# Lab Session: Bluemix for Webpage Design

## Part 1: Hello World Webpage Design

### Preparation

1. A Bluemix account.

(Sign up: <https://developer.ibm.com/sso/bmregistration?lang=en_US&ca=dw-_-bluemix-_-cl-deploy-a-hello-world-webpage-to-bluemix-app-_-article> )

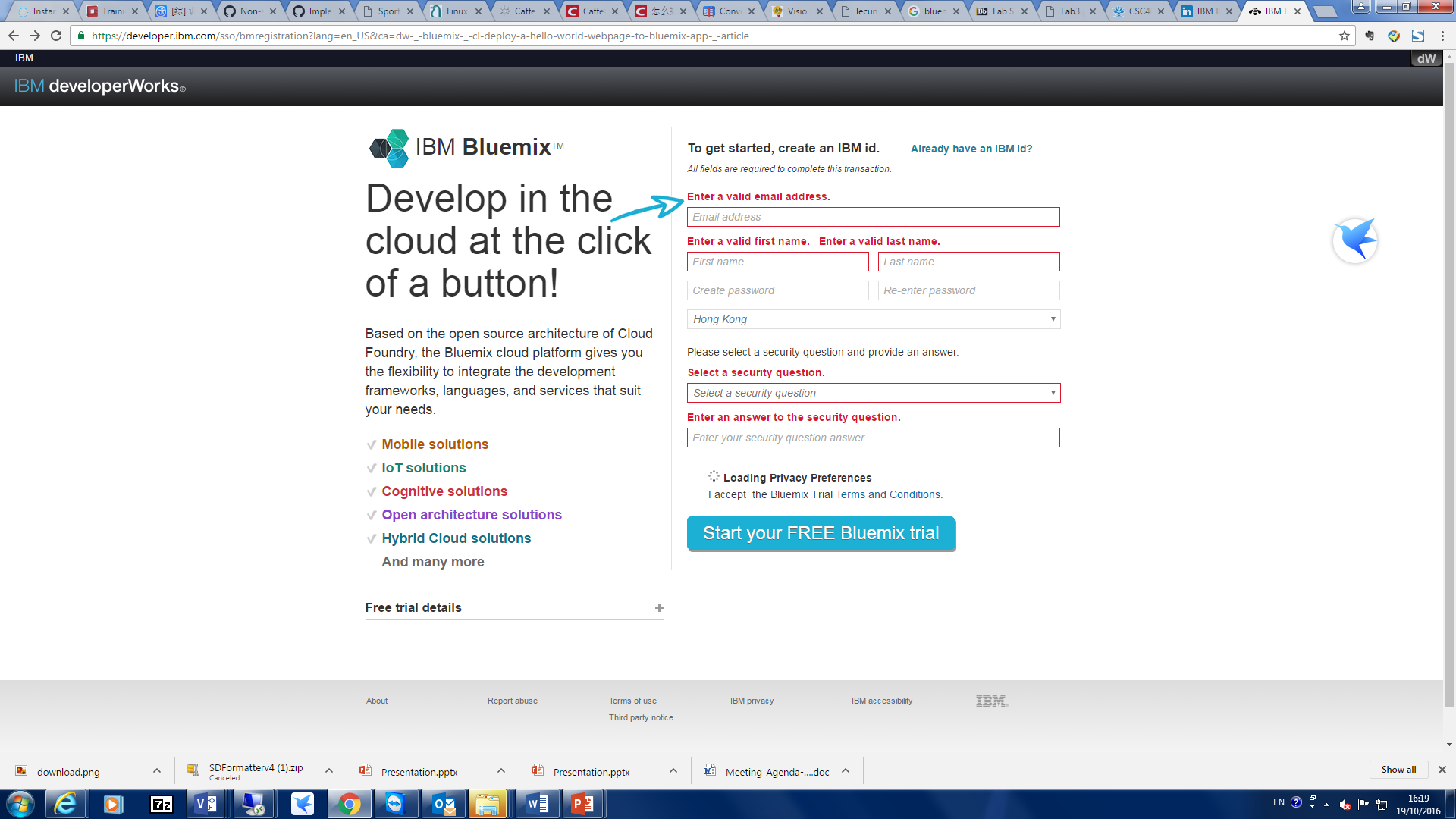


Fig.1 Sign-up page

***Please remember to confirm the account in your email.***

1. A GitHub account.

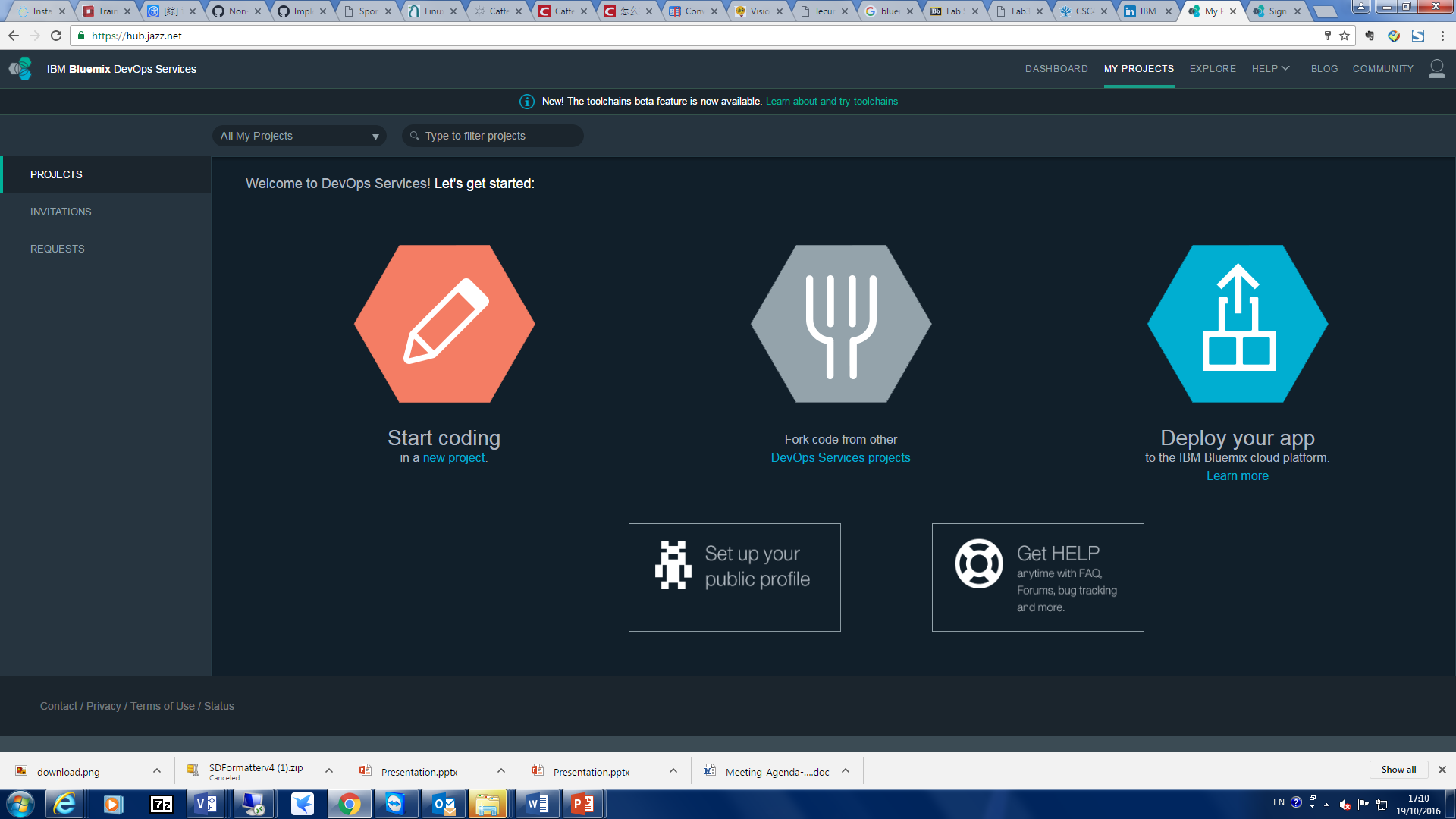
(Sign up: <https://github.com/> )

