Provision a Managed Service

Goal

Provision an instance of a mysql database managed service and connect to it from your PHP page.

Approximate time: 40 minutes

Exercises

Modify the PHP app to include a PHP function

1. Modify the index.php file in your version 2 application so that it includes an external file named db.php. Also call a function in that file named getSampleDataFromDB(). You can copy the code from below:

```
<?php include_once("db.php"); ?>
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <title>Hello world</title>
    </head>
    <body>
        <p>Hello world<?php echo " version 2"; ?></p>
        <p>Database value is <?php echo getSampleDataFromDB(); ?></p>
    </body>
</html>
```

2. In your version 2 application directory, add the db.php file as shown:

```
<?php
function getSampleDataFromDB()
{
//get environment information
echo phpinfo();
return;
}
?>
```

3. cf push the application. Notice that the db.php file is also automatically deployed with the application.
4. The application's page should have a lot of information, all coming from the phpinfo() call. Notice that the VCAP_SERVICES environment variable is not set, because our app does not have any services associated with it.

Create a mysql managed service instance

1. Use Apps Manager's Marketplace or the CLI to create an instance of the ClearDB MySQL Database named my-test-mysql-db.
2. Bind the mysql service instance to your version 2 application.
3. Use Apps Manager and the CLI to verify that the service is bound to your application.
4. cf push the application. Verify that the VCAP_SERVICES environment variable is set three ways. 1) View the variables in the application's Env Variables tab in Apps Manager. 2) View the application's environment variables using the CLI. 3) View the variables in the phpinfo() page of your application.

Enable the mysqli extension in the PHP buildpack

1. We will use PHP's mysqli extension to query the database. This extension is not enabled by default in the PHP buildpack. Information on the buildpacks is in github: https://github.com/cloudfoundry/php-buildpack . You can see how to configure this buildpack by looking at the Configuration Options section in github.
2. To configure the buildpack, add a .bp-config directory to your application's directory. Then add an options.json file to that directory with the following content:

```
{
    "PHP_EXTENSIONS": [ "mysqli"]
}
```

Note: Windows users may need to use a command window to create the .bp-config directory. Make sure that the options.json file doesn't have a hidden .txt extension.

Test that you can connect to the database using hard-coded credentials

1. Change the db.php file to contain the following: (**you will have to modify to specify your own values** - look under `VCAP_SERVICES`) available here (https://github.com/S2EDU/PCFImmersionStudentFiles/blob/master/lab-services/db.php):

```
<?php
function getSampleDataFromDB()
{

  // Verify that mysqli has been enabled
    if (function_exists('mysqli_connect') == false) {
        echo "need to enable mysqli!" ;
        error_log("need to enable mysqli!", 0);
        return;
    }

  // Connect with hard-coded connection info- look under VCAP_SERVICES
    $dbname = "YOUR INFO HERE";  //under 'name' and something like: ad_a4fffdc3d80e473
    $hostname = "YOUR INFO HERE"; // something like us-cdbr-iron-east-02.cleardb.net
    $username = 'YOUR INFO HERE'; // something like bdfe1eb555af52
    $password = "YOUR INFO HERE"; // something like 348d05d5

  // Connect to the database
    $mysql = mysqli_connect($hostname, $username, $password, $dbname );
    if (mysqli_connect_errno()) {
        echo "Failed to connect to MySQL: " . mysqli_connect_error();
        error_log("Failed to connect to MySQL: " . mysqli_connect_error());
        return;
    } else {
        echo "Success connecting to the db!";
    }
}
?>
```

2. Run cf push to deploy the application. Use a browser to verify that you are getting a success message on your application's index.php page.

Test that you can connect to the database using environment variables

1. Change the db.php file to contain the following: (notice that we are using PHP's getenv() function to read the environment variables rather than hardcoding them)

```
<?php
function getSampleDataFromDB()
{

//verify that mysqli has been enabled
    if (function_exists('mysqli_connect') == false) {
        echo "need to enable mysqli!";
        error_log("need to enable mysqli!", 0);
        return;
    }

//connect with connection info from environment variables
    $json = getenv('VCAP_SERVICES');
    $service_name;
    if (strpos($json,'cleardb')){
      $service_name='cleardb'; // service is on PWS
    } elseif (strpos($json,'p-mysql')){
      $service_name='p-mysql'; // service is on Pivotal MySQL service
    } else {
      echo "Error determining service name. Check VCAP_SERVICES";
      error_log("Error determining service name. Check VCAP_SERVICES");
      return;
    }
    $arr = json_decode($json, true);
    $hostname = $arr[$service_name][0]['credentials']['hostname'];
    $username = $arr[$service_name][0]['credentials']['username'];
    $password = $arr[$service_name][0]['credentials']['password'];
    $dbname = $arr[$service_name][0]['credentials']['name'];

//connect to the database
    $mysql = mysqli_connect($hostname, $username, $password, $dbname );
    if (mysqli_connect_errno()) {
        echo "Failed to connect to MySQL: " . mysqli_connect_error();
        error_log("Failed to connect to MySQL: " . mysqli_connect_error());
        return;
    } else {
```

```
        echo "Success connecting to the db!";
    }
}
?>
```

2. Run cf push to deploy the application. Use a browser to verify that you are getting a success message on your application's index.php page. Notice that the connection information is no longer hardcoded in your application.

Read data from the database and display it on the application's page

1. Change the db.php file to contain the following: (this code creates a table in the database if necessary, then queries the table and returns a sample value of 5.)

```
<?php
function getSampleDataFromDB()
{

//verify that mysqli has been enabled
    if (function_exists('mysqli_connect') == false) {
        echo "need to enable mysqli!" ;
        error_log("need to enable mysqli!", 0);
        return;
    }


 //connect with connection info from environment variables
    $json = getenv('VCAP_SERVICES');
    $service_name;
    if (strpos($json,'cleardb')){
      $service_name='cleardb'; // service is on PWS
    } elseif (strpos($json,'p-mysql')){
      $service_name='p-mysql'; // service is on Pivotal MySQL service
    } else {
      echo "Error determining service name. Check VCAP_SERVICES";
      error_log("Error determining service name. Check VCAP_SERVICES");
      return;
    }
    $arr = json_decode($json, true);
    $hostname = $arr[$service_name][0]['credentials']['hostname'];
    $username = $arr[$service_name][0]['credentials']['username'];
    $password = $arr[$service_name][0]['credentials']['password'];
    $dbname = $arr[$service_name][0]['credentials']['name'];

//connect to the database
    $mysql = mysqli_connect($hostname, $username, $password, $dbname );
    if (mysqli_connect_errno()) {
        echo "Failed to connect to MySQL: " . mysqli_connect_error();
        error_log("Failed to connect to MySQL: " . mysqli_connect_error());
        return;
```

```
    }

//check if testtable has been created, if not then create it
        $sql = "select num_total from testtable";
        $result = mysqli_query($mysql, $sql);
        if (empty($result)) {
            echo "testtable is empty!" ;
            // sql to create table
             $sql = "CREATE TABLE testtable (
                id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
                num_total VARCHAR(30) NOT NULL
                )";
             if (mysqli_query($mysql, $sql)) {
                echo "Table created successfully" ;
            return;
             } else {
                 echo "Error creating table: " . mysqli_error($mysql);
             error_log("Error creating table: " . mysqli_error( $mysql));
                 return;
             }
        }
    $anArray = mysqli_fetch_assoc($result);
    if (empty($anArray)){
        $sql = "INSERT INTO testtable (num_total) VALUES (5)";
        if (mysqli_query($mysql, $sql)) {
            echo "num_total value added successfully";
        } else {
            echo "Error adding num_total value: " . mysqli_error( $mysql);
            error_log("Error adding num_total value: " . mysqli_error( $mysql));
        }
    }

//get and return sample data from the database
    $sql = "select num_total from testtable" ;
    $result = mysqli_query($mysql, $sql);
    if (empty($result)){
      return "result is empty";
```

```
        error_log("result is empty");
    }
    if ((empty($anArray)) or (empty($anArray[ "num_total" ]))) {
        echo "error: return value not set" ;
        error_log("error: return value not set");
        return;
    } else {
        return $anArray ["num_total"];
    }
}
?>
```

2. Run cf push to deploy the application. Use a browser to verify that you are seeing a value of 5 from the database on your application's index.php page. You may have to **click refresh a few times** to populate the database before seeing the value.

You have successfully provisioned and used a managed service!

Last updated 2016-01-08 06:14:44 PST