LAB 02

Platform-as-a-Service with Heroku

Fullname: Trương Đặng Trúc Lâm

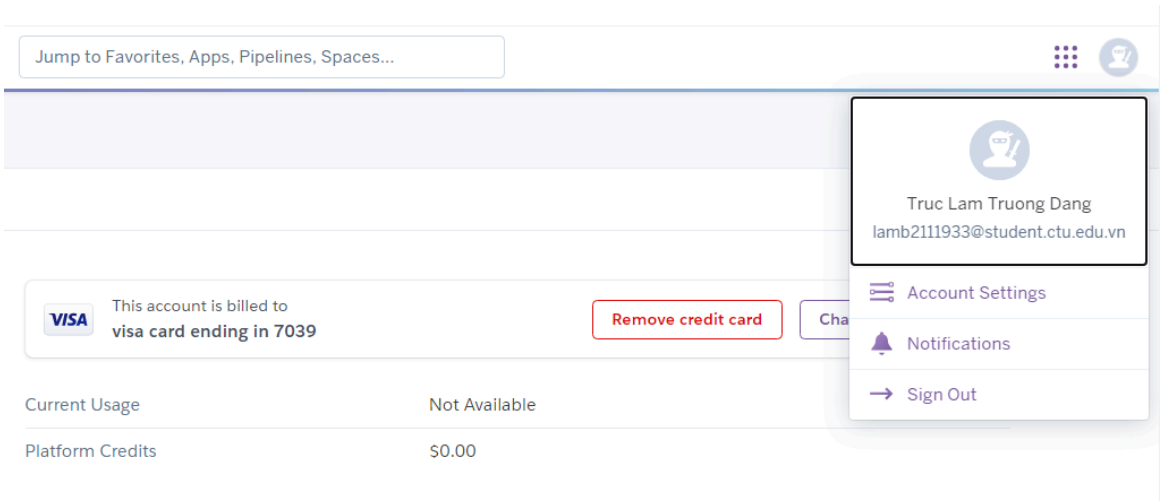
Student ID: B2111933

- Note: screenshots need to be clear and good-looking; submissions must be in PDF format.

Heroku is a platform as a service (PaaS) that enables developers to build, run, and operate applications entirely in the cloud. In this lab, you will learn about building, deploying, and managing your apps on Heroku by finishing the exercises below.

1. **Deploying a sample PHP application with Heroku**
   https://devcenter.heroku.com/articles/getting-started-with-php
   (Take 2 screenshots of the sample PHP application, one after “Push local changes” step and another after “Provision a database” step)

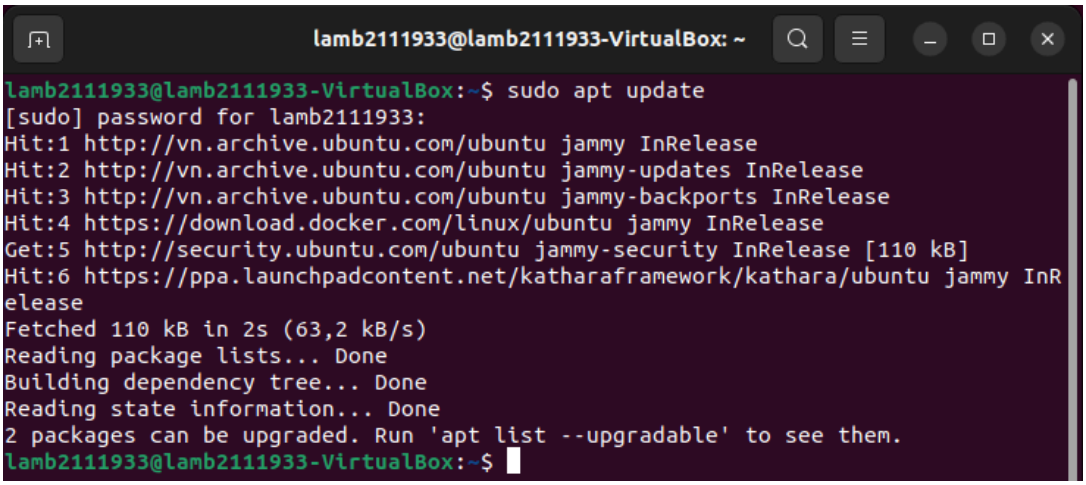
- Prepare a verified Heroku Account



Verified Heroku Account

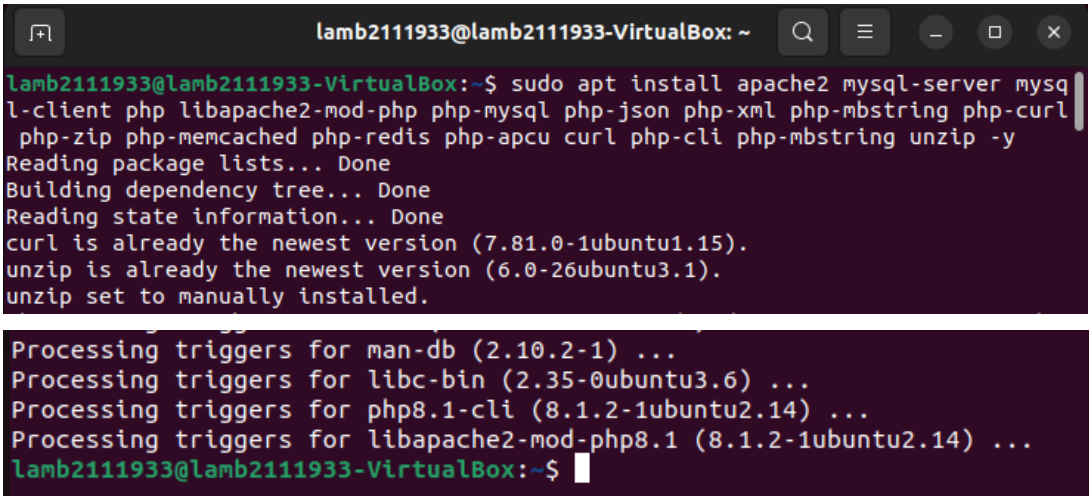
- Install apache, php and heroku

```
$sudo apt update
```



Update **apt** packages

```
$sudo apt install apache2 mysql-server mysql-client php libapache2-mod-php php-mysql php-json php-xml php-mbstring php-curl php-zip php-memcached php-redis php-apcu curl php-cli php-mbstring unzip -y
```

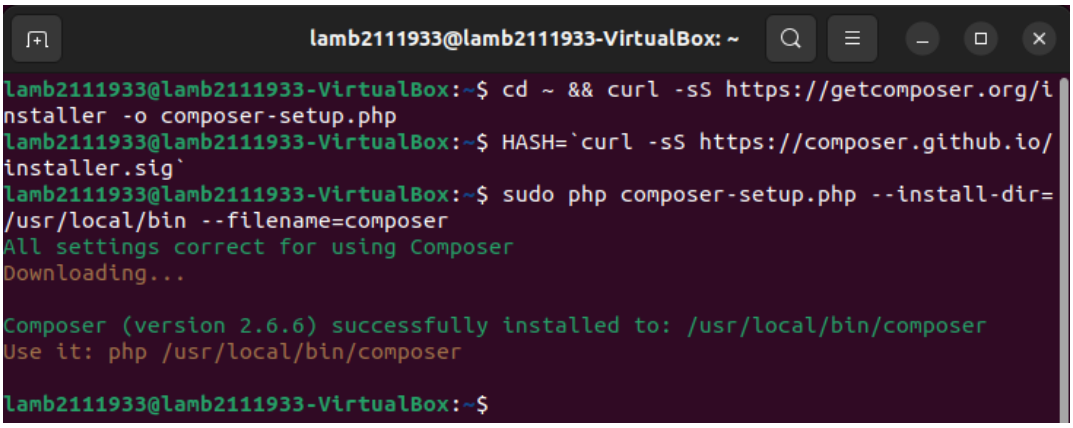


Install **apache**, **php** and **heroku**

- Install composer

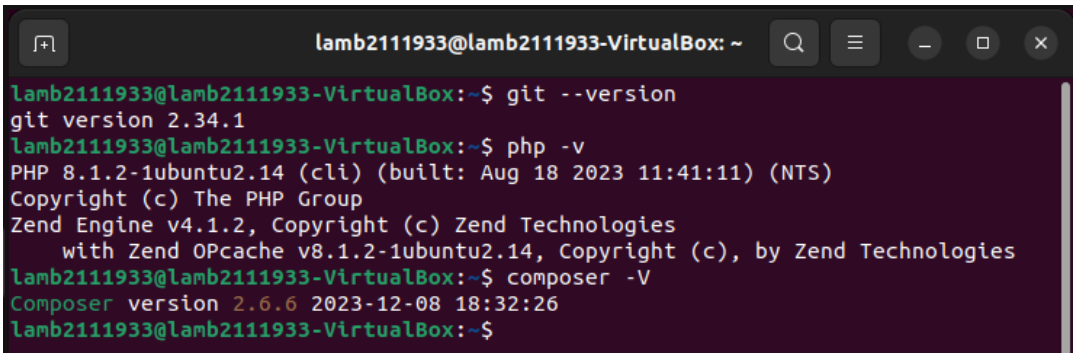
```
$cd ~ && curl -sS https://getcomposer.org/installer -o composer-setup.php
$HASH=`curl -sS https://composer.github.io/installer.sig`
```

```
$sudo php composer-setup.php --install-dir=/usr/local/bin
--filename=composer
```



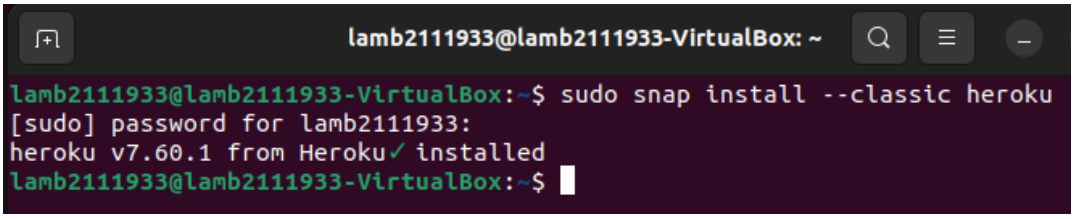
Install **composer**

- Check that you have the prerequisites installed properly.