***Please remember to confirm the account in your email.***

1. Latest version of Chrome, Firefox, Internet Explorer, or Safari. The following steps are implemented on Chrome browser.

### Step 1 Create a project

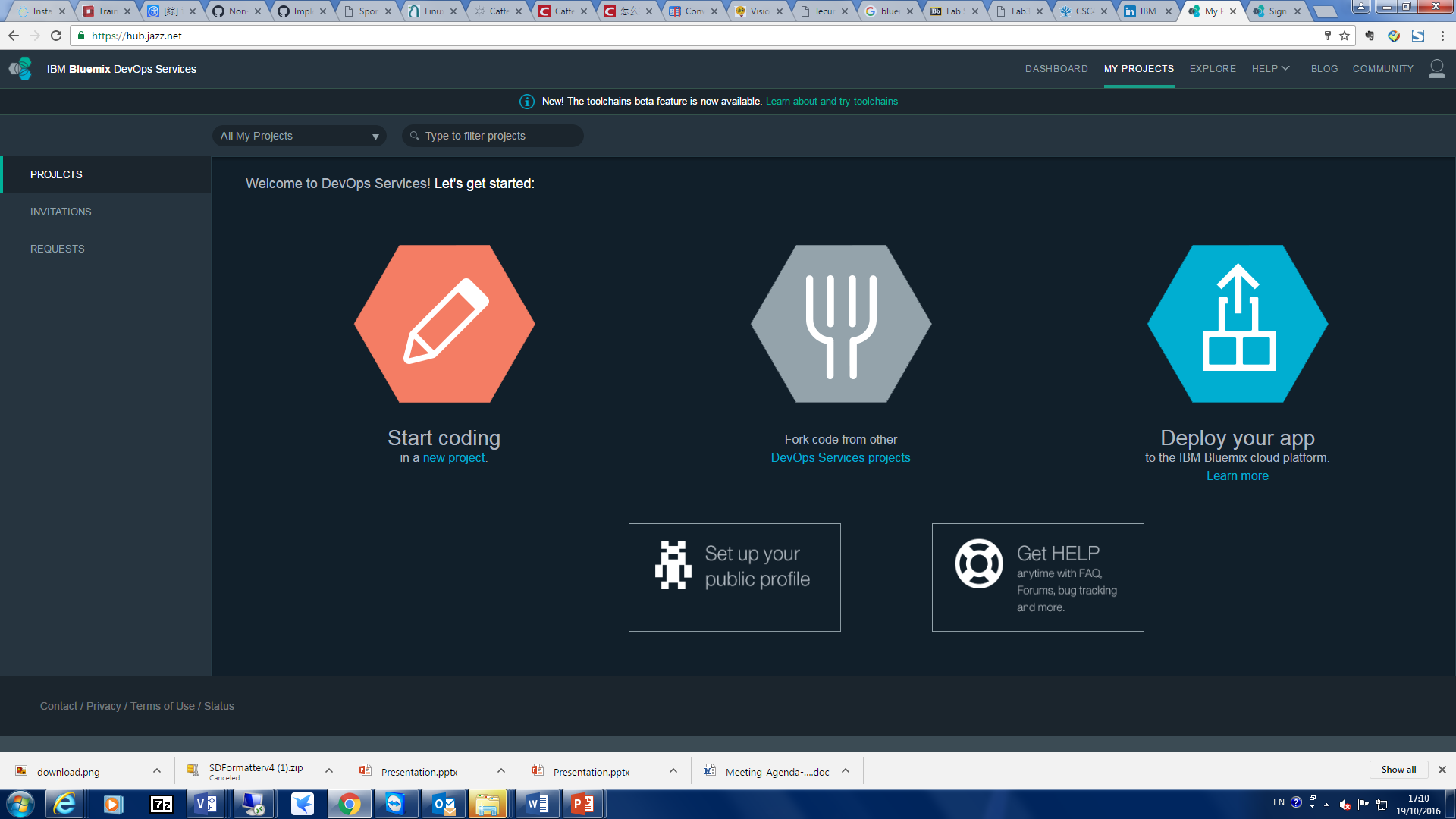
1. Log in to [IBM Bluemix DevOps Services](https://hub.jazz.net/?utm_source=dw&utm_campaign=bluemix&utm_content=cl-deploy-a-hello-world-webpage-to-bluemix-app&utm_medium=article). ( <https://hub.jazz.net/> )
2. Before you create a new project, you need to set up the environment. Click **Dashboard** in the top navigation.



