



Edge Hill University

The Department of Computer Science

Applicant Visit Day

Getting to grips with Code!

Tutor: Hafeez Ullah Amin, PhD
Email: aminh@edgehill.ac.uk

Applicant Visit Day

Overview & Aims

The aim of the online session:

- Help you understand the importance of coding.
- Help you understand how coding relates to our degree programmes.
- Introduce some basic coding terminology.
- Give you the opportunity to solve some coding challenges for yourself.

Activities

Activity one – Connecting to Google Collaboratory

The Collaboratory is a fully functional software development environment that runs in the cloud. You can use the Collaboratory to write and execute Python software. This is a useful environment for Python beginners, as it abstracts away many of the complex aspects associated with learning to program. To use the Collaboratory, all you'll need is a google account, e.g. a **Gmail Account**. When you login to the Collaboratory, it creates a computer just for you to work with. The environment will allow you to create *Python code*, and execute it. The environment is special, as it allows code to be executed in an “interactive fashion”. This means you don't need to write a complete source code file. Instead you can execute individual commands one at a time. This is great for learning.

Pre-requisites

1. To complete the activities described here, you need to install the Google Chrome web browser first (if it is not installed). **Google Chrome** is available for free, thus if you don't have it, it is easy to install. You can find the Chrome installation file for your operating systems via the link below:

https://www.google.com/intl/en_uk/chrome/

Once Chrome is installed, you'll be able to use the tools described in this document without any issues.

2. Create **Gmail Account** (if you don't have it)
Go to Google Account sign-in page, and Click **Create Account**

Where to find the Collaboratory?

Google Collaboratory can be found at the following URL:

<https://colab.research.google.com/>

Signing in to the Collaboratory is achieved via the following steps.

1. Connect to the URL: <https://colab.research.google.com/>.
2. Sign-in when prompted or by clicking the sign-in button which should be in the top-right-hand-corner of the browser window. Sign-in using your Google account.
3. Once signed into the Collaboratory, you likely be presented with this default page:

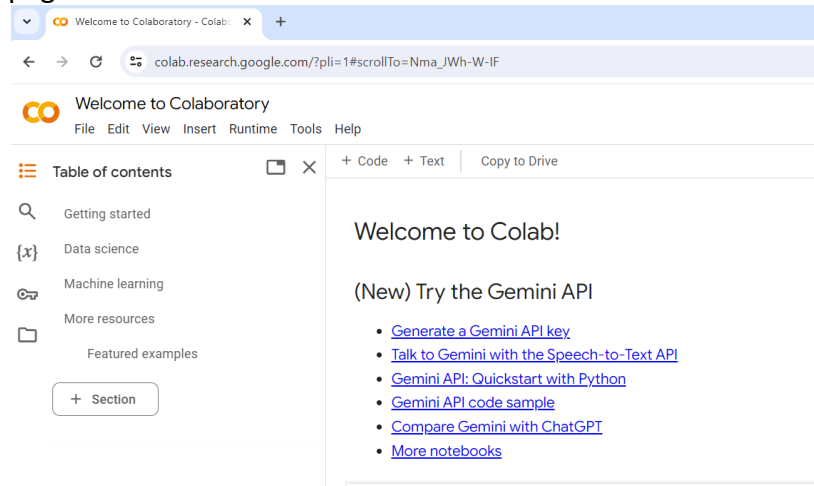


Figure 1. The default Google Collaboratory page.

Activity two – Creating a Notebook

The following steps will explain how you can create a new Collaboratory notebook.

1. The default page is great, but we want to create our own Collaboratory notebooks. Notebooks are files that contain the code and text we enter in to the Collaboratory environment. The notebooks are stored inside Google Drive and not on your local computer. Though you can download and upload notebook files. To create our own notebook, head to the menu bar at the top of the screen. Click:

File -> New notebook.

2. This will create an empty notebook file called something like "Untitled.ipynb". Why not rename the notebook – click inside the box that holds the name of the notebook. Rename it to something more descriptive, e.g. MyFirstNotebook.ipynb. Don't alter the extension of the file (.ipynb) because that extension identifies the file type, and without it, the Collaboratory won't open it.

Finding your way around the notebook

Notebooks are comprised of cells. Cells can contain code, or they can contain text that explains the content of a code cell. Text cells can use rich text formatting – that means we can style the text, use bold or italic fonts etc. However the cell shown in Figure 2. Is a code cell. We know this because it has the “play” symbol shown to its left. This indicates it can be executed. Let’s try to write some code that can be executed.