- Create heroku application (have done in exercise 1.)

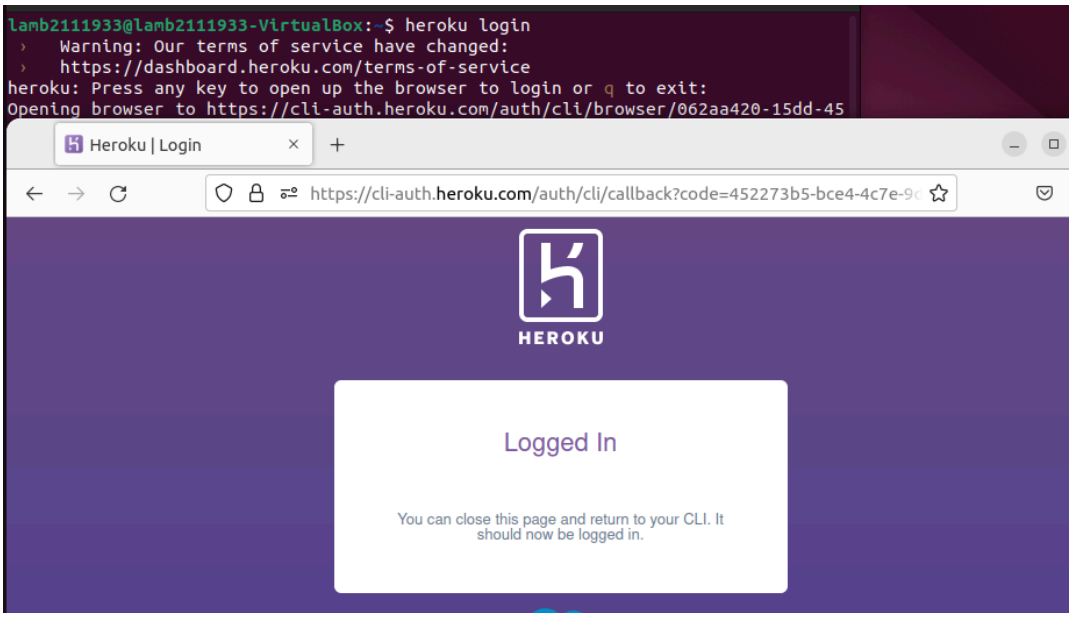
```
$sudo snap install --classic heroku
```



Install **heroku**

```
$heroku login
```

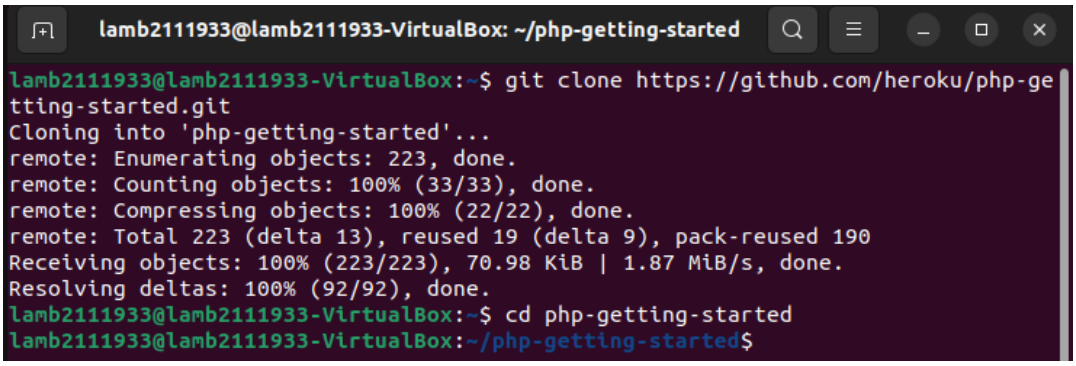
if the web browser cannot open automatically, please run it manually; then copy the log in link to the address bar.



Log in **heroku**

- Clone the **php-getting-started.git** to your local machine

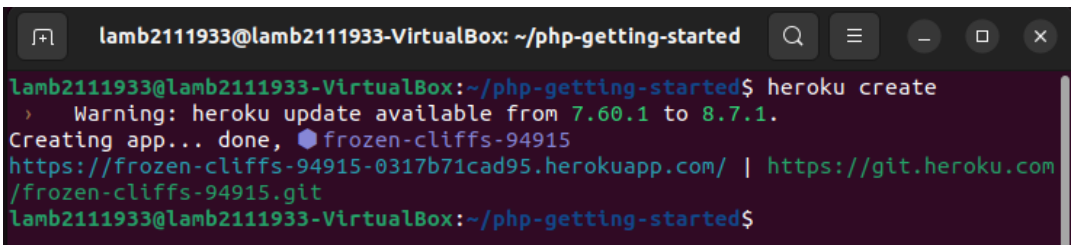
```
$git clone https://github.com/heroku/php-getting-started.git
$cd php-getting-started
```