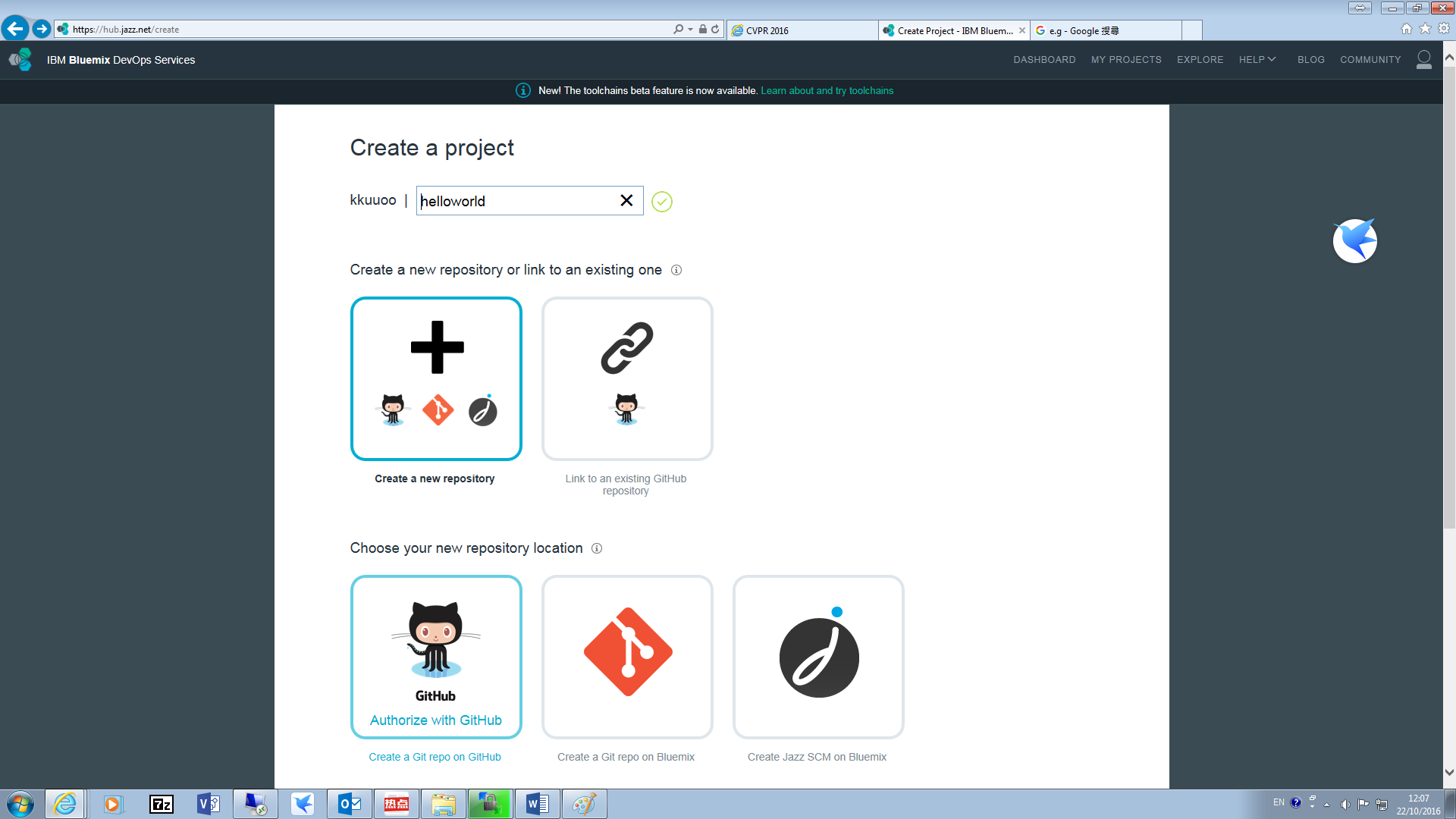
Input the **organization name** and **space name** according to the guide.



1. After you have successfully create the space, go back to previous page. Click **My Projects** in the top navigation.



1. Click **new project** under the start code icon.
2. On the "Create a project" page:
3. Type a name for your project.
4. Choose a source control option. You could link to your GitHub account.

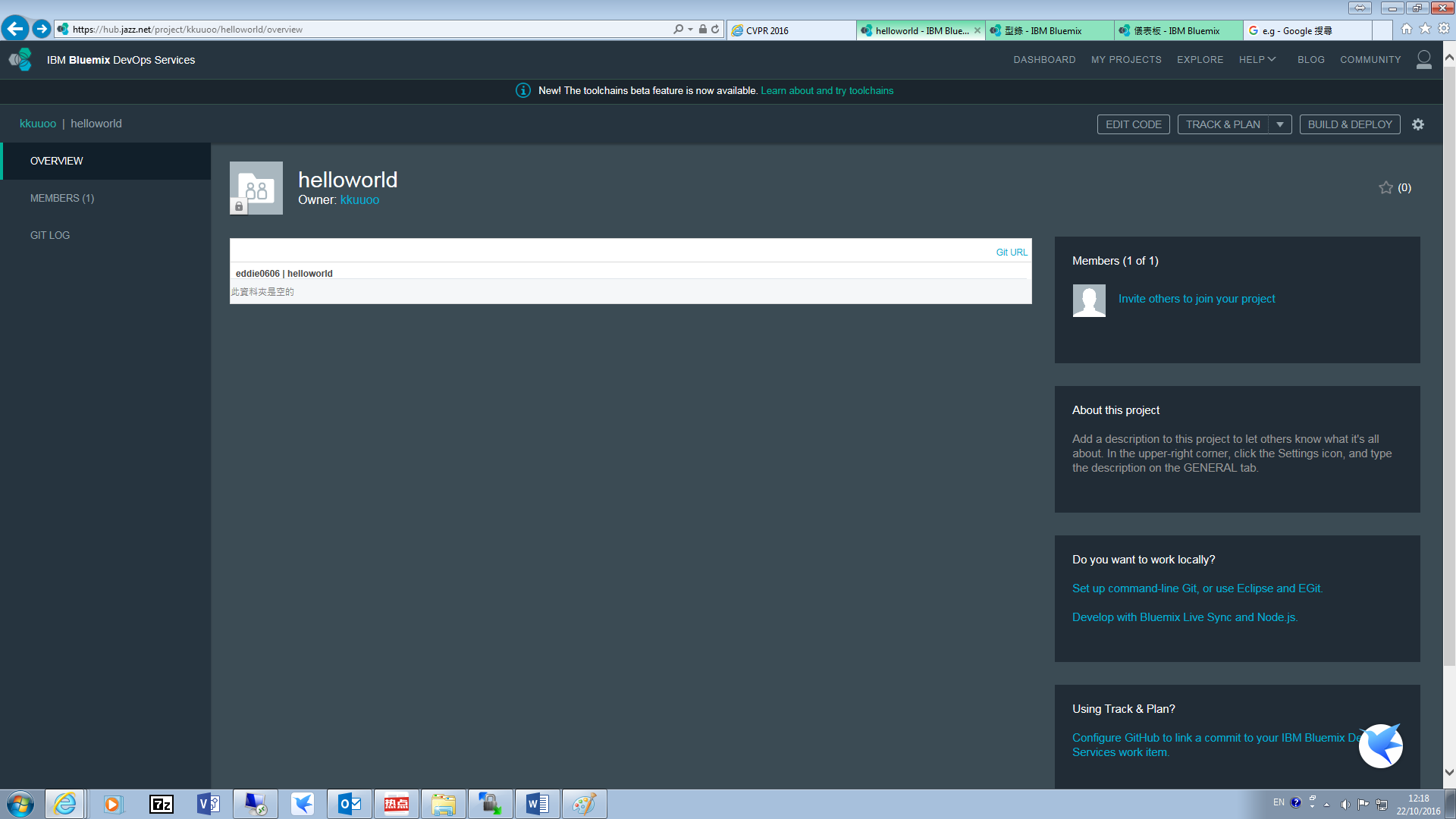
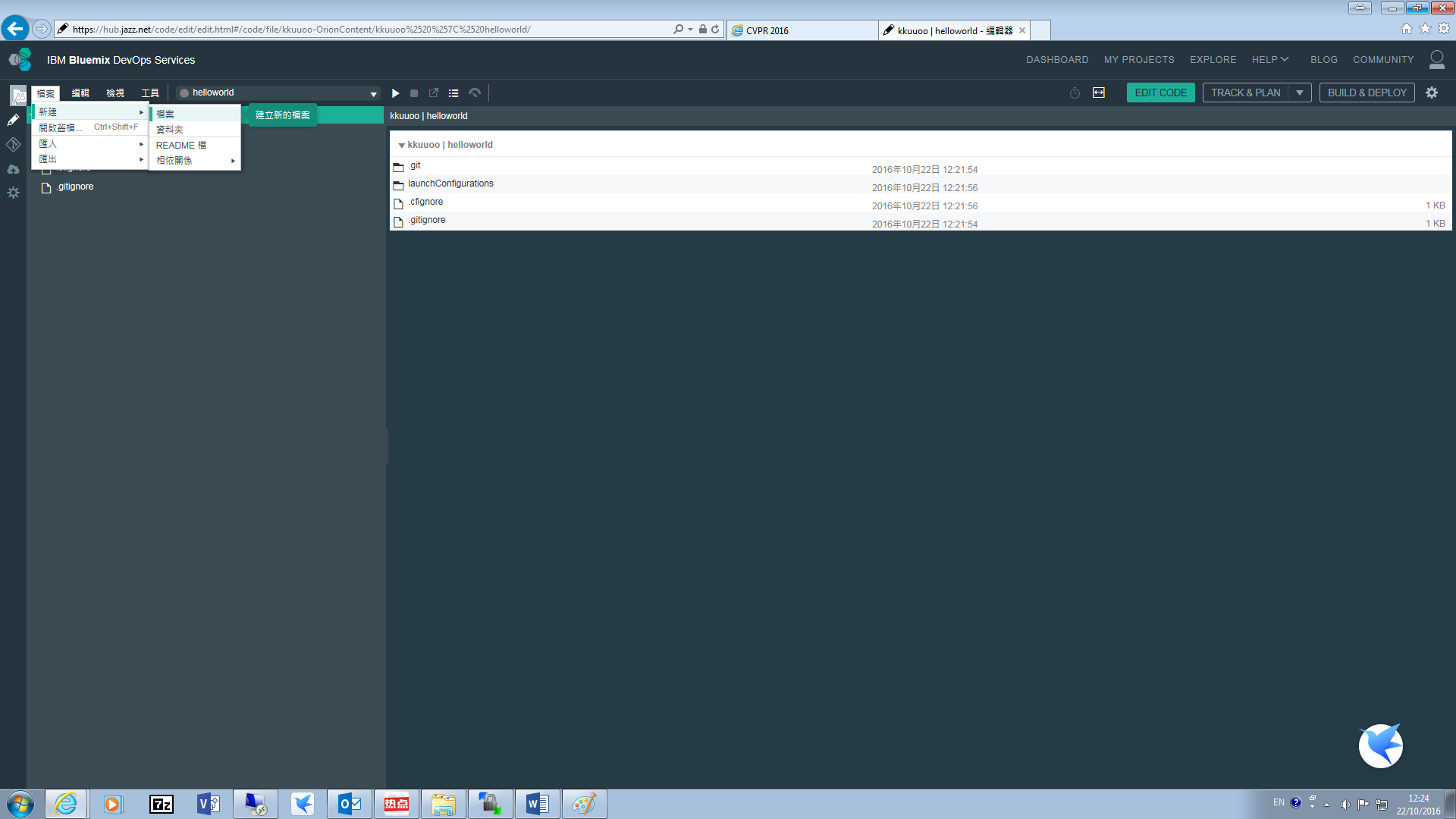


1. Tick the **Make this a Bluemix Project** box.

**Make sure you check this box or you cannot publish your webpage.**

1. Click **CREATE**. If you cannot create, check the space you create in dashboard setting.

### Step 2 Create a simple webpage

1. You need to do some coding job in your project. On your project’s overview page, click the **EDIT CODE** button. 
2. The Web IDE will open. A few files will be created for you automatically. In the Web IDE, select File > New > File. The file is highlighted and you can name it **index.html**. 
3. In the editor area on the right, paste the following:

<html>

<body>​

Hello, Bluemix World!​

</body>​

</html>​

### Step 3. Create a manifest

A *manifest* is a file that tells Bluemix how to deploy the application. It’s possible to deploy using the launch configuration wizard in the Web IDE without creating a manifest, but we’ll create a very simple manifest so that your app can be easily deployed using *either* the Web IDE or the CF Command Line tool.

1. In the Web IDE, select **File** > **New** > **File**. The file is highlighted so that you can name it. Enter **manifest.yml**.
2. In the editor area on the right, paste the following:

applications:​

- buildpack: https://github.com/cloudfoundry/staticfile-buildpack.git​

host: helloworld-html-${random-word}​

name: helloworld-html​

memory: 64M​

stack: cflinuxfs2​

The host is essentially the URL of your application. The host you choose must be unique across all of Bluemix, ***so replace ${random-word} with a unique string, such as your initials****.*

Each line in the manifest provides important instructions to Bluemix about how to deploy your app.