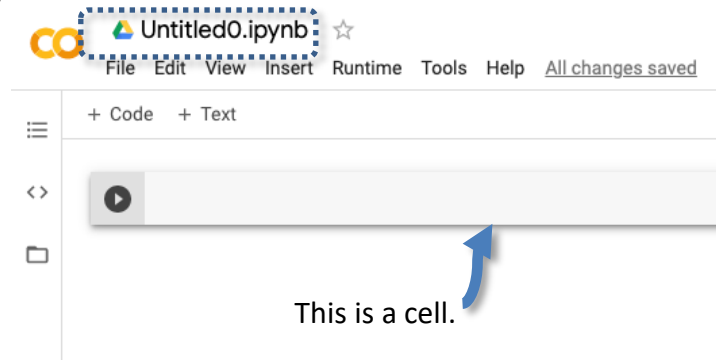


Figure 2. The area where you can rename the notebook, circled using a dashed blue line. It also shows the first cell created in the notebook.

1. Click to the right of the “play” button, directly into the cell. Type the following exactly:

```
print("Hi CIS1111")
```

When done, press the “play” button to see what happens.



Figure 3. The “play” symbol used to execute code.

2. When you press play, the Collaboratory executes the code you’ve just written. It then writes out the output of the code, onto the screen for you to see. This is shown in Figure 4. If you see this output then congratulations, you just wrote your first Python program. Ok, ok, it is a basic program.

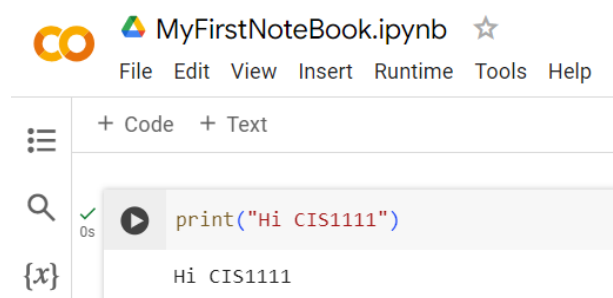


Figure 4. The code we’ve written and the output shown after executing it.

So why not write a program to do something a little more complicated? Why not try to calculate a math equation?

Type in the following exactly then press the “play” button:

```
x = 5 + 2 - 3
print(x)
```

3. If you’ve written the code from step 2 correctly, you should see the output shown in Figure 5. You just solved your first simple equation in Python – you resolved `x` to a value. The value of `x` has now been defined, meaning we can use it in other calculations. This is possible because the Collaboratory remembers the values you define in cells and stores them in memory. For example, following on from the code you just wrote, write the following exactly and then press the “play” button:

```
y = x + 8
print(y)
```

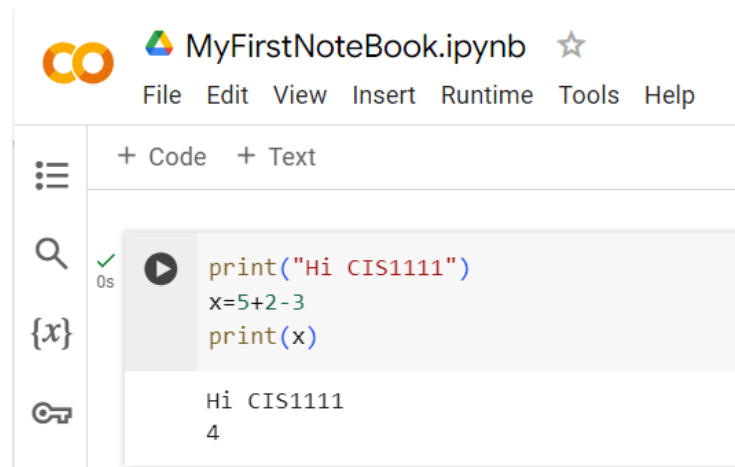


Figure 5. The code we’ve written and the output shown after executing it.

4. If you’ve written the code correctly you should see the output shown in **Figure 6**.

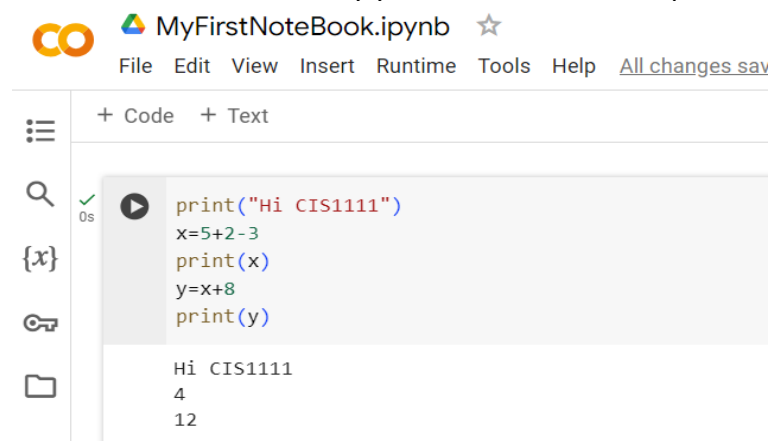


Figure 6. The code we’ve written – we now see the value of `y` printed out.