Get the local version of **php-getting-started.git**

- Create an app on Heroku, which prepares Heroku to receive your source code

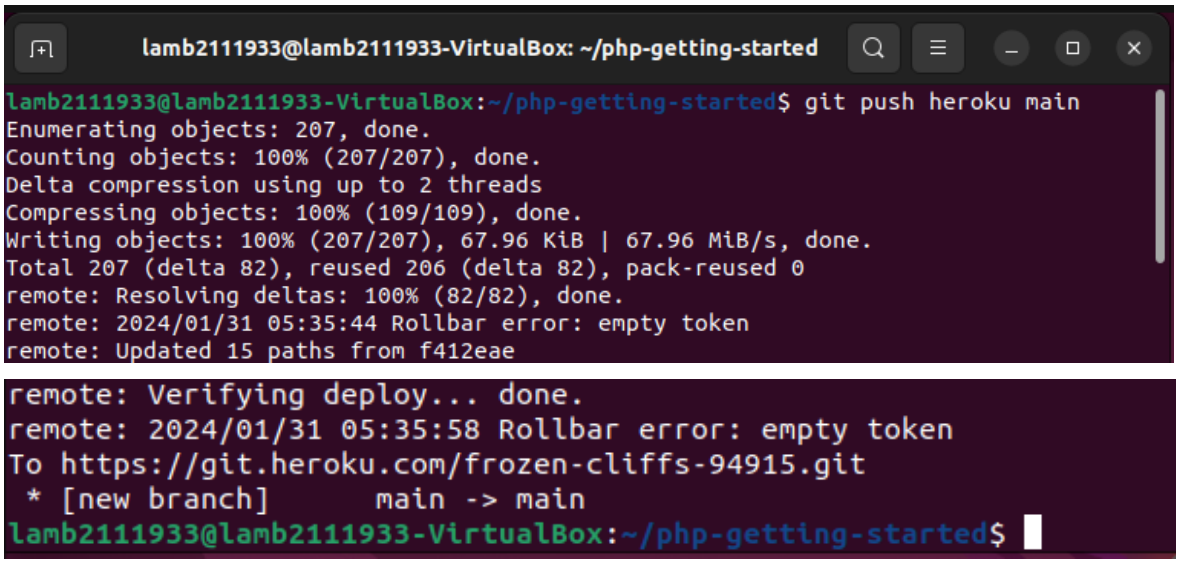
```
$heroku create
```



Create an **application** on **Heroku**

- Now deploy your code

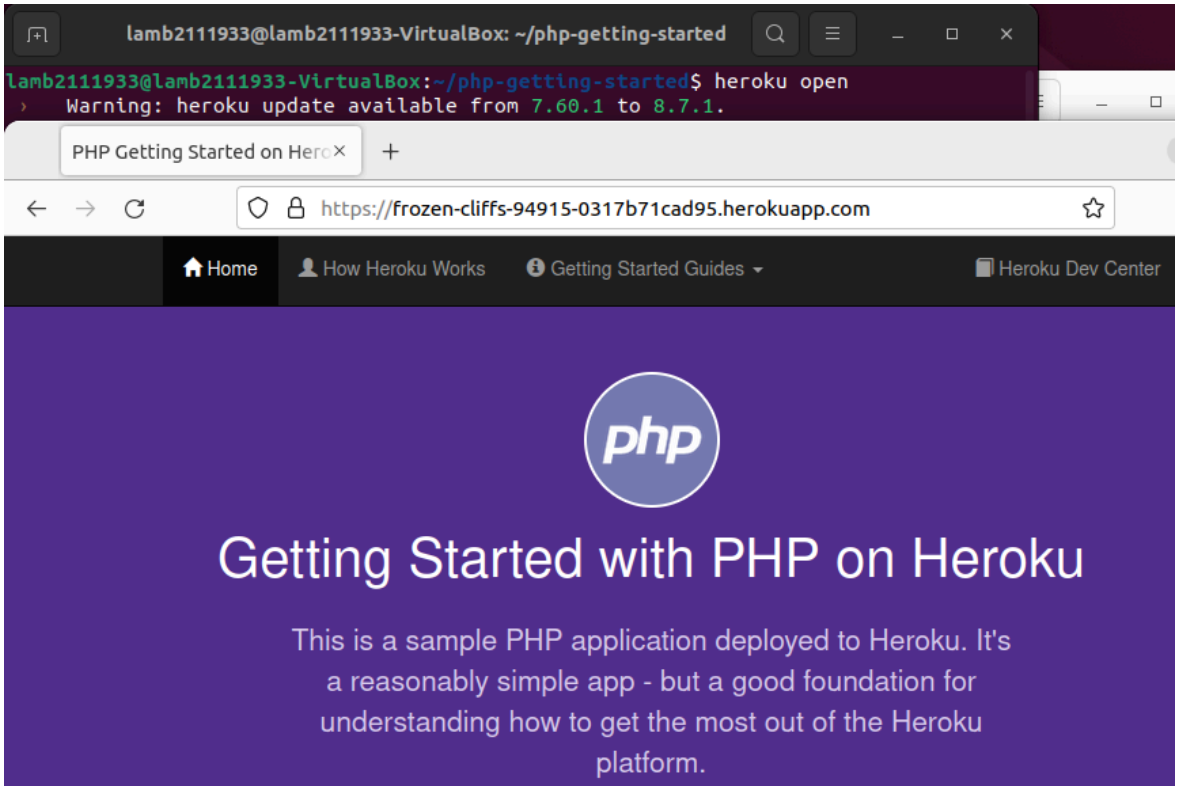
```
$git push heroku main
```



Deploy the code of **php-getting-started.git**

- Launch your application

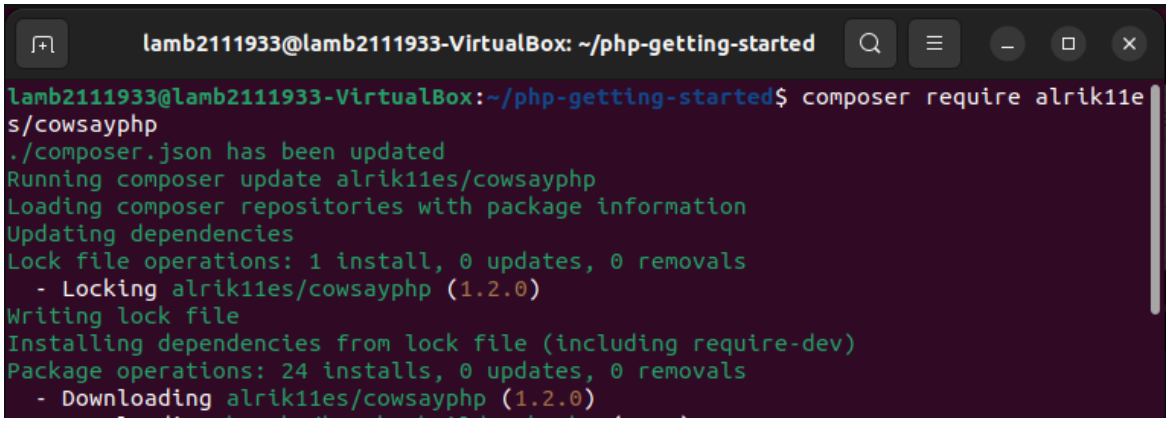
```
$heroku open
```



This is your **PHP** application

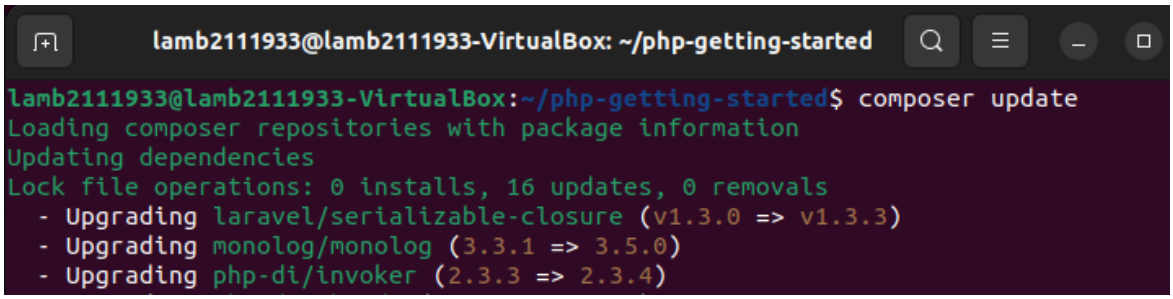
Push local changes:

```
$composer require alrik11es/cowsayphp
```



Use **Composer** to require the new dependency

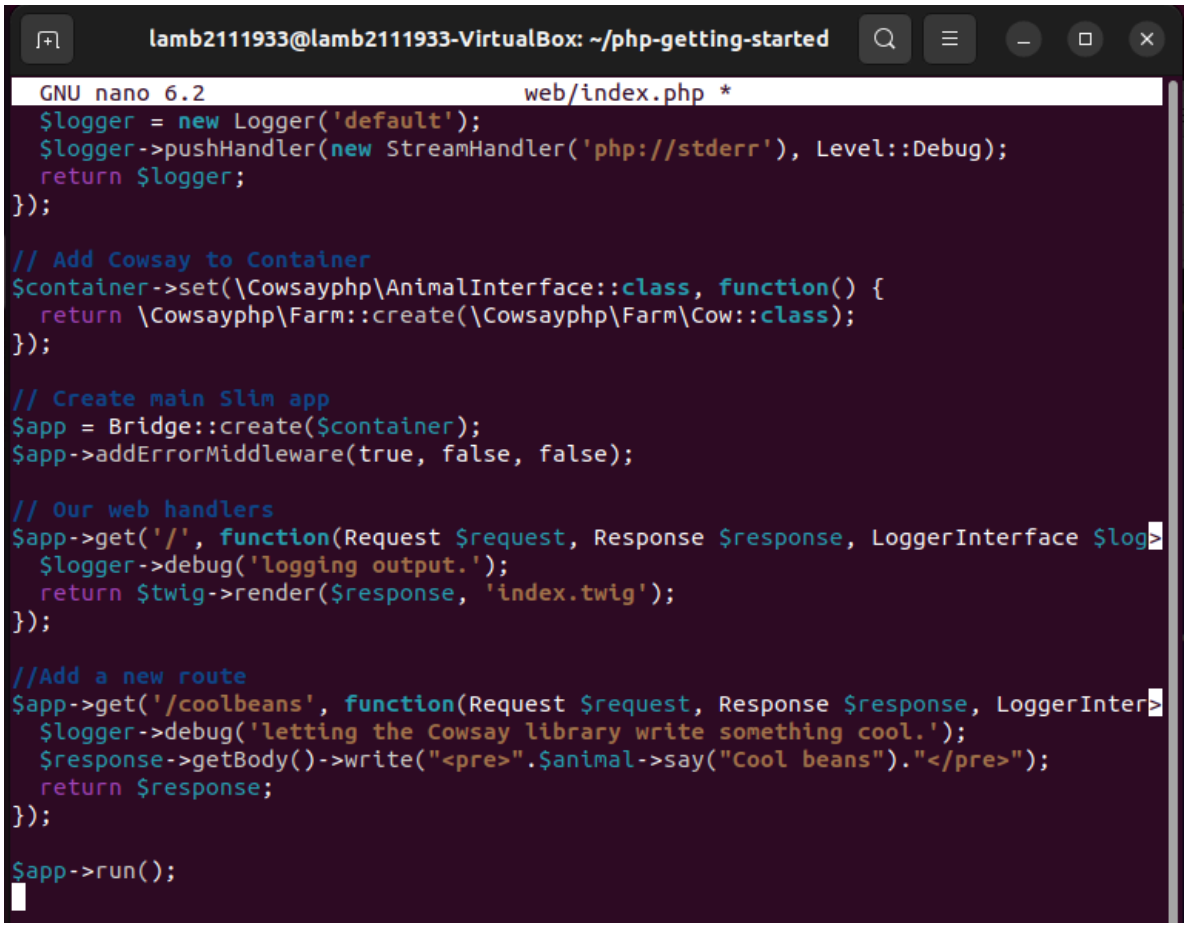
```
$composer update
```



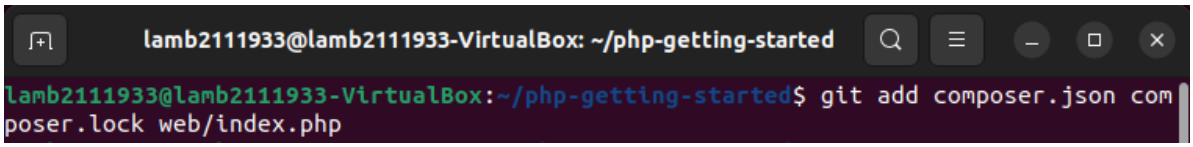
Be sure to update the dependencies

```
$nano web/index.php
// Add Cowsay to Container
$container->set(\Cowsayphp\AnimalInterface::class, function() {
  return \Cowsayphp\Farm::create(\Cowsayphp\Farm\Cow::class);
});

$app->get('/coolbeans', function(Request $request, Response
$response, LoggerInterface $logger, \Cowsayphp\AnimalInterface
$animal) {
$logger->debug('letting the Cowsay library write something
cool.');
$response->getBody()->write("<pre>".$animal->say("Cool
beans")."</pre>");
  return $response;
```

```
});
```

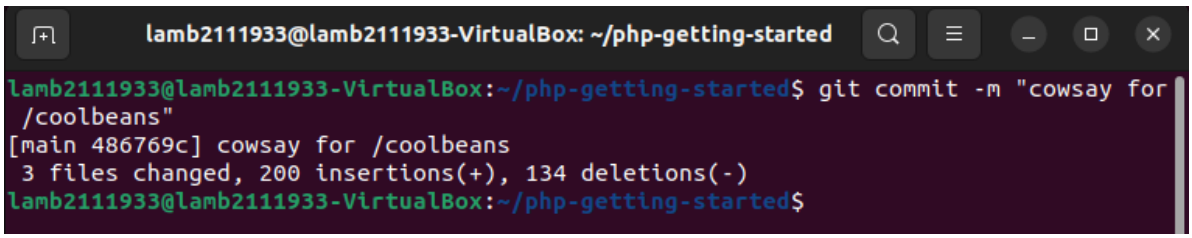



```
$ git add composer.json composer.lock web/index.php
```



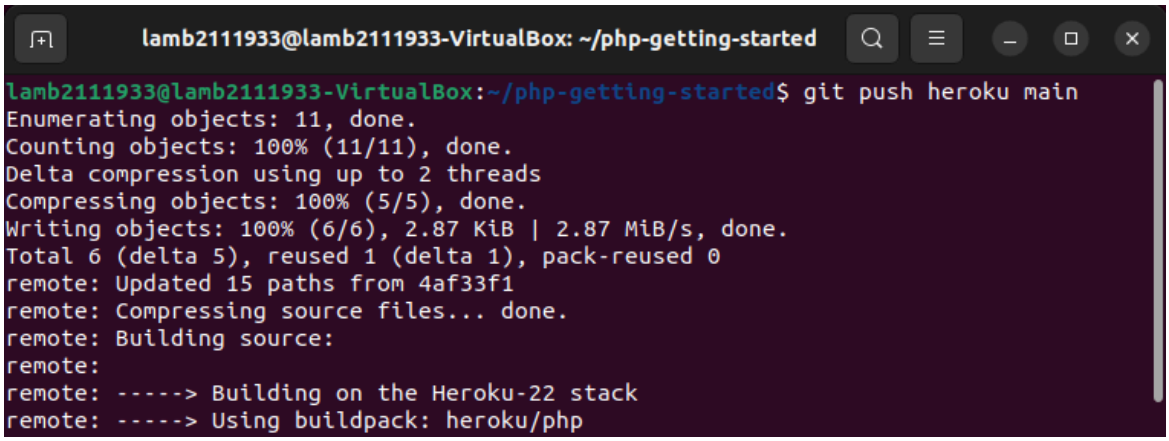
Add the modified files to your local repository

```
$ git commit -m "cowsay for /coolbeans"
```



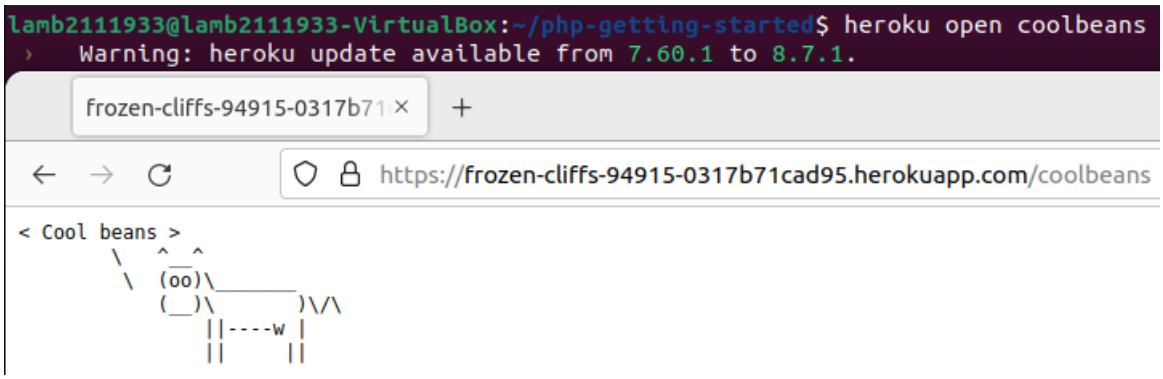
Commit these changes

```
$ git push heroku main
```



Deploy just as you did previously

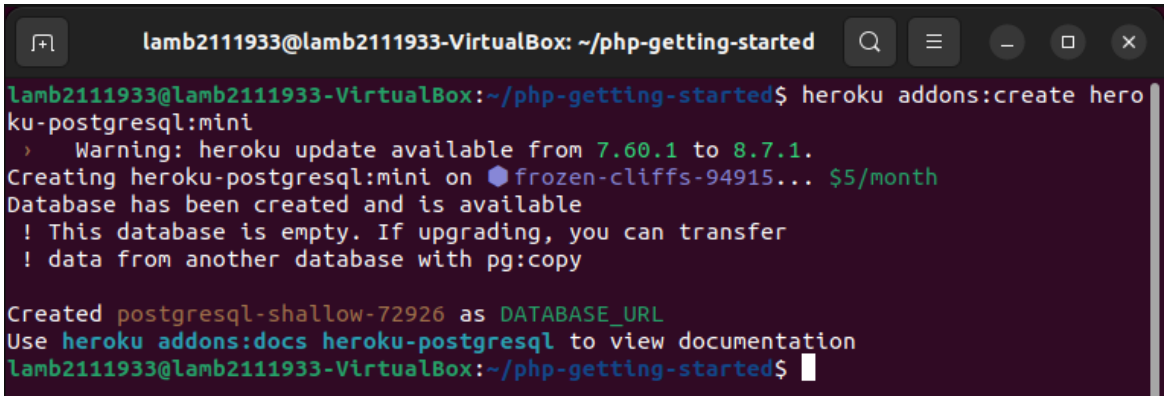
```
$ heroku open coolbeans
```



This is the result of the **coolbeans** page

Provision a database:

```
$heroku addons:create heroku-postgresql:mini
```



Add the **database**

```
$nano web/index.php

// Add Database connection to Container
$container->set(PDO::class, function() {
    $dburl = parse_url(getenv('DATABASE_URL') ?: throw new
Exception('no DATABASE_URL'));
  return new PDO(sprintf(
    "pgsql:host=%s;port=%s;dbname=%s;user=%s;password=%s",
    $dburl['host'],
    $dburl['port'],
      ltrim($dburl['path'], '/'), // URL path is the DB name,
must remove leading slash
    $dburl['user'],
    $dburl['pass'],
  ));
});
```

Add **Database connection** to **Container**

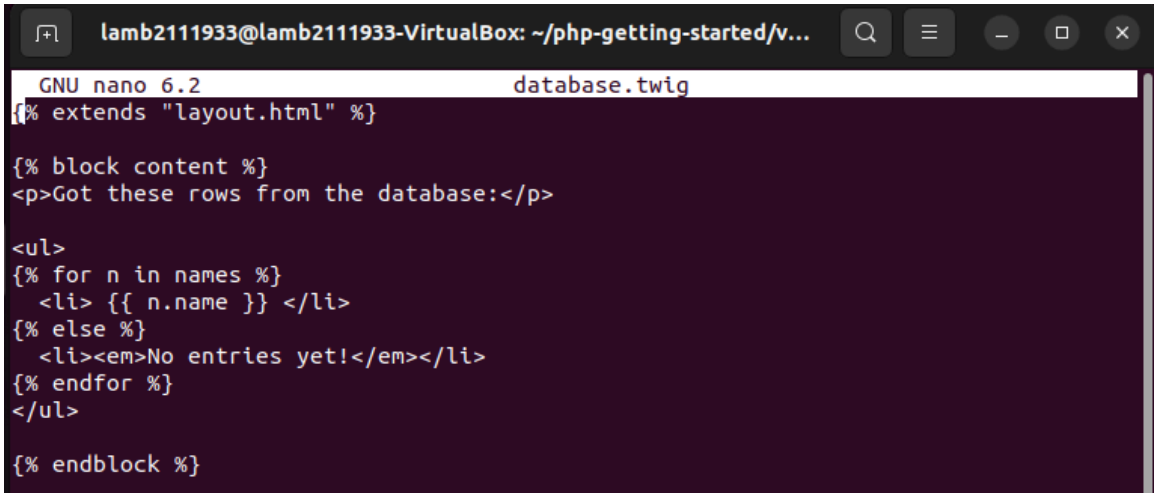
```
//add a new route handler for URL /db
$app->get('/db', function(Request $request, Response $response,
LoggerInterface $logger, Twig $twig, PDO $pdo) {
  $st = $pdo->prepare('SELECT name FROM test_table');
  $st->execute();
```

```
  $names = array();
  while($row = $st->fetch(PDO::FETCH_ASSOC)) {
    $logger->debug('Row ' . $row['name']);
    $names[] = $row;
  }
  return $twig->render($response, 'database.twig', [
    'names' => $names,
  ]);
});
```

```
//Add a new route handler for URL /db
$app->get('/db', function(Request $request, Response $response, LoggerInterface $l>
  $st = $pdo->prepare('SELECT name FROM test_table');
  $st->execute();
  $names = array();
  while($row = $st->fetch(PDO::FETCH_ASSOC)) {
    $logger->debug('Row ' . $row['name']);
    $names[] = $row;
  }
  return $twig->render($response, 'database.twig', [
    'names' => $names,
  ]);
});

$app->run();
```

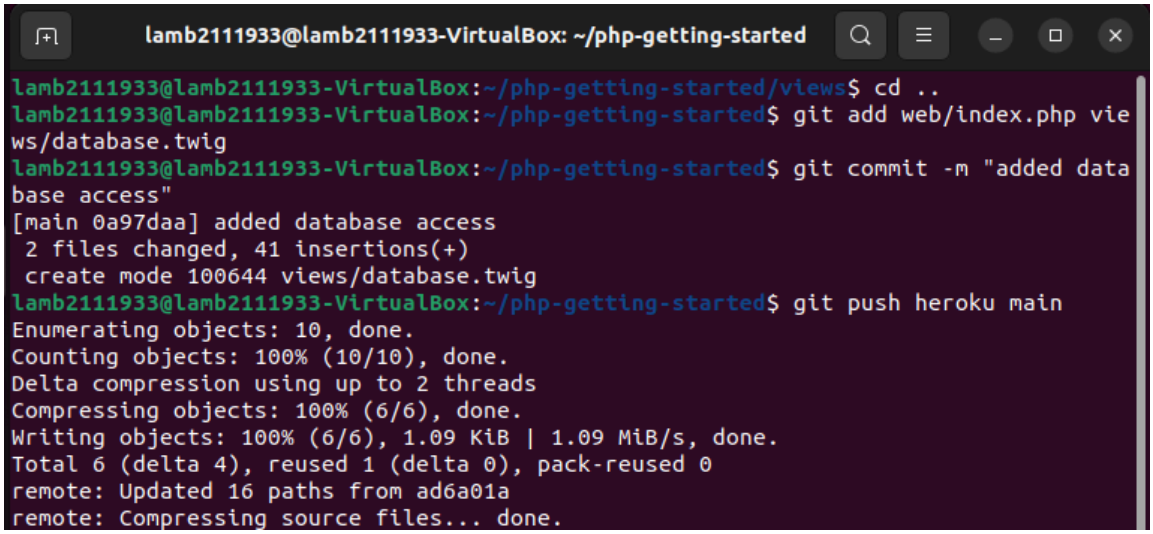
Add a **new route handler** for URL **/db**

```
$cd views
$nano database.twig

{% extends "layout.html" %}

{% block content %}
<p>Got these rows from the database:</p>

<ul>
{% for n in names %}
  <li> {{ n.name }} </li>
{% else %}
  <li><em>No entries yet!</em></li>
{% endfor %}
</ul>

{% endblock %}
```



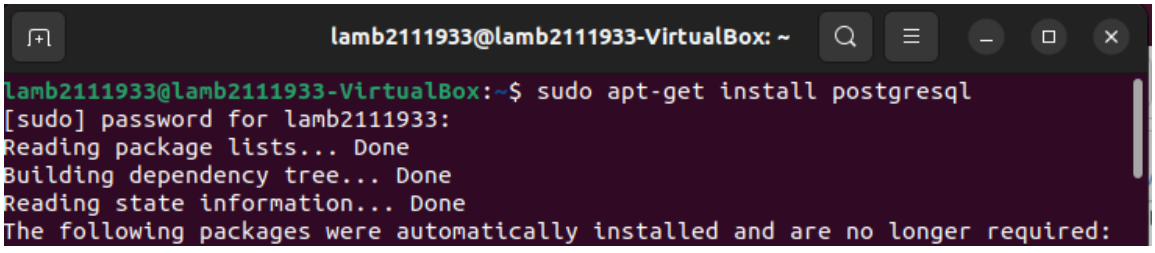
Create this **database.twig** template file inside the app's **views/** directory

```
$ cd ..
$ git add web/index.php views/database.twig
$ git commit -m "added database access"
$ git push heroku main
```



Deploy this app once more time

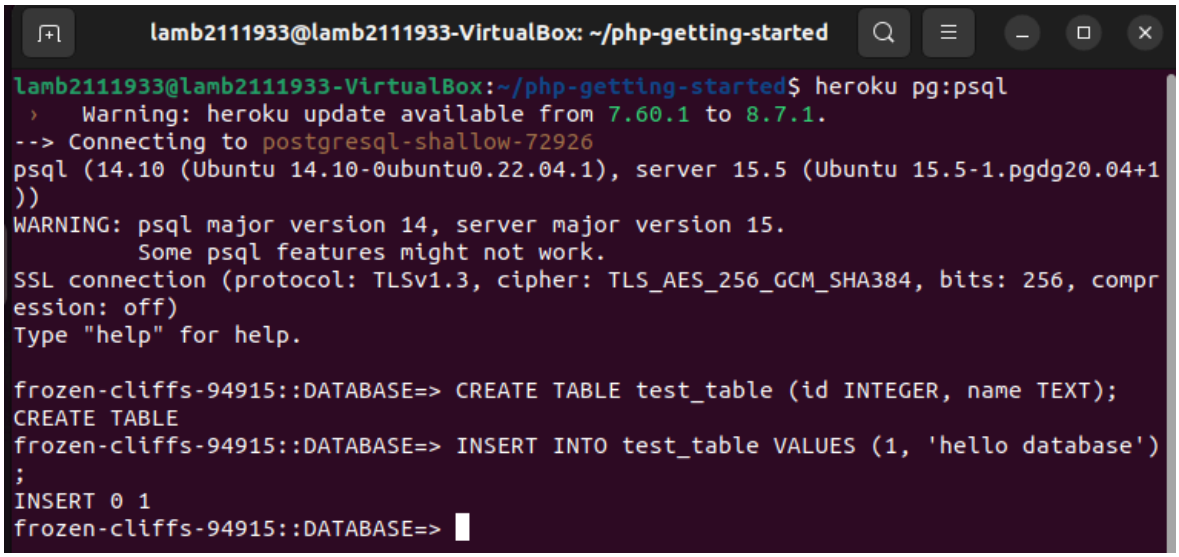
```
$ sudo apt-get install postgresql
```



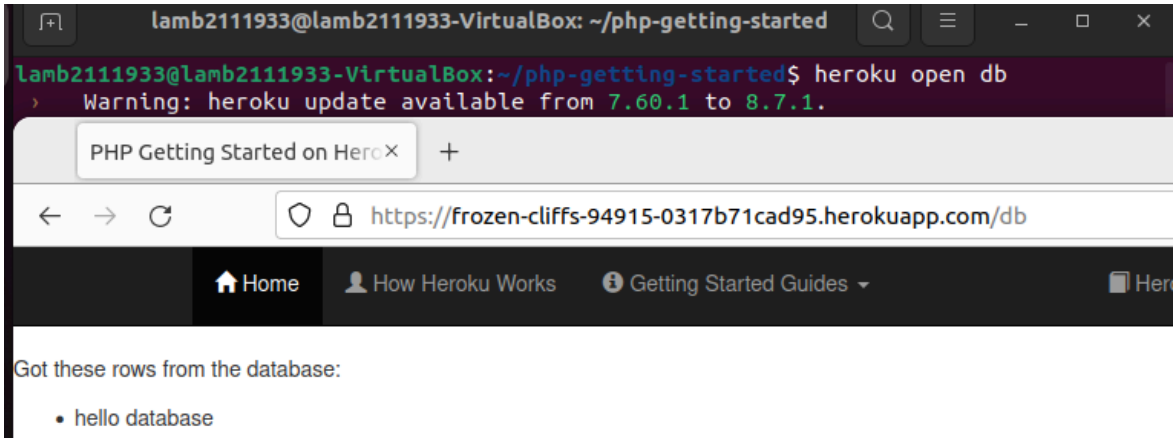
Install **Postgresql**

Connect to the database and use some commands:

```
$ heroku pg:psql
```



Use some commands like **CREATE** or **Insert** on database



This is the result of the **database page**

2. Deploying a online shopping website to Heroku

2.1. Deploying an online-shopping website to run locally on your computer

- Install apache, php and heroku

```
$sudo apt update
$sudo apt install apache2 mysql-server mysql-client php libapache2-mod-php php-mysql php-json php-xml php-mbstring php-curl php-zip php-memcached php-redis php-apcu curl php-cli php-mbstring unzip -y
```

(have done in exercise 1.)

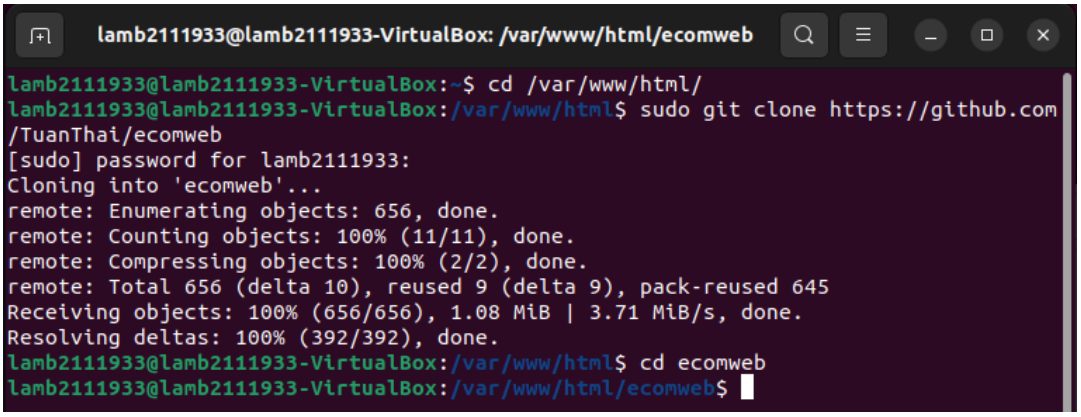
- Install composer (have done in exercise 1.)

```
$cd ~ && curl -sS https://getcomposer.org/installer -o composer-setup.php
$HASH=`curl -sS https://composer.github.io/installer.sig`
$sudo php composer-setup.php --install-dir=/usr/local/bin --filename=composer
```

(have done in exercise 1.)

- Clone the source of the website

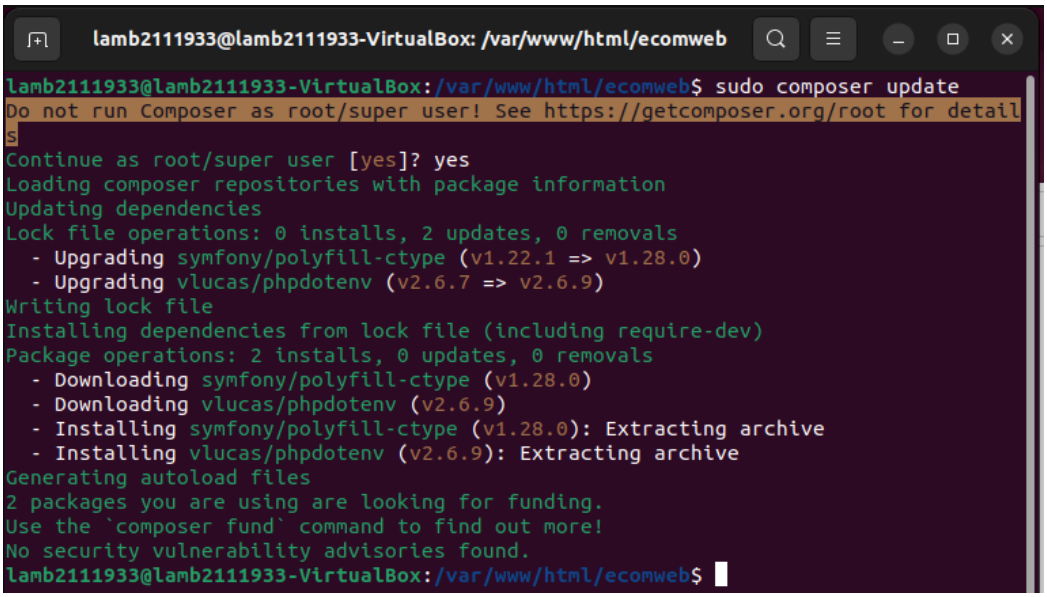
```
$cd /var/www/html/
$sudo git clone https://github.com/TuanThai/ecomweb
$cd ecomweb
```



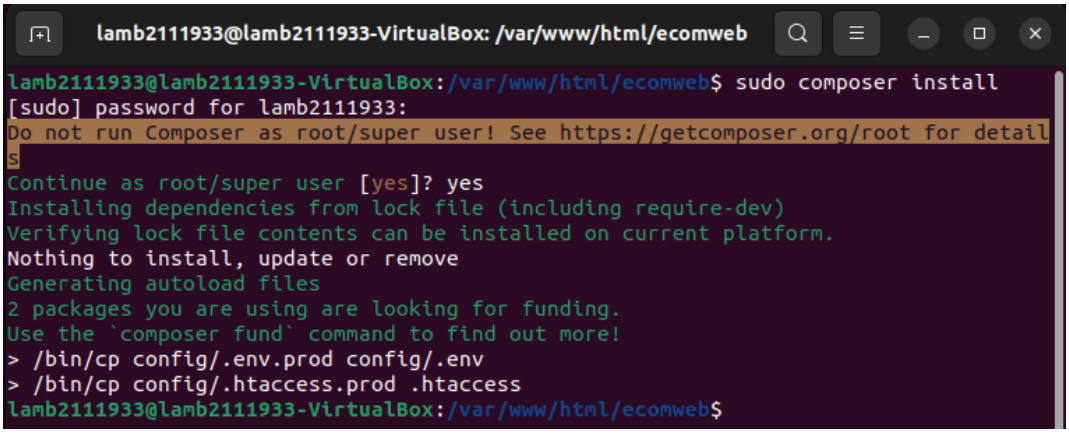
Create a local repository

- Install required modules using composer

```
$sudo composer update
```



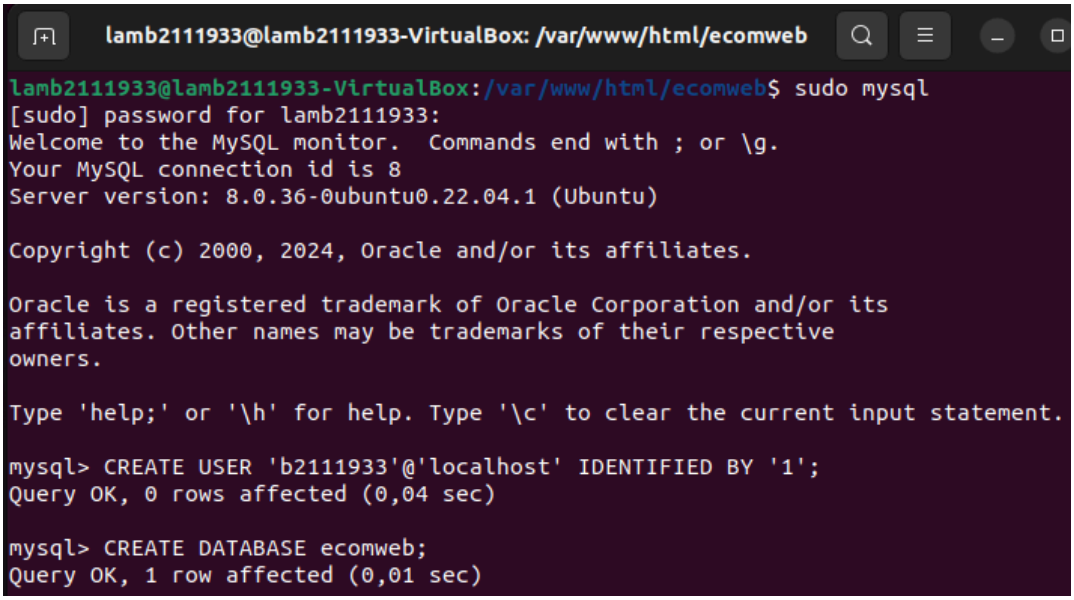
```
$sudo composer install
```



Install required modules using **composer**

- Create a mysql user and import database

```
$sudo mysql
mysql>CREATE USER 'tmtuan'@'localhost' IDENTIFIED BY 'password';
mysql>CREATE DATABASE ecomweb;
mysql>GRANT ALL PRIVILEGES ON ecomweb.* TO 'tmtuan'@'localhost';
mysql>use ecomweb;
mysql>source ./config/db_dump.sql;
mysql>exit;
```



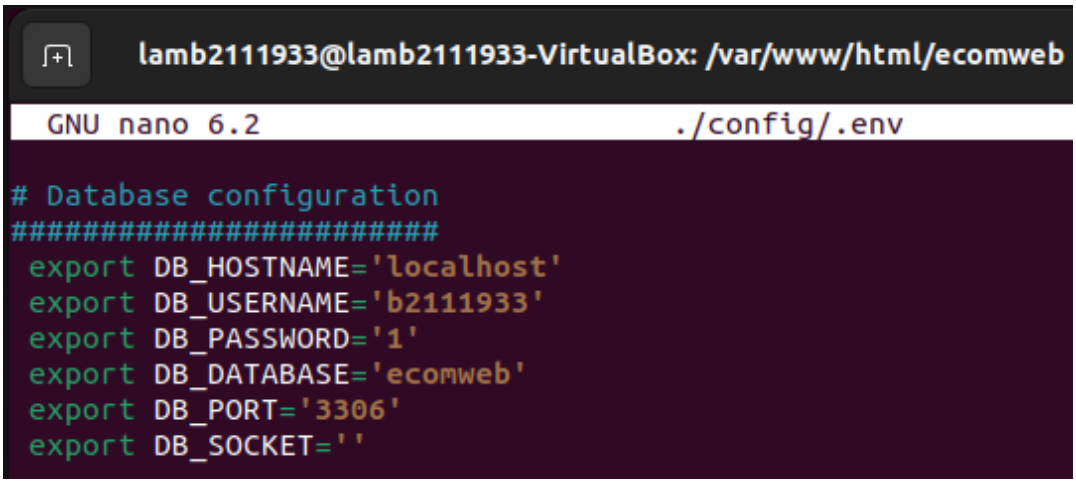
```
mysql> GRANT ALL PRIVILEGES ON ecomweb.* TO 'b2111933'@'localhost';
Query OK, 0 rows affected (0,01 sec)

mysql> use ecomweb;
Database changed
mysql> source ./config/db_dump.sql;
Query OK, 0 rows affected (0,00 sec)

Query OK, 0 rows affected (0,00 sec)

Query OK, 0 rows affected (0,00 sec)

Query OK, 0 rows affected, 1 warning (0,00 sec)

Query OK, 0 rows affected, 2 warnings (0,03 sec)

Query OK, 0 rows affected, 1 warning (0,00 sec)

Query OK, 0 rows affected, 2 warnings (0,03 sec)
```

```
Query OK, 1 row affected (0,00 sec)

Query OK, 1 row affected (0,00 sec)

Query OK, 1 row affected (0,00 sec)

Query OK, 0 rows affected (0,01 sec)

mysql> exit
Bye
lamb2111933@lamb2111933-VirtualBox:/var/www/html/ecomweb$
```

Create a **mysql user** and **import database** from the repository above

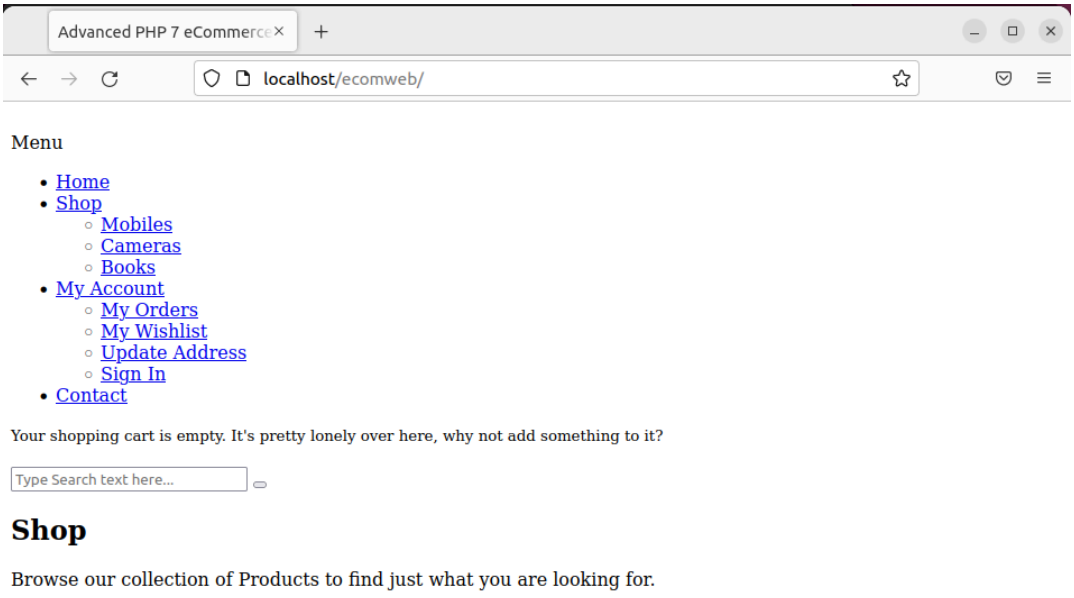
- Modify database configuration

```
$sudo nano ./config/.env
    #Modify the content of .env file
    export STORE_URL='http://localhost/ecomweb'
    ...
    export DB_HOSTNAME='localhost'
    export DB_USERNAME='tmtuan'
    export DB_PASSWORD='password'
    export DB_DATABASE='ecomweb'
    export DB_PORT='3306'
```



Modify database configuration

- Access the website using a web browser: http://localhost/ecomweb
(Take a screenshot of the website homepage)

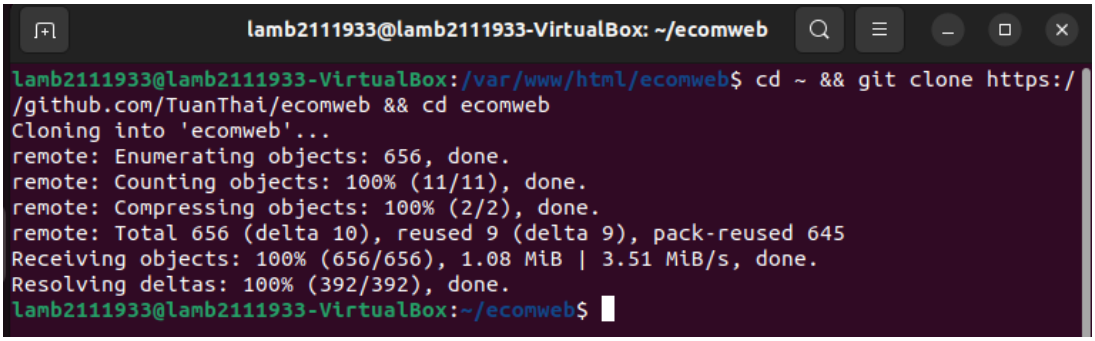


This is the **online-shopping website** that run locally

2.2. Deploying the website to Heroku

- Clone the source of the website

```
$cd ~ && git clone https://github.com/TuanThai/ecomweb && cd ecomweb
```



Clone the source code to your local machine

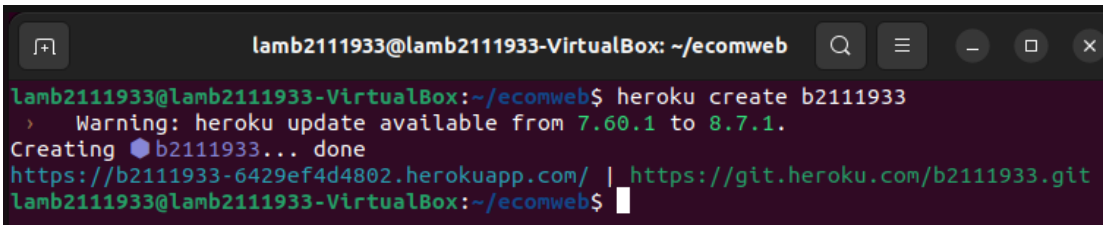
- Create heroku application

```
$sudo snap install --classic heroku
$heroku login
```

if the web browser cannot open automatically, please run it manually; then copy the log in link to the address bar.

(have done in exercise 1.)

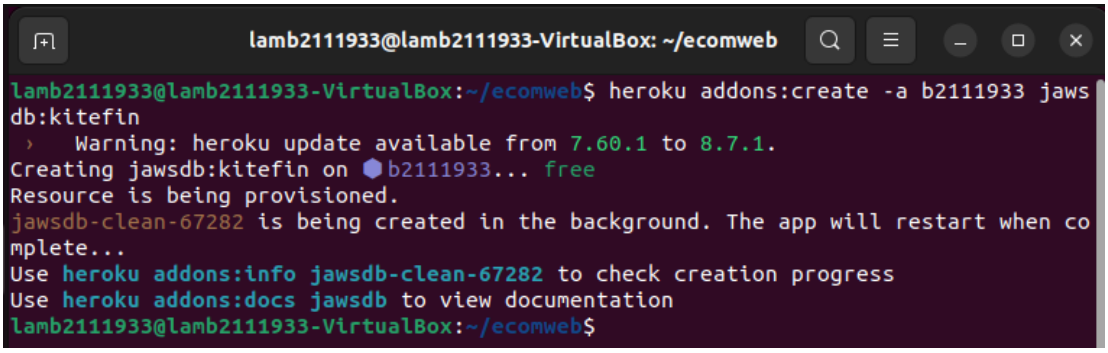
```
$heroku create <student_ID>
```



Create an **application** with **heroku**

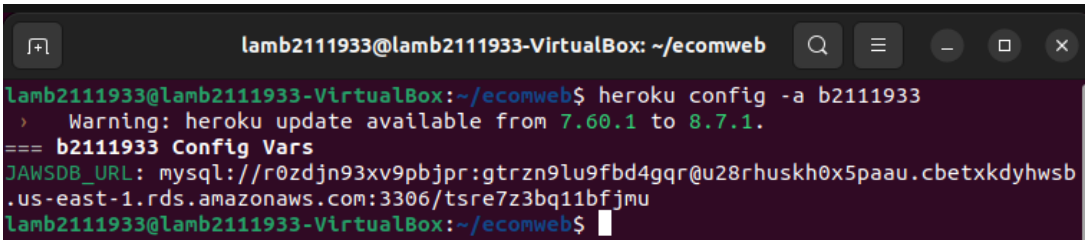
- Create database and import website data

```
$heroku addons:create -a <student_ID>  jawsdb:kitefin
```



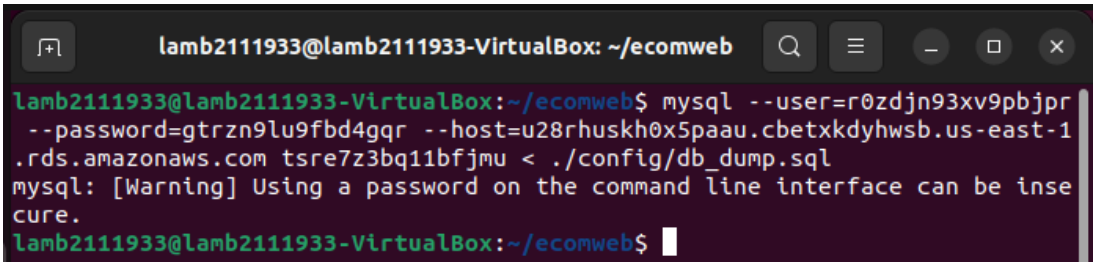
Create database

```
$heroku config -a <student_ID>
      #You will get the database information as below:
      #JAWSDB_URL:mysql://<username>:<password>@<server_name>/<database_name>?reconnect=true
```



Get the database information

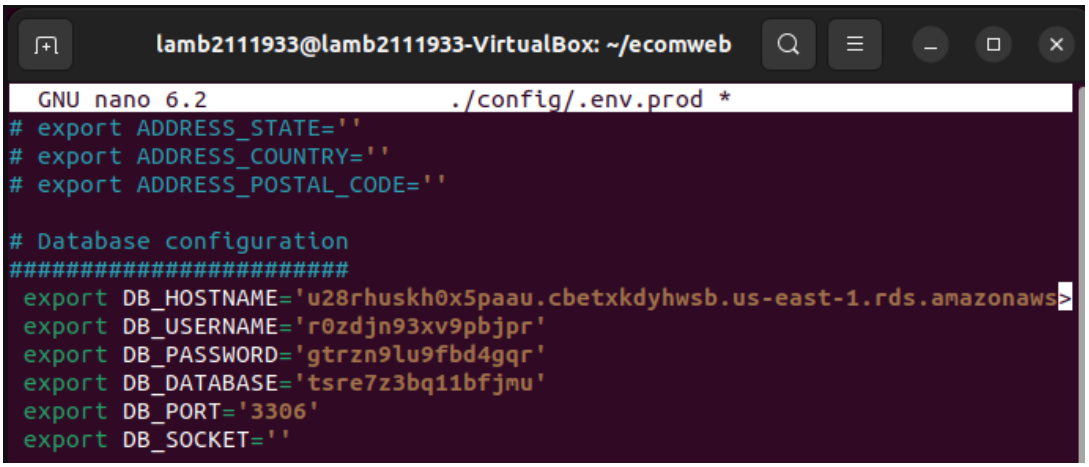
```
      $mysql          --user=username          --password=password --host=server_name database_name < ./config/db_dump.sql
```



Import data of the website

- Modify database configuration

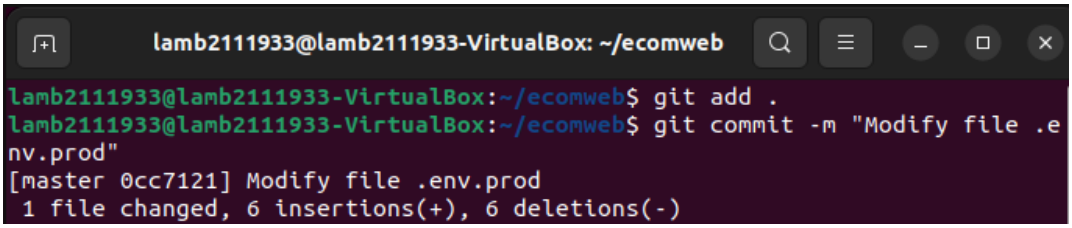
```
      $nano ./config/.env.prod

      #Modify the content of .env file
            export STORE_URL='<Your Heroku APP URL>'
            ...
            export DB_HOSTNAME='server_name'
            export DB_USERNAME='username'
            export DB_PASSWORD='password'
            export DB_DATABASE='database_name'
            export DB_PORT='3306'
```



Modify database **configuration**

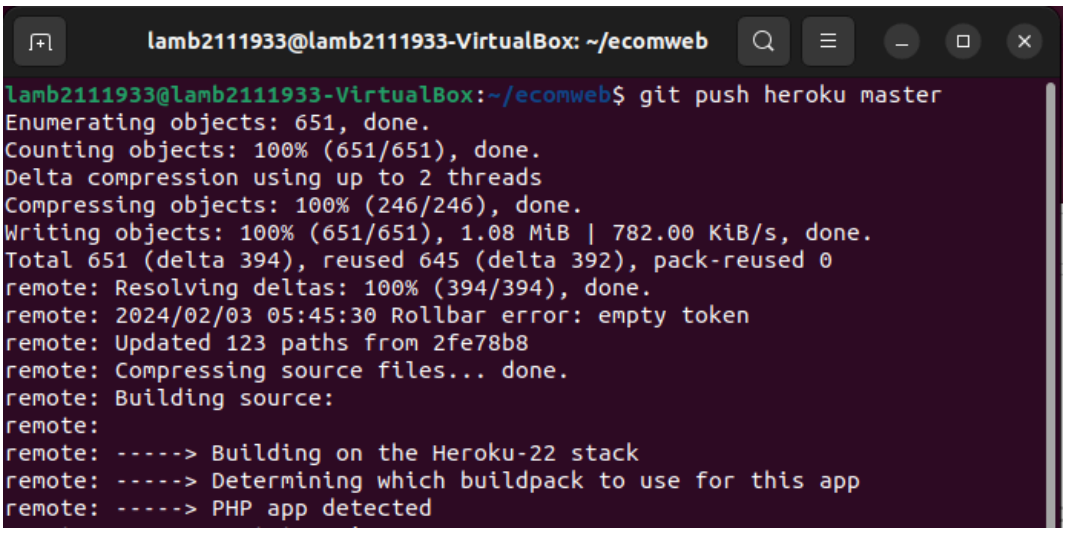
- Push the source code to heroku

```
$git add .
$git commit -m "Modify file .env.prod" # Cẩn thận lỗi dấu nháy
```



Add **modified files** to **init**

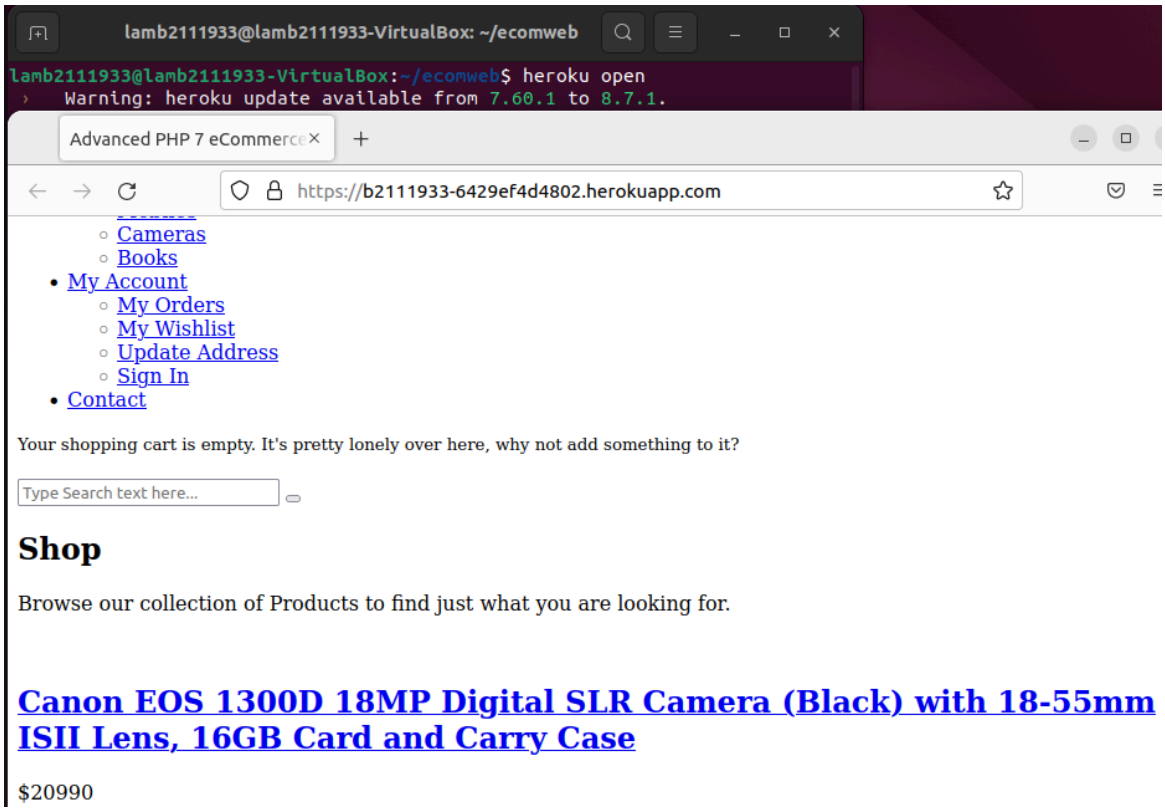
```
$git push heroku main
```



Push the source code to **heroku** (I used **master** branch instead)

- Launch your application

```
$heroku open
```



The result

This is my website URL: https://b2111933-6429ef4d4802.herokuapp.com/

(Take a screenshot of your website homepage on Heroku + and add your website URL here)

---END---