* The **buildpack** contains the necessary framework and/or runtime support for Bluemix to run your application. For this application, we're using a [static file buildpack](https://github.com/cloudfoundry/staticfile-buildpack) that is hosted on GitHub.
* The **host** is essentially the URL of your application. The host you choose must be unique across all of Bluemix. We added the ${random-word} keyword so that your deployment will not collide with the deployments of others who follow this tutorial. You can customize the host to be whatever you like.
* The **memory** refers to how much memory you are allocating to your app in your Bluemix space. Memory is not a required part of a manifest, but it is included in this example so that the app does not consume a large portion of the memory allocated to your Bluemix space.
* The **stack** is defined by Cloud Foundry as "a prebuilt root filesystem (rootfs) which works in tandem with a buildpack and is used to support running applications." The static file buildpack we're using requires the cflinuxfs2 stack. Bluemix currently defaults to an older stack, so we need to specify the stack as cflinuxfs2. Most buildpacks will not require you to specify a stack.

READ:[Deploying with Application Manifests](http://docs.cloudfoundry.org/devguide/deploy-apps/manifest.html)

### Step 4. Deploy and open the app

Now that you have a webpage and a manifest created, you’re ready to deploy!

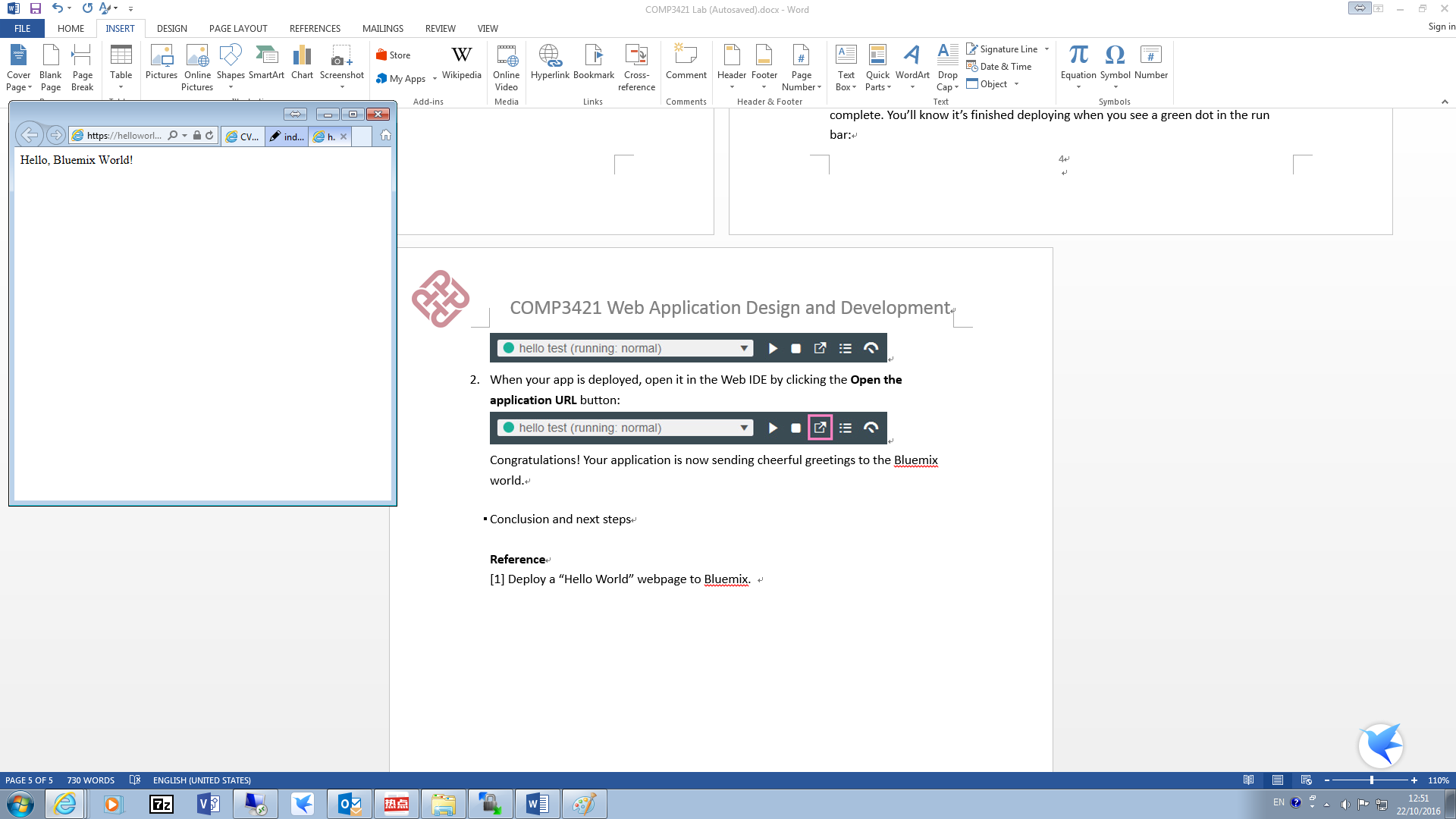
1. In the Web IDE, click the **Deploy** button:txt

Your application is now deployed to Bluemix. The process takes about a minute to complete. You’ll know it’s finished deploying when you see a green dot in the run bar:

txt

1. When your app is deployed, open it in the Web IDE by clicking the **Open the application URL** button:txt

Congratulations! Your application is now sending cheerful greetings to the Bluemix world.



## Part 2. Ajax

***Please use Chrome browser in this part!***

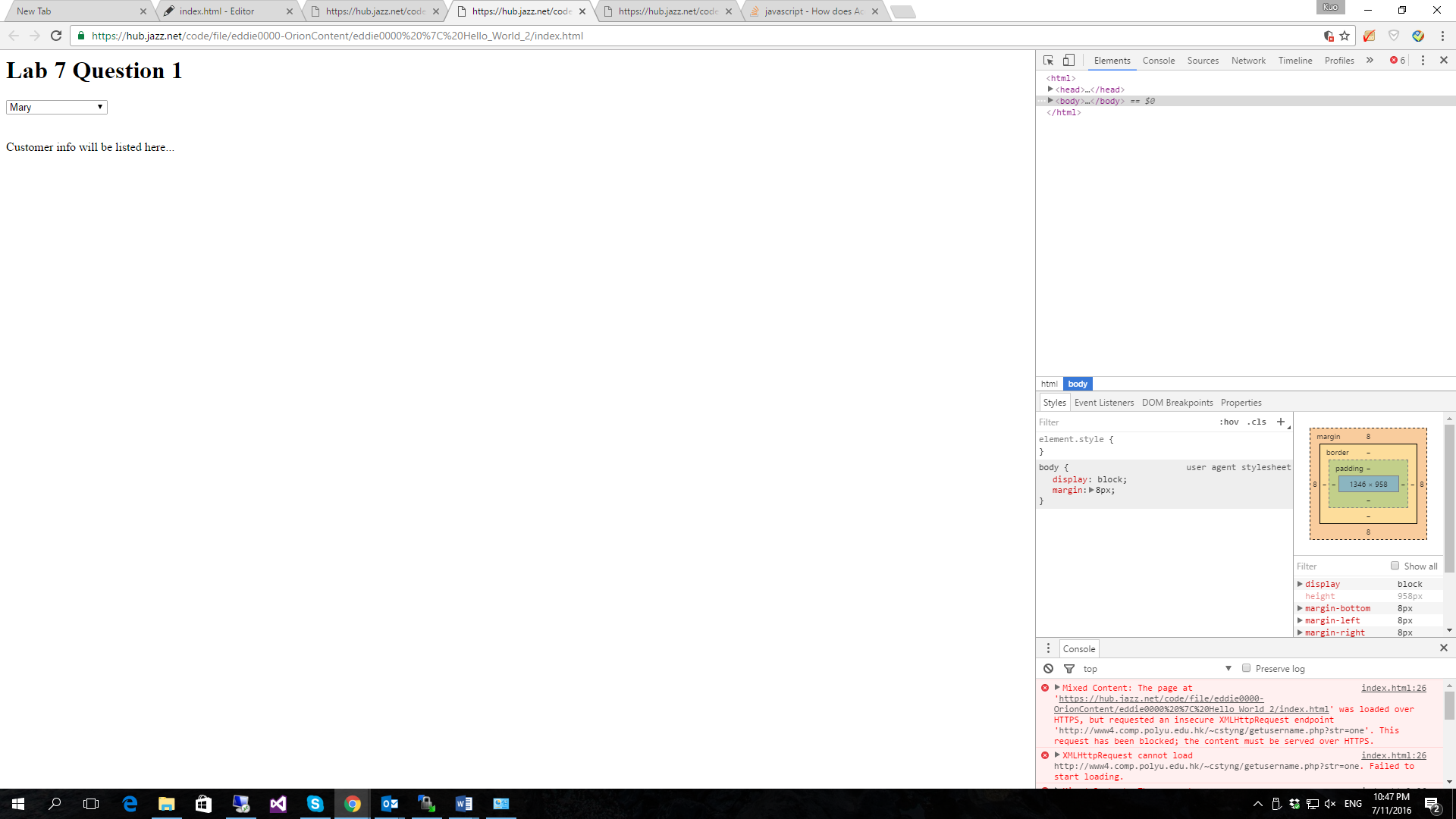
In this part, we will try the lab about ajax on Bluemix. The same manifest.yml created in part 1 is used in this section. You could implement the following examples on your hello\_world project.

### Example 1. Retrieve Customer Information

***Problem:*** Retrieve the information after selecting a customer.

***Solution***:

1. Import html files from your computer**. File 🡪 Import 🡪 1.html**
2. Rename 1.html to index.html and debug the webpage by **View🡪Open with🡪Web Browser.** If you are using chrome, you could **right click🡪 inspect** to check the error in the page.



It shows the request is blocked and because the website is loaded with https, but requested an insecure endpoint.

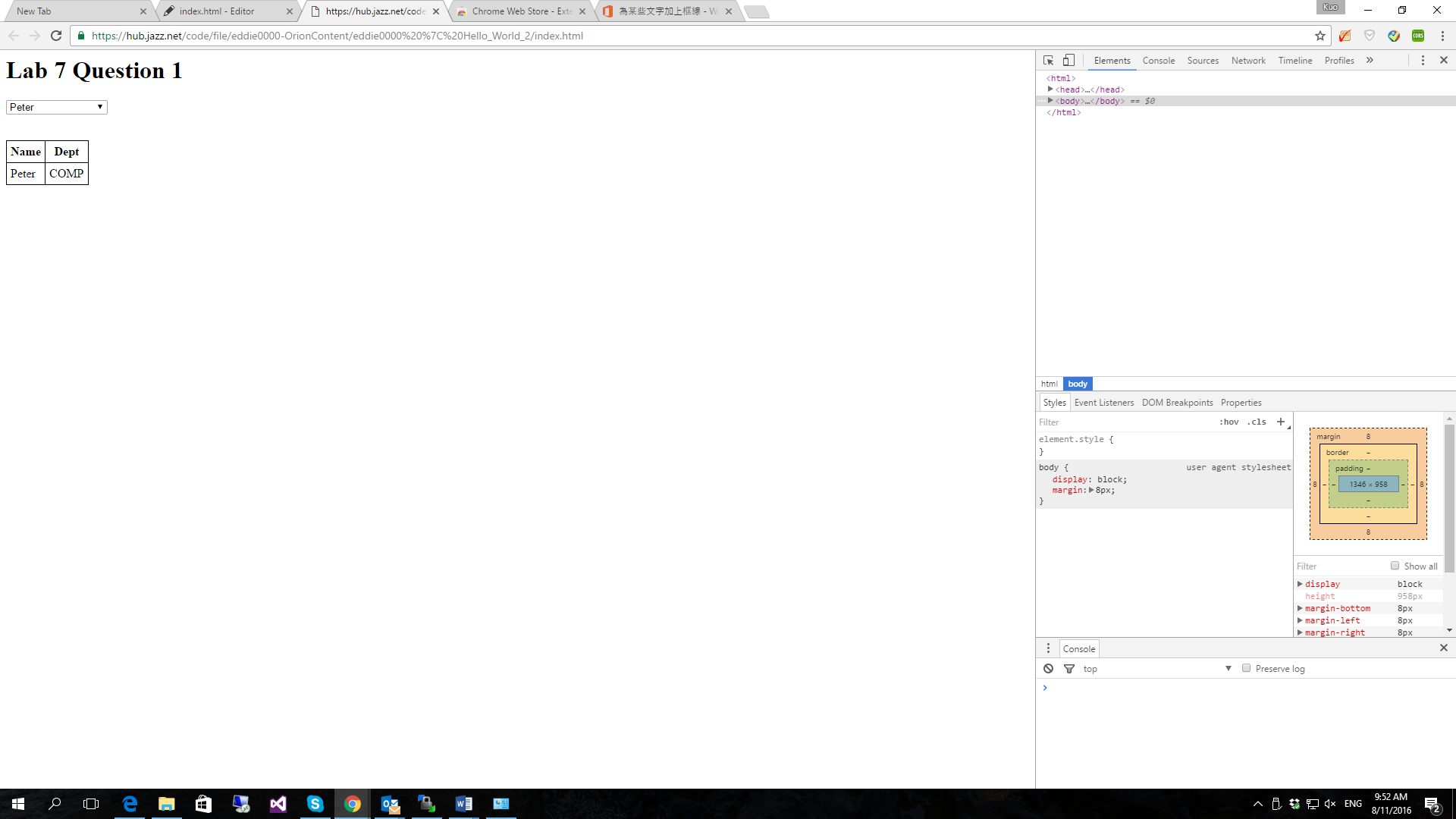
The solution is changing

xhttp.open("GET", "http://www4.comp.polyu.edu.hk/~cstyng/getusername.php?str="+str, true);

to

xhttp.open("GET", "https://www4.comp.polyu.edu.hk/~cstyng/getusername.php?str="+str, true);

1. Due to the same origin policy, it needs an extension Cross-Origin XMLHttpRequest to allow you request to any sites with ajax from any source. (https://chrome.google.com/webstore/detail/allow-control-allow-origi/nlfbmbojpeacfghkpbjhddihlkkiljbi?hl=en-US). Now the webpage could display correctly as below. You can check your webpage without deploy. Select the html file, **View🡪Open with🡪Web Browser.**

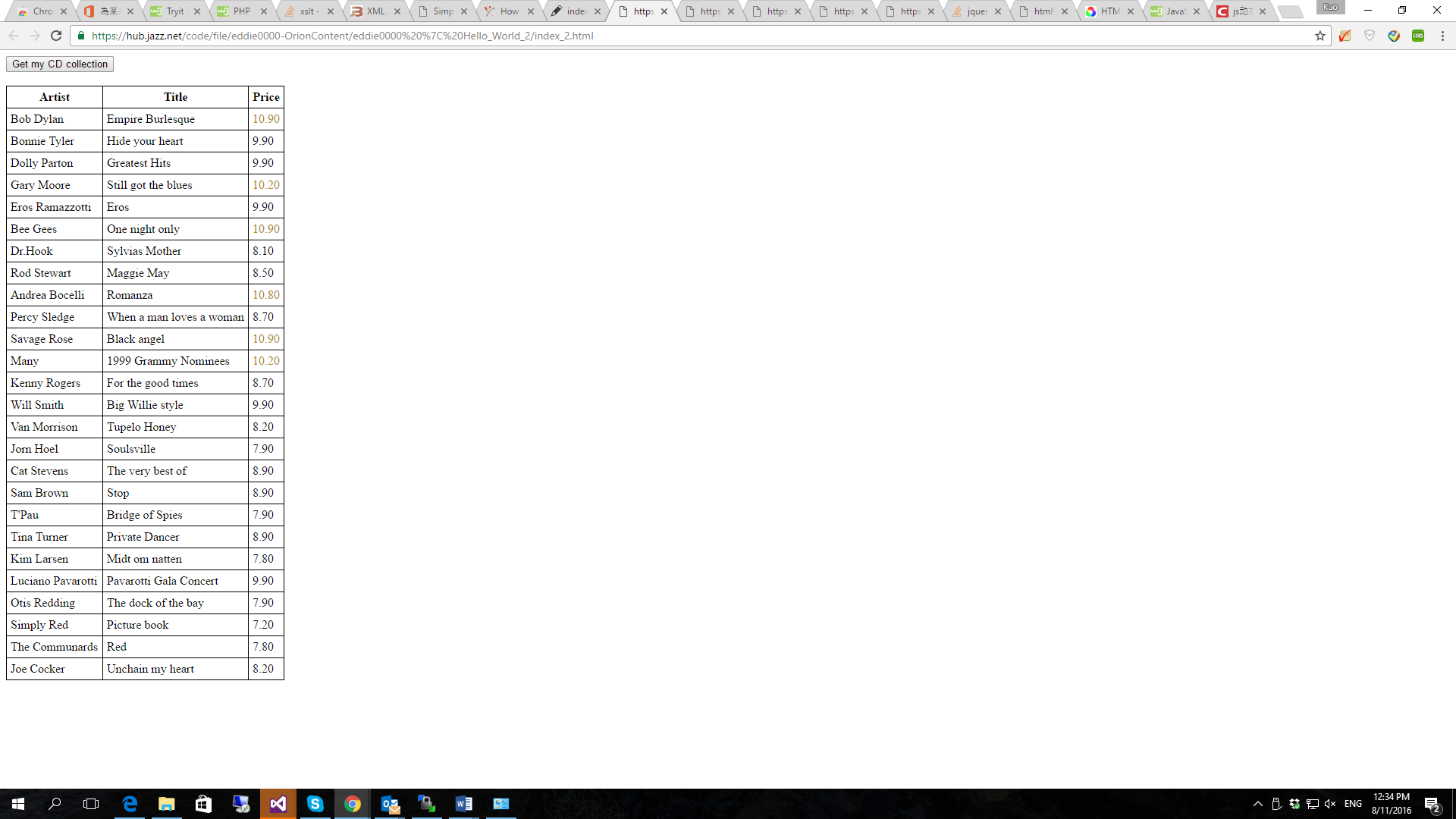


### Example 2: Design a XML and Ajax program

1. Load 2.html to your project**. File🡪Import🡪2.html**.
2. Load cd\_catalog.xml to your project. **File🡪Import🡪cd\_catalog.xml**.
3. Select 2.html file. **View🡪Open with🡪Web Browser**. You can check your webpage.

Please do the following exercises:

* Display the prices of the CDs in the third column.
* If the price is higher than 10 then highlight the table cell in yellow.



**Refence Code:**

<!DOCTYPE html>

<html>

<style>

table,th,td {

border : 1px solid black;

border-collapse: collapse;

}

th,td {

padding: 5px;

}

</style>

<body>

<button type="button" onclick="loadXMLDoc()">Get my CD collection</button>

<br><br>

<table id="demo"></table>

<script>

function loadXMLDoc() {

var xmlhttp = new XMLHttpRequest();

xmlhttp.onreadystatechange = function() {

if (this.readyState == 4 && this.status == 200) {

myFunction(this);

}

};

xmlhttp.open("GET", "cd\_catalog.xml", true);

xmlhttp.send();

}

function myFunction(xml) {

var i;

var xmlDoc = xml.responseXML;

var table="<tr><th>Artist</th><th>Title</th><th>Price</th></tr>";

var x = xmlDoc.getElementsByTagName("CD");

for (i = 0; i <x.length; i++) {

var tok = x[i].getElementsByTagName("PRICE")[0].childNodes[0].nodeValue;

if (tok > 10)

{

table += "<tr><td>" +

x[i].getElementsByTagName("ARTIST")[0].childNodes[0].nodeValue +

"</td><td>" +

x[i].getElementsByTagName("TITLE")[0].childNodes[0].nodeValue +

"</td><td><font color = B18904>"+x[i].getElementsByTagName("PRICE")[0].childNodes[0].nodeValue + "</font></td></tr>";

}

else

{

table += "<tr><td>" +

x[i].getElementsByTagName("ARTIST")[0].childNodes[0].nodeValue +

"</td><td>" +

x[i].getElementsByTagName("TITLE")[0].childNodes[0].nodeValue +

"</td><td><font color = 000000>"+x[i].getElementsByTagName("PRICE")[0].childNodes[0].nodeValue + "</font></td></tr>";

}

}

document.getElementById("demo").innerHTML = table;

}

</script>

</body>

</html>

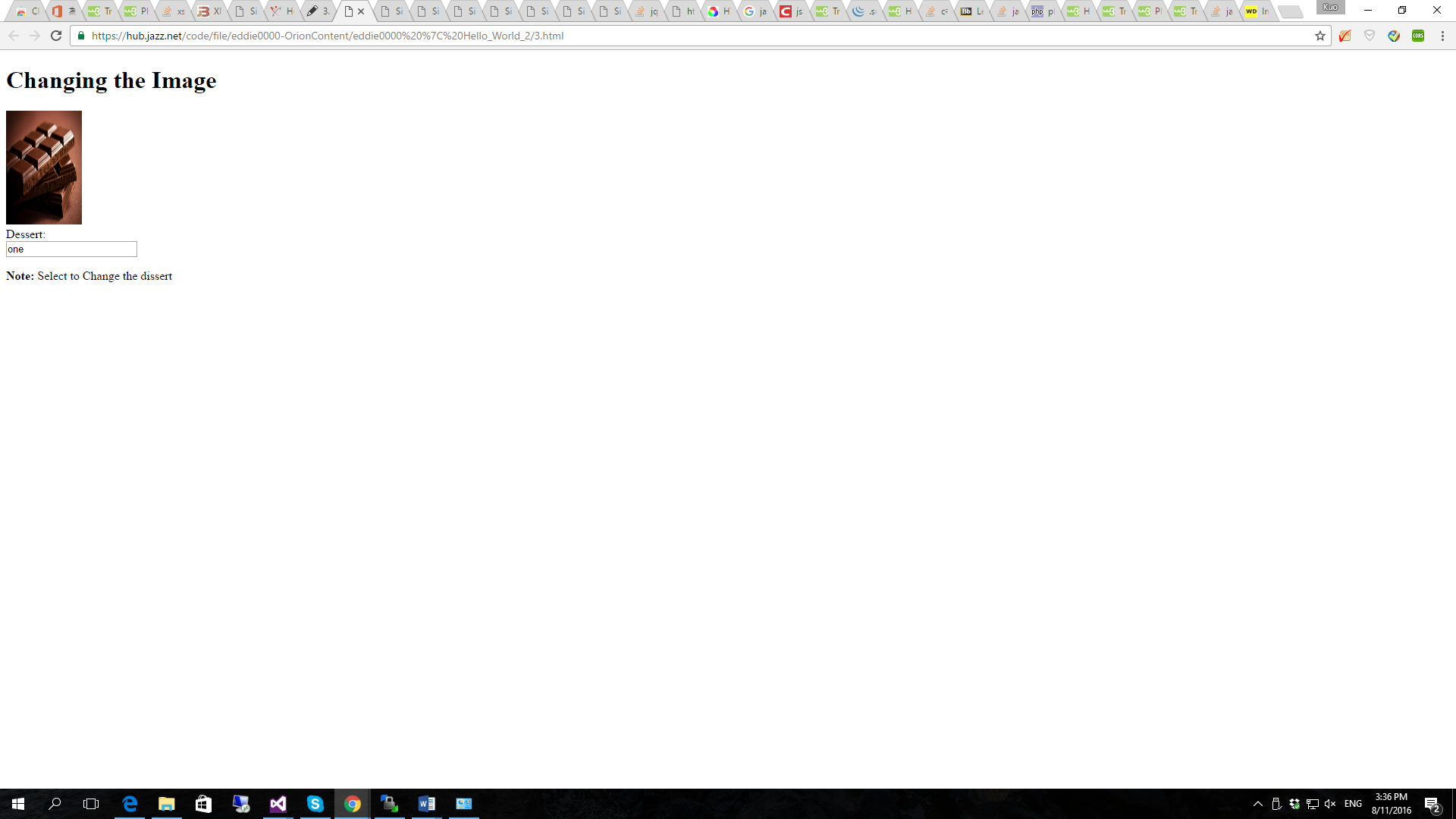
### Example 3: Textbox and Image

**Problem:** Provide an html file with an ajax program, which will prompt a user to enter ‘one’, ‘two’ or other character strings with a textbox. According to the input string, a fruit photo will be shown without refreshing the webpage. The 3.html will help you to solve the question.

**Solution:**

* + 1. Create a folder named “Image”, and put all three images into the folder.
    2. Import 3.html to your project.

In this part, you need to create a textbox instead of the option list in the original file. The final result looks like the image below.



Reference code:

<!DOCTYPE html>

<html>

<head>

<title>Simple Image change</title>

</head>

<body>

<h1>Changing the Image</h1>

<img src="" id="myImage" width="100" height="150">

<form action="">

Dessert:<br>

<input type="text" id="myText" name="str" value="peach" onchange="showDessert();">

<br>

</form>

<script>

**function showDessert() {**

**var str = document.getElementById("myText").value;**

**var xhttp;**

**if (str == "") {**

**document.getElementById("myImage").src = "";**

**return;**

**}**

**xhttp = new XMLHttpRequest();**

**xhttp.onreadystatechange = function() {**

**if (this.readyState == 4 && this.status == 200) {**

**document.getElementById("myImage").src ="Image/"+ this.responseText;**

**}**

**};**

**xhttp.open("GET", "https://www4.comp.polyu.edu.hk/~cstyng/test1.php?str="+str, true);**

**xhttp.send();**

**}**

</script>

<p><strong>Note:</strong> Select to Change the dissert</p>

</body>

</html>

**Reference**

[1] Deploy a “Hello World” webpage to Bluemix. <http://www.ibm.com/developerworks/cloud/library/cl-deploy-a-hello-world-webpage-to-bluemix-app/index.html>

# [2] [“No 'Access-Control-Allow-Origin' header is present on the requested resource”](http://stackoverflow.com/questions/20035101/no-access-control-allow-origin-header-is-present-on-the-requested-resource) http://stackoverflow.com/questions/20035101/no-access-control-allow-origin-header-is-present-on-the-requested-resource