Now how about we create some new cells. You can create new code cells by clicking on the + Code button, and text cells by clicking the + Text button. These are shown in **Figure 7**. Click these buttons a few times and see what happens.

5. **Figure 7** shows two code cells followed by a text cell. We can see how the text cell has editing options, that allows us to customise our writing. Note how cells can be moved up or down using the arrows on the right hand side of the cell, shown in **Figure 7**. This is important, as the order of code cells matters.

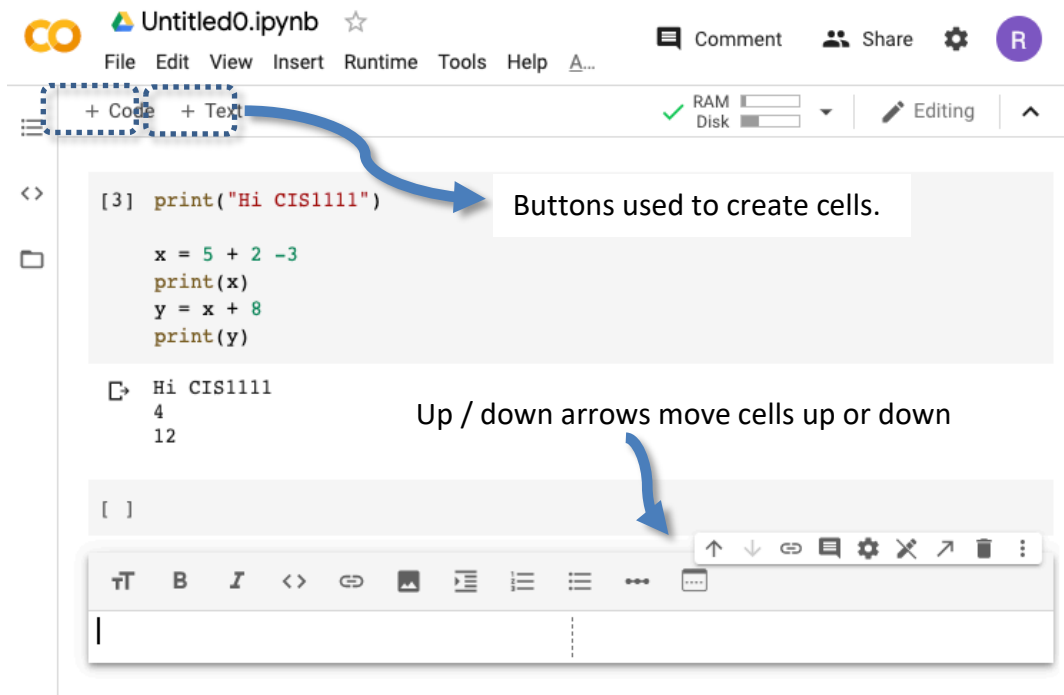


Figure 7. Here we see three cells – two for code and one for text. The text cell has editing options.

6. We can save our notebooks from the **File** menu. By default this saves them to Google Drive. However, you can also download the notebook to your local computer. We can also load notebooks created elsewhere and run their content. We'll do that next.

Activity three – Loading the activity Notebook

For today's activity I created a file called:

ApplicatVisitDay.ipynb

This file can be found in the Blackboard environment we're using (directions given on the day). Alternatively, you can download it here:

<https://github.com/hafeezullahamin/ProgrammingTasterEHU>

At the above link, click the `code`, then `Download zip` option.

We need to load these document in to the Collaboratory environment, so we can use it. To do this, we must do the following:

1. Download the `ApplicatVisitDay.ipynb` file and save it to your personal computer. Save it somewhere easy to remember, like your `Downloads` folder.
2. Once the file is downloaded, head back to the Google Collaboratory web-interface.
3. On the Collaboratory, click the `File` menu, and then select the `Open Notebook` option.
4. When the file chooser dialog appears, select the `ApplicatVisitDay.ipynb` file you saved to your computer in Step 1. Once you select the file, it will load.
5. Check that the file has loaded in your browser window – if not, ask for help.
6. Follow the instructions provided in the notebook and complete the challenges.

Activity four – Working through the Notebook

From here the notebook provides all the instