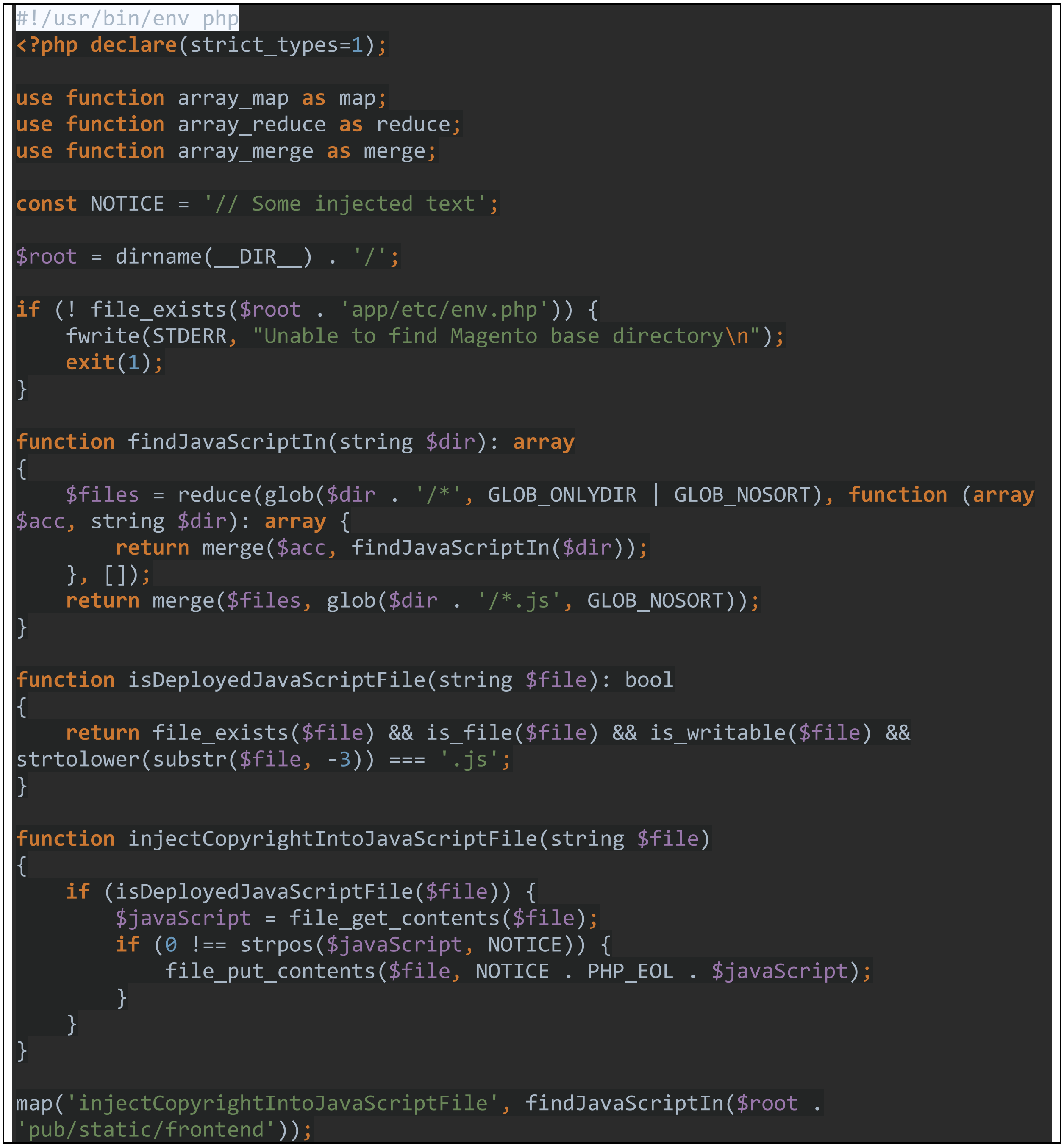
   A. Build
   B. Pre-Deploy
   C. Deploy ***(correct)***
   D. Post-Deploy

6.2 Deployment Scripts

Exercise
Extend the Build script which will add a copyright to the deployed JS files

Solution

1. Create a script like the example shown here.



```php
#!/usr/bin/env php
<?php declare(strict_types=1);

use function array_map as map;
use function array_reduce as reduce;
use function array_merge as merge;

const NOTICE = '// Some injected text';

$root = dirname(__DIR__) . '/';

if (! file_exists($root . 'app/etc/env.php')) {
    fwrite(STDERR, "Unable to find Magento base directory\n");
    exit(1);
}

function findJavaScriptIn(string $dir): array
{
    $files = reduce(glob($dir . '/*', GLOB_ONLYDIR | GLOB_NOSORT), function (array
$acc, string $dir): array {
        return merge($acc, findJavaScriptIn($dir));
    }, []);
    return merge($files, glob($dir . '/*.js', GLOB_NOSORT));
}

function isDeployedJavaScriptFile(string $file): bool
{
    return file_exists($file) && is_file($file) && is_writable($file) &&
strtolower(substr($file, -3)) === '.js';
}

function injectCopyrightIntoJavaScriptFile(string $file)
{
    if (isDeployedJavaScriptFile($file)) {
        $javaScript = file_get_contents($file);
        if (0 !== strpos($javaScript, NOTICE)) {
            file_put_contents($file, NOTICE . PHP_EOL . $javaScript);
        }
    }
}

map('injectCopyrightIntoJavaScriptFile', findJavaScriptIn($root .
'pub/static/frontend'));
```

2. Commit this script to your own repository and change composer.json in the Cloud repository to add your repository (or commit it somewhere in the Cloud project just for training, but it's better to do it the right way).

3. Change the hooks / build section in the .magento.app.yaml file. The script should be between build:generate and build:transfer scripts.

```yaml
hooks:
    # We run build hooks before your application has been packaged.
    build: |
        php ./vendor/bin/ece-tools build:generate
        php ./vendor/testmodule/testscript
        php ./vendor/bin/ece-tools build:transfer
```

4. If the script was added through your own repository, then run composer update.

5. Commit changed files in the cloud repository and push.

6. After deployment, check that the script ran and changes are applied.

Exercise Questions

1. Where can you find information that a patch was applied?

   A. var/log/cloud.log ***(correct)***
   B. /var/cloud.log
   C. Cloud UI deployment history log ***(correct)***
   D. CLI command output if redeployment was triggered by this command ***(correct)***
   E. Open a support ticket to request information

2. Why should you not extend the build script at the end which generates/modifies static content?

   A. The file system is read-only at this point
   B. All generated files were already moved to the init directory by the main script ***(correct)***
   C. You need to add this script in the deploy phase
   D. If the last script is not the ece-tools, it won’t be run

6.3 Script Configurations

Question

If you want to have the maximum verbosity level for the deploy script only, what changes should you make?

A. Set stage/global/VERBOSE_COMMANDS as "" and stage/deploy/VERBOSE_COMMANDS as “-vvv”

B. Set stage/global/VERBOSE_COMMANDS as "" and stage/deploy/VERBOSE_COMMANDS as “-v”

C. Set only stage/build/VERBOSE_COMMANDS as "” as by default all stages have “-vvv”

D. Set only stage/deploy/VERBOSE_COMMANDS as "-vvv" ***(correct)***

Unit 7. Static Content Deployment (SCD)

7.1 Generating SCD During Deployment

Exercise

Generate static content on the Build phase, where:

- For the admin theme, static content should be generated only for en_US
- Nothing should be generated for the luma theme
- Generate en_US with de_DE for the dark theme

Note: The dark theme should be installed on previous modules.

Solution

1. Connect to cloud project through ssh:

```
$ magento-cloud ssh
```

2. Run ece-tools CLI command config:dump to move locale data to files:

```
$ vendor/bin/ece-tools config:dump
```

3. Copy updated file to your local cloud project:

```
rsync <SSH URL>:app/etc/config.php ./app/etc/config.php
```

4. Edit the .magento.env.yaml file (if it does not exist, copy from .magento.env.yaml.dist) by adding necessary configurations for generating SCD:

```
stage:
    global:
        SCD_MATRIX:
            "magento/backend":
                language:
                    - en_US
            "magento/luma":
                language: []
            "Vendor/dark":
                language:
                    - en_US
                    - de_DE
```

5. Commit changed files config.php .magento.env.yaml .
6. Check that SCD was generated on Build phase (from logs).
7. Check that pub/static directory contains necessary data.

Exercise

Then, try the following:

- Skip generating static content for the Blank theme
- Verify by looking in the `pub/static/frontend` directory

7.2 Avoiding SCD During Deployment

What is the cause if after a deployment with SKIP_SCD: true the static content looks broken? Note that on the previous deployment SCD was run on the Build phase with default settings SKIP_SCD: false and STATIC_CONTENT_SYMLINK: true.

A. You have enabled CLEAN_STATIC_FILES which removes all previously generated static files
B. You have broken symlinks in the pub/static directory **(correct)**
C. You have enabled STATIC_CONTENT_SYMLINK which remove files from pub/static and create symlinks to the empty init directory
D. SCD_MATRIX configuration has an empty array for all themes

7.3 Generating Static Content on Demand

Exercise Questions

1. What is the best configuration to minimize the next deployment time when using SCD with the compact strategy?

   A. Use *stage/global/SCD_ON_DEMAND: true* and *stage/global/SCD_STRATEGY: compact*
   B. Move SCD to the Build phase and use *stage/global/SCD_STRATEGY: compact* with *stage/global/SKIP_HTML_MINIFICATION: true* ***(correct)***
   C. Leave SCD on the Deploy phase and use *stage/global/SCD_STRATEGY: compact*
   D. Move SCD to the Build phase and use *stage/global/SKIP_SCD: true* with *stage/global/SKIP_HTML_MINIFICATION: true*

2. What is the best configuration to store previously generated static content for the next deployment? The previous deployment generated static content on the Deploy phase

   A. Use *stage/global/CLEAN_STATIC_FILES: false*
   B. Use *stage/global/SCD_ON_DEMAND: true*
   C. Use *stage/global/SKIP_SCD: true* ***(correct)***
   D. Move SCD to the Build phase and use *stage/global/SKIP_SCD: false* and *stage/global/STATIC_CONTENT_SYMLINK: false*

Unit 8. Development

8.1 Magento Configurations

Exercise Question

Magento has the following data for timezone field (path `general/locale/timezone`)

- DB has records for:
  - ‘default’ scope – America/Los_Angeles
  - ‘stores’ scope and code ‘default’ – America/Chicago
- app/etc/config.php has system/default/general/locale/timezone => Europe/London
- app/etc/env.php has system/default/general/locale/timezone => Europe/Stockholm

What timezone does the store view have with code ‘default’?

A. America/Los_Angeles
B. America/Chicago ***(correct)***
C. Europe/London
D. Europe/Stockholm

8.2 Changing Configurations

Exercise

1. Set the value of design/head/demonotice to 1 for one store view on an integration instance and to 0 for another store view using environment variables via the Magento Cloud UI, .magento.app.yaml, or the magento-cloud CLI command.

2. Then find the key and value in the PHP environment.

Solution

In UI use env:CONFIG__DEFAULT__DESIGN__HEAD__DEMONOTICE
In `yaml:`

```
env:
   variables:
      CONFIG__DEFAULT__DESIGN__HEAD__DEMONITICE: 1
$_ENV['CONFIG__DEFAULT__DESIGN__HEAD__DEMONITICE’] = 1
```

Replace __DEFAULT__ with __STORES__<STORECODE>__ for the store scope setting.

8.3 Changing a Locale

Exercise

Add different locales to your Cloud project that do not have SCD_ON_DEMAND: true

Solution

If you need reminders for how to do this, review the steps covered in this section.

8.4 Adding Extensions

Exercise Question 1
While reviewing a third-party extension you see the following constructor:

```
public function __construct(Config $config, EntityFactory $factory = null)
{
    $this->config = $config;
    $this->factory = $factory ??
        ObjectManager::getInstance()->create(EntityFactory::class);
}
```

What step do you take during the review to ensure that the class is compatible with Magento Commerce Cloud?

A. Flag the class as incompatible because the ObjectManager is used
B. Check that the Entity base class of the factory exists
C. Mark the class as compatible ***(correct)***
D. Change the code so the ObjectManager is also injected as a dependency

Background: The factory is generated during setup:di:compile because it is listed in the constructor signature, even if it is an optional parameter.

Exercise Question 2
While reviewing a third-party extension you see the following constructor:

```
public function __construct(Config $config)
{
    $this->config = $config;
    $this->command = ObjectManager::getInstance()->create(Command\Proxy::class);
}
```

What step do you take during the review to ensure that the class is compatible with Magento Commerce Cloud?

A. Flag the class as incompatible because ObjectManager::create() is used to create the command proxy ***(correct)***
B. Check that the Proxy base class exists
C. Mark the class as compatible
D. Change the code so that the Command\Proxy is passed as a constructor argument

The proxy is not generated because it is used in the constructor body and not the signature of the method.

Note: generated lazy instantiation proxies should not be referred to in source code even in the signature, even though it does technically work. Instead configure it as a type argument in di.xml.

Exercise

Create an extension that stores the date and time for each storefront page that is rendered after a full page cache flush in the file system.

Solution

Refer to the archive “8.4 MagentoU_RenderTime.tgz” for the full module code.

The most important part is the method getFileName() shown here, which shows how to ensure that a file path is within the var/ directory branch.

```php
<?php declare(strict_types=1);

namespace MagentoU\RenderTime\Model;

use Magento\Framework\App\Filesystem\DirectoryList;
use Magento\Framework\Filesystem;

class RenderTime
{
    /**
     * @var Filesystem
     */
    private $filesystem;

    public function __construct(Filesystem $filesystem)
    {
        $this->filesystem = $filesystem;
    }

    public function getFilename(): string
    {
        $var = $this->filesystem->getDirectoryWrite(DirectoryList::VAR_DIR);

        return $var->getAbsolutePath('rendered-at.log');
    }

    public function record(string $pathInfo): void
    {
        $str = $pathInfo . '   ' . date('Y-m-d H:i:s') . PHP_EOL;
        \file_put_contents($this->getFilename(), $str, \FILE_APPEND);
    }

    public function getRecordedTimeFor(string $pathInfo): string
    {
        if (! file_exists($this->getFilename())) {
            return '';
        }
        return $this->findRecord($pathInfo, $this->getFilename());
    }

    private function findRecord(string $pathInfo, string $filename): string
    {
        foreach (explode(PHP_EOL, \file_get_contents($filename)) as $line) {
            if (0 === strpos($line, $pathInfo . '   ')) {
                return rtrim(substr($line, strlen($pathInfo) + 3));
            }
        }

        return '';
    }

    public function clean()
    {
        if (file_exists($this->getFilename())) {
            unlink($this->getFilename());
        }
    }
}
```

8.5 Enabling & Disabling a Module

Exercise 1
Disable the Magento_Version module

Solution

Magento_Version module is a simple module that can be disabled without impact to the overall system.

This module allows you to check the Magento version in a browser – just open http://*your_store_url*/magento_version

Steps to complete the task:

1. Run on the local environment "php ./bin/magento module:disable Magento_Version"
2. Commit and push the config.php file to Cloud
3. Check that after redeploying, the URL http://your_store_url/magento_version returns a 404 error page

Exercise 2
Remove a custom module

Solution
Review this section and the previous section (8.4) if you need a reminder of the steps to complete this exercise.

8.7 Data & Media Synchronization

Exercise

1. SSH into the environment you want to create a database dump from
2. Find the database login

3. Create a database dump
4. Enable SSH agent forwarding
5. Transfer the database dump to another remote environment with an ***rsync*** command to the ***/tmp*** folder
6. SSH into the environment you want to migrate the database into
7. Import the database dump with ***zcat*** command
8. Update Base URLs

Use the Magento Dev Docs as a guide. Dev docs has very clear instructions that could be copied and pasted for the exercise at https://devdocs.magento.com/guides/v2.2/cloud/live/stage-prod-migrate.html

Solution

1. SSH into the environment you want to create a database dump from

```
kbp-lm-677330:~ mmalyshenko$ ssh 2ewbmop5dr3xu-master-7rqtwti--mymagento@ssh.eu-
3.magento.cloud


 __  __                    _          ___ _                _
|  \/  |__ _ __ _ ___ _ _| |_ ___   / __| |___ _  _ __| |
| |\/| / _` / _` / -_) ' \  _/ _ \ | (__| / _ \ || / _` |
|_|  |_\__,_\__, \___|_||_|\__\___/  \___|_\___/\_,_\__,_|
            |___/

 Welcome to Magento Cloud.

 This is environment master-7rqtwti
 of project 2ewbmop5dr3xu.

web@mfiooki7destzwplgejbt66gpe:~$
```

2. Find the database login

```
web@mfiooki7destzwplgejbt66gpe:~$     php -r
'print_r(json_decode(base64_decode($_ENV["MAGENTO_CLOUD_RELATIONSHIPS"]))->database);'
Array
(
    [0] => stdClass Object
        (
            [username] => user
            [scheme] => mysql
            [service] => mysql
            [ip] => 169.254.16.37
            [hostname] => upuqfwksrp7mg4y42tqtygpwem.mysql.service._.magentosite.cloud
            [cluster] => 2ewbmop5dr3xu-master-7rqtwti
            [host] => database.internal
            [rel] => mysql
            [path] => main
            [query] => stdClass Object
```

```
                (
                    [is_master] => 1
                )

            [password] =>
            [port] => 3306
        )

)
web@mfiooki7destzwplgejbt66gpe:~$
```

3. Create a database dump

```
web@mfiooki7destzwplgejbt66gpe:~$ mysqldump -h database.internal --single-transaction --
triggers main | gzip - > /tmp/database.sql.gz
web@mfiooki7destzwplgejbt66gpe:~$ ls /tmp/
database.sql.gz                                 zend_cache---
Zend_LocaleC_en_GB_scientificnumber_
log                                             zend_cache---Zend_LocaleL_en_GB_symbols_
nginx                                           zend_cache---internal-metadatas---
Zend_LocaleC_en_001_parentlocale_
sessions                                        zend_cache---internal-metadatas---
Zend_LocaleC_en_GB_decimalnumber_
zend_cache---Zend_LocaleC_en_001_parentlocale_  zend_cache---internal-metadatas---
Zend_LocaleC_en_GB_parentlocale_
zend_cache---Zend_LocaleC_en_GB_decimalnumber_  zend_cache---internal-metadatas---
Zend_LocaleC_en_GB_scientificnumber_
zend_cache---Zend_LocaleC_en_GB_parentlocale_   zend_cache---internal-metadatas---
Zend_LocaleL_en_GB_symbols_
web@mfiooki7destzwplgejbt66gpe:~$
```

4. Enable SSH agent forwarding to rsync data between branches

On your local computer:

```
kbp-lm-677330:~ mmalyshenko$ vi ~/.ssh/config
Host *
 AddKeysToAgent yes
 UseKeychain yes
 IdentityFile ~/.ssh/id_rsa
 ForwardAgent yes
```

Run SSH Agent:

```
kbp-lm-677330:~ mmalyshenko$ ssh-agent
```

SSH back to your cluster with
`-A parameter`

```
kbp-lm-677330:~ mmalyshenko$ ssh 2ewbmop5dr3xu-master-7rqtwti--mymagento@ssh.eu-
3.magento.cloud -A


 __  __                   _          ___ _                 _
|  \/  |__ _ __ _ ___ _ _| |_ ___   / __| |___ _  _ __| |
| |\/| / _` / _` / -_) ' \  _/ _ \ | (__| / _ \ || / _` |
|_|  |_\__,_\__, \___|_||_\__\___/  \___|_\___/\_,_\__,_|
            |___/

 Welcome to Magento Cloud.

 This is environment master-7rqtwti
 of project 2ewbmop5dr3xu.
```

5. Transfer the database dump to another remote environment with an ***rsync*** command to the ***/tmp*** folder

```
web@mfiooki7destzwplgejbt66gpe:~$ rsync -azvP /tmp/database.sql.gz 2ewbmop5dr3xu-staging-
vdt2zeq--mymagento@ssh.eu-3.magento.cloud:/tmp
Could not create directory '/app/.ssh'.
Warning: Permanently added 'ssh.eu-3.magento.cloud,34.251.110.37' (RSA) to the list of known
hosts.
sending incremental file list
database.sql.gz
      2,835,271 100%   22.27MB/s    0:00:00 (xfr#1, to-chk=0/1)

sent 2,405,535 bytes  received 35 bytes  1,603,713.33 bytes/sec
total size is 2,835,271  speedup is 1.18
```

6. SSH into the environment you want to migrate the database into

```
web@mfiooki7destzwplgejbt66gpe:~$ ssh 2ewbmop5dr3xu-staging-vdt2zeq--mymagento@ssh.eu-
3.magento.cloud
Could not create directory '/app/.ssh'.
Warning: Permanently added 'ssh.eu-3.magento.cloud,34.240.75.192' (RSA) to the list of
known hosts.


 __  __                   _          ___ _                 _
|  \/  |__ _ __ _ ___ _ _| |_ ___   / __| |___ _  _ __| |
| |\/| / _` / _` / -_) ' \  _/ _ \ | (__| / _ \ || / _` |
|_|  |_\__,_\__, \___|_||_\__\___/  \___|_\___/\_,_\__,_|
            |___/

 Welcome to Magento Cloud.

 This is environment staging-vdt2zeq
 of project 2ewbmop5dr3xu.
```

7. Import the database dump with ***zcat*** command

```
web@owgpgh4nwz2bnkc2wwfqrwvxzi:~$ zcat /tmp/database.sql.gz | mysql -h database.internal -u
user main
```

check front end;

8. **Update your base URL and clean cache**

```
mysql -h database.internal main
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A


Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 47593
Server version: 10.0.31-MariaDB-1~jessie mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [main]>
MariaDB [main]> select * from core_config_data where path like '%base%';
+-----------+---------+----------+-------------------------------+------------------------+
| config_id | scope   | scope_id | path                          | value                  |
+-----------+---------+----------+-------------------------------+------------------------+
|         1 | default |        0 | web/unsecure/base_url         | http://clownpnd.work/  |
|         2 | default |        0 | web/secure/base_url           | https://clownpnd.work/ |
|         6 | default |        0 | currency/options/base         | USD                    |
|       181 | default |        0 | web/unsecure/base_static_url  | NULL                   |
|       182 | default |        0 | web/unsecure/base_media_url   | NULL                   |
|       183 | default |        0 | web/secure/base_static_url    | NULL                   |
|       184 | default |        0 | web/secure/base_media_url     | NULL                   |
+-----------+---------+----------+-------------------------------+------------------------+
7 rows in set (0.01 sec)

MariaDB [main]> UPDATE core_config_data SET value = 'http://staging-vdt2zeq-
2ewbmop5dr3xu.eu-3.magentosite.cloud/' WHERE path LIKE 'web/unsecure/base_url';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [main]> UPDATE core_config_data SET value = 'https://staging-vdt2zeq-
2ewbmop5dr3xu.eu-3.magentosite.cloud/' WHERE path LIKE 'web/secure/base_url';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [main]> select * from core_config_data where path like '%base%';
```

```
+-----------+---------+----------+------------------------------+--------------------------
-------------------------------------+
| config_id | scope   | scope_id | path                         | value
|
+-----------+---------+----------+------------------------------+--------------------------
-------------------------------------+
|         1 | default |        0 | web/unsecure/base_url        | http://staging-vdt2zeq-
2ewbmop5dr3xu.eu-3.magentosite.cloud/  |
|         2 | default |        0 | web/secure/base_url          | https://staging-vdt2zeq-
2ewbmop5dr3xu.eu-3.magentosite.cloud/ |
|         6 | default |        0 | currency/options/base        | USD
|
|       181 | default |        0 | web/unsecure/base_static_url | NULL
|
|       182 | default |        0 | web/unsecure/base_media_url  | NULL
|
|       183 | default |        0 | web/secure/base_static_url   | NULL
|
|       184 | default |        0 | web/secure/base_media_url    | NULL
|
+-----------+---------+----------+------------------------------+--------------------------
-------------------------------------+
7 rows in set (0.00 sec)

MariaDB [main]> exit
Bye
web@owgpgh4nwz2bnkc2wwfqrwvxzi:~$ php bin/magento cache:clean
Cleaned cache types:
config
layout
block_html
collections
reflection
db_ddl
eav
customer_notification
full_page
config_integration
config_integration_api
target_rule
translate
config_webservice
web@owgpgh4nwz2bnkc2wwfqrwvxzi:~$
```

8.8 Variables

Exercise Question

What are two cloud specific variables?

A. BLACKFIRE_SERVER_ID
B. MAGENTO_CLOUD_APPLICATION
C. MAGENTO_CLOUD_RELATIONSHIPS ***(correct)***
D. COMPOSER_AUTH
E. MAGENTO_CLOUD_VARIABLES ***(correct)***

Unit 9. Integrations

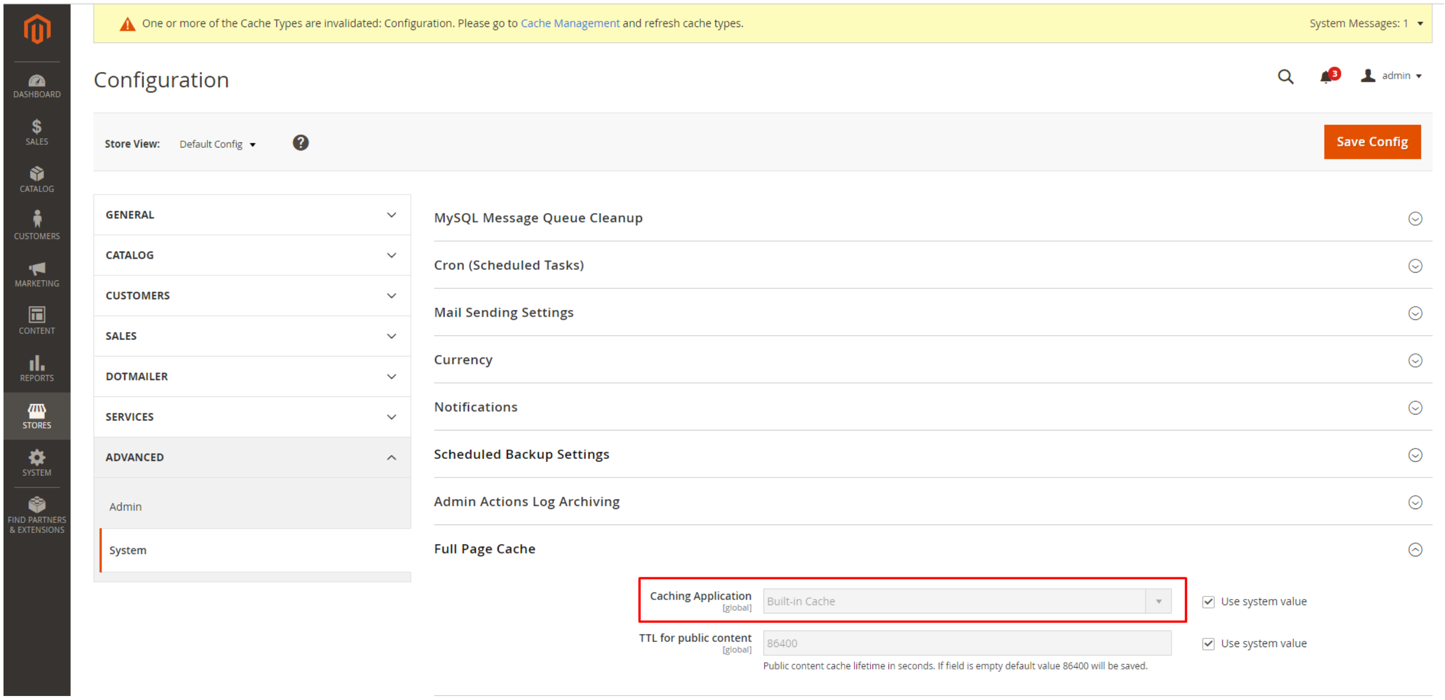
9.1 Fastly

Exercise

1. Switch Magento caching application to Fastly
2. Locate the button “Upload VCL to Fastly”
3. Locate the “Admin path timeout” setting
4. Change Fastly wait timeout for the Magento Admin backend

Solution

1. Log in into the Magento Admin backend and navigate to
   Stores > Configuration > Advanced > System > Full Page Cache


One or more of the Cache Types are invalidated: Configuration. Please go to Cache Management and refresh cache types.

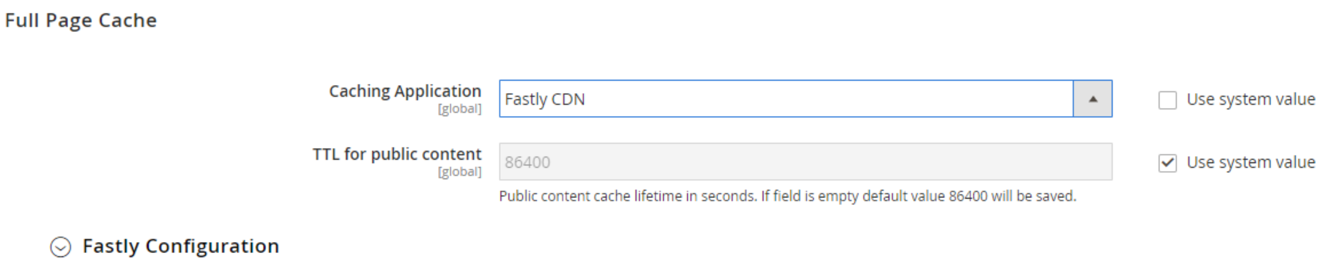
Store View: Default Config

Caching Application [global]: Built-in Cache ☑ Use system value

TTL for public content [global]: 86400 ☑ Use system value

Public content cache lifetime in seconds. If field is empty default value 86400 will be saved.

2. Switch “**Caching Application**” from Built-in to **Fastly CDN**



Full Page Cache

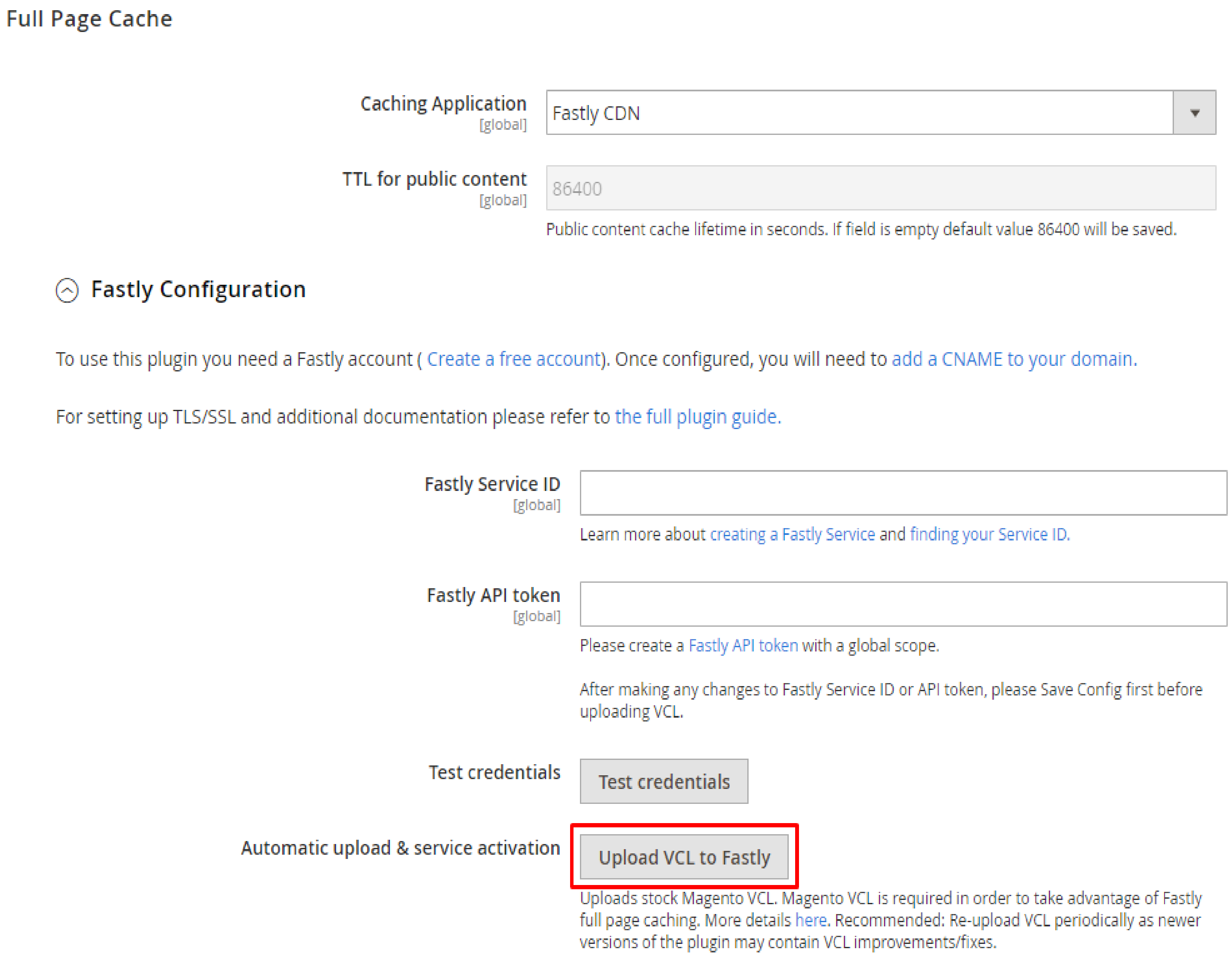
Caching Application [global]: Fastly CDN ☐ Use system value

TTL for public content [global]: 86400 ☑ Use system value

Public content cache lifetime in seconds. If field is empty default value 86400 will be saved.

Fastly Configuration

3. Locate the button **Upload VCL to Fastly**



Full Page Cache

Caching Application [global]: Fastly CDN

TTL for public content [global]: 86400

Public content cache lifetime in seconds. If field is empty default value 86400 will be saved.

Fastly Configuration

To use this plugin you need a Fastly account (Create a free account). Once configured, you will need to add a CNAME to your domain.

For setting up TLS/SSL and additional documentation please refer to the full plugin guide.

Fastly Service ID [global]:

Learn more about creating a Fastly Service and finding your Service ID.

Fastly API token [global]:

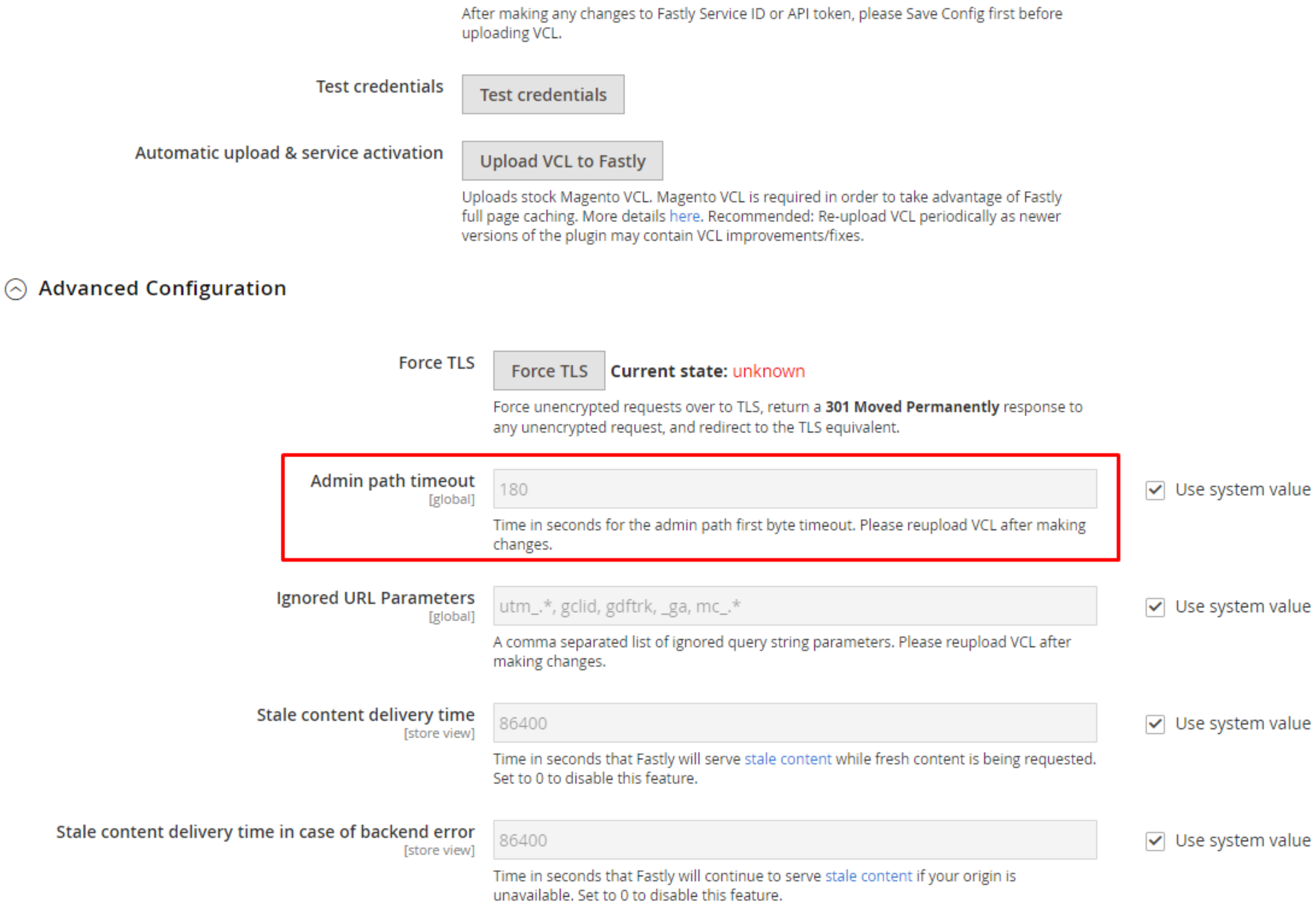
Please create a Fastly API token with a global scope.

After making any changes to Fastly Service ID or API token, please Save Config first before uploading VCL.

Uploads stock Magento VCL. Magento VCL is required in order to take advantage of Fastly full page caching. More details here. Recommended: Re-upload VCL periodically as newer versions of the plugin may contain VCL improvements/fixes.

4. Locate the **“Admin path timeout”** setting

Expand “Advanced Configuration”



After making any changes to Fastly Service ID or API token, please Save Config first before uploading VCL.

Test credentials — Test credentials

Automatic upload & service activation — Upload VCL to Fastly

Uploads stock Magento VCL. Magento VCL is required in order to take advantage of Fastly full page caching. More details here. Recommended: Re-upload VCL periodically as newer versions of the plugin may contain VCL improvements/fixes.

Advanced Configuration

Force TLS — Force TLS **Current state:** unknown

Force unencrypted requests over to TLS, return a **301 Moved Permanently** response to any unencrypted request, and redirect to the TLS equivalent.

Admin path timeout [global]: 180 ☑ Use system value

Time in seconds for the admin path first byte timeout. Please reupload VCL after making changes.

Ignored URL Parameters [global]: utm_.*, gclid, gdftrk, _ga, mc_.* ☑ Use system value

A comma separated list of ignored query string parameters. Please reupload VCL after making changes.

Stale content delivery time [store view]: 86400 ☑ Use system value

Time in seconds that Fastly will serve stale content while fresh content is being requested. Set to 0 to disable this feature.

Stale content delivery time in case of backend error [store view]: 86400 ☑ Use system value

Time in seconds that Fastly will continue to serve stale content if your origin is unavailable. Set to 0 to disable this feature.

9.2 – 9.3 New Relic APM & 9.3 Blackfire

Exercise

1. Add *newrelic* and *blackfire* services to the .magento.app.yaml
2. Add a license for the *newrelic* service
3. Add `project:variable:set env:BLACKFIRE_SERVER_ID`
4. Add `project:variable:set env:BLACKFIRE_SERVER_TOKEN`
5. Add the Blackfire.io integration
6. Install a browser Blackfire plugin
7. Open your website
8. Profile it using a plugin
9. Open a Blackfire page and check for profiling results

Solution

Note: Magento CLI must be installed.

1. Add *newrelic* and *blackfire* services to the .magento.app.yaml

   a. Check out the code

   b. Edit .magento.app.yaml

   c. Add appropriate services, so .magento.app.yaml should contain:

   ```
   runtime:
           extensions:
       -   Newrelic
       -   blackfire
   ```

2. Add a license for the *newrelic* service

   Run the magento CLI command

   ```
   magento-cloud variable:set php:newrelic.license <your-new-relic-license-key>
   ```

   - You can find the license key under the Magento Cloud UI profile section.
   - Find Server Tokens and Server ID at their blackfire account integration with Magento

3. Add `project:variable:set env:BLACKFIRE_SERVER_ID`

   Run the magento CLI command

   ```
   magento-cloud --project=‘<ID>' project:variable:set env:BLACKFIRE_SERVER_ID <token>
   ```

4. Add project:variable:set env:BLACKFIRE_SERVER_TOKEN

Run the magento CLI command

```
magento-cloud --project=’<ID>' project:variable:set env:BLACKFIRE_SERVER_TOKEN <token>
```

5. Add the Blackfire.io integration

```
magento-cloud integration:add \ --project=‘<ID>' \ --type=webhook \ --
url='https://<ID>:<id>@blackfire.io/api/v1/build/env/<id’s>/magento_cloud'
```

All the Blackfire commands should be copied and pasted from the Blackfire setting because they contain corresponding project IDs.

Users can check in their Cloud UI that variables for New Relic and Blackfire have been added and listed appropriately.

6. Install a browser Blackfire plugin
7. Open your website
8. Profile it using a plugin
9. Open a Blackfire page and check for profiling results

Unit 10. Troubleshooting

10.1 Logs

Exercise Questions

1. Where are application exceptions logged?
2. Which logs do you to check when you're getting 503 errors on the storefront?
3. Where is the MySQL slow log located on the integration and Pro environments?

Solutions

1. The Magento exception.log is located in the var/log/ directory, which is located on the application directory (docroot). The same path is valid for all environment types.

2. When you're getting 503 errors, check the PHP and Nginx error log. If a critical PHP error happened (like failure of the connection to MySQL or Redis) it will be there. Then check the application reports folder var/report/ to see if there are critical application issues (like "no such entity with id=xxx"). Then check if there are any critical errors in the var/log/ folder (look for exception.log, support_report.log).

   If there's no critical error, the maintenance mode might be enabled (in this case the Magento application generates 503 HTTP responses without any errors in logs). So, check the following:

   - `/var/log/error.log`
   - `var/report/`
   - `var/log/exception.log`
   - `php bin/magento maintenance:status`

3. The MySQL slow log isn't available on the integration environment due to its nature (you don't have SSH access to a container where MySQL is running). On Pro environments the MySQL slow log is located in the /var/log/mysql/ directory.

10.2 Snapshots. Backups, & Restores

Question 1
How often are Snapshots taken for Production and Staging environments on Commerce Pro?

A. Every 12 hours
B. Every 6 hours ***(correct)***
C. Every 4 hours
D. Every 2 hours

Question 2
How are snapshots initiated for the Production and Staging environments on Commerce Pro?

A. Automatic for both Pro and Stater
B. Manual for both Pro and Starter

C. Automatic for Pro and manual for Starter **(correct)**
D. Manual for Pro and Automatic for Starter

Question 3
What is included in Snapshots of the Production and Staging environments on Magento Commerce Pro?

A. Files only
B. Files and database only
C. Everything on the EBS volume, including the persistent data from all running services **(correct)**
D. Everything on the EBS volume, except the persistent data from all running services

Question 4
When backups are transferred to S3, which region are they stored in be default?

A. The region furthest from the customers production region
B. The same AWS region as the production environment **(correct)**
C. Any region other than the same AWS region as the production environment
D. All of the AWS regions

Question 5
Are backups of the production environment encrypted and why?

A. No, backups are not encrypted because S3 storage is not encrypted by default
B. Yes, backups are encrypted because S3 storage is encrypted by default
C. No, backups are not encrypted because the production EBS volumes are not encrypted
D. Yes, backups are encrypted because the EBS volumes are encrypted **(correct)**

Question 6
How are restores initiated in the Production environment for Pro and Starter?

A. Production restores can be initiated via the Cloud UI/CLI for both Starter and Pro
B. Production restores are initiated via a support ticket for both Starter and Pro
C. Production restores for Pro are initiated via a support ticket and production restores for Starter are initiated via the Cloud UI/CLI **(correct)**
D. Production restores for Pro are initiated via the Cloud UI/CLI and production restores for Starter are initiated via a support ticket

10.3 Debugging (Xdebug)

Question
What's the preferred method of debugging on Magento Cloud?

A. Committing var_dump()s and die()s into your code
B. SSH'ing into the server and manually inserting var_dump()s and die()s into your code
C. Creating an extension and installing it via Composer which injects var_dump()s and die()s into your code
D. Using Xdebug **(correct)**

10.4 Fixes in Patches

Exercise 1
Apply custom patch to Cloud

Solution

The sample patch will activate MySQL logging. After you apply the patch you can find a new log file ./var/log/db.log

1. By default "**m2-hotfixes**" directory is in your project root directory on the local environment
   ```
   <Magento project root dir>/m2-hotfixes
   ```

2. Copy the patch files to that directory

3. Add files to git and commit changes
   ```
   git add -A && git commit -m "Apply patch"
   ```

4. Push changes to Cloud
   ```
   git push origin <branch name>
   ```

5. Clean the Magento cache after successful deploy
   ```
   php <Magento project root dir>/bin/magento cache:clean
   ```

6. To check that the patch is applied, log in to the Magento Admin and check that the new log file "db.log" appears in ./var/log/ directory

Question 2
Find general patches on local environment
Solution

1. Open the local environment root directory in the CLI

2. Enter the following command to get latest pack of general patches
   ```
   composer update magento/ece-tools
   ```

3. See available general patches in the ./vendor/magento/ece-tools/patches folder

10.5 Branch Synchronization

Question 1 (2 of 4)
You have attempted to merge code up to the parent branch and the merge failed due to a conflict. How do you identify the conflicting files?

A. Use the Magento CLI to view the system logs
B. Submit a support ticket
C. View the event log in the Cloud UI ***(correct)***
D. Use the git CLI correct ***(correct)***

Question 2 (2 of 5)
You have identified the files that caused the Git merge conflict. Which tool will you use to resolve the conflicts?

A. Magento Cloud UI
B. magento-cloud CLI
C. git CLI ***(correct)***
D. git UI ***(correct)***
E. Your analytics tool

Unit 11. Migration to Cloud

11.1 Magento On-Premise Installation Migration

Exercise Question 1
Do you need to copy your composer.lock to the Cloud environment while migrating?

Solution
You have to make some changes in the composer.json (magento/project-enterprise-edition => magento/magento-cloud-metapackage, add some extra options). This requires the regeneration of the composer.lock file, so you cannot just copy this file because it won't be used if the checksum of composer.json has changed, and that isn't what you want.

Exercise Question 2
What do you need to do for migrating media files directly from the Magento hosting to the Cloud environment using rsync or scp?

Solution
On the Magento hosting you must have an SSH key which is added to your Magento Cloud account. In this case you'll be able to log in over SSH on the Cloud environment and copy files using rsync or scp.

11.2 Magento 1 Migration

Exercise Question 1
Is the Data Migration Tool required for migrating the on-premise installation to Cloud?

Solution
The Data Migration Tool is used for migrating from Magento 1 to Magento 2. It is not used for migration to Cloud.

Unit 12. Go Live

12.1 DNS

Exercise Questions

1. Where should the domain name be pointed?
   - A. Fastly ***(correct)***
   - B. Magento Cloud
   - C. Server

12.2 TLS

Exercise Question

How do you test the SSL certificate on the server?

Solution

Using SSH enter: `dig txt <domain.name>`

```
dig txt <domain.name>
```

```
;; ANSWER SECTION:
clownpnd.work. 3600 IN TXT "_globalsign-domain-verification=BcVk5jK9L89hrF-
agPFiS7pXBamdJRwvmoKFF4AtLd"
clownpnd.work. 3600 IN TXT "v=spf1 include:sendgrid.net ~all"
```

12.3 Fastly

Exercise

Test your branch with Fastly using https://magento-tester.global.ssl.fastly.net/magento-tester/

Solution

https://magento-tester.global.ssl.fastly.net/magento-tester/#https://staging-vdt2zeq-2ewbmop5dr3xu.eu-3.magentosite.cloud||eu-3.magentosite.cloud

Unit 13 Maintaining a Magento Cloud Project

13.1 Upgrading to a New Version

Exercise 1
What is the process for upgrading ECE-Tools?

1. Create a new branch off of your development branch, either:
    a. Using the Web GUI
        - Select the development branch
        - Click the branch icon in the upper right
        - Name appropriately
        - git checkout the new branch locally
    b. Via CLI
        - Browse to local checkout of the project
        - `magento-cloud environment:branch <name_of_test_branch> <name_of_develop_branch>`
2. From the root of the checkout, run the command `composer update magento/ece-tools`
3. git add & commit & push

Exercise 2
How would you upgrade Magento from 2.2.2 to 2.2.3?

1. Create a new branch off of your development branch, either:
    a. Using the Web GUI
        - Select the development branch
        - Click the branch icon in the upper right
        - Name appropriately
        - git checkout the new branch locally
    b. Via CLI
        - Browse to local checkout of the project
        - `magento-cloud environment:branch <name_of_test_branch> <name_of_develop_branch>`
2. `composer require magento/magento-cloud-metapackage:">=2.2.4 <2.2.5" --update-with-dependencies`
3. git add & commit & push

13.2 Upsizing

Exercise Questions

1. You are in the planning stage for a large promotion with 40% off all products and you expect a significant increase in traffic. When do you notify Magento of the event?

    A. Immediately

    B. At least 2-3 days prior to the event ***(correct)***

    C. The day of the event

    D. Never

2. Which three key data points do you need to provide to Magento 72 hours prior to a planned traffic surge event? (pick 3)

   A. When the event will occur ***(correct)***

   B. How much revenue the event will generate

   C. % increase in traffic / orders ***(correct)***

   D. Expected duration of the event ***(correct)***

   E. Which products will be most popular during the event

3. When reviewing New Relic you notice a surge in requests, and you realize you are experiencing an unexpected traffic surge. When do you need to notify Magento for upsizing to occur?

   A. Immediately

   B. It is too late, your site will crash

   C. Magento will upsize the environment as necessary to maintain the SLA ***(correct)***

   D. The next day

4. You have notified Magento of a planned traffic surge event that will occur in approximately 72 hours. What are the benefits of this prior notice? (pick 2)

   A. The additional resources are available when the traffic arrives ***(correct)***

   B. The Support team will be aware of the event ***(correct)***

   C. Upsizing prior to a surge event guarantees a higher conversion rate

   D. There are no benefits to prior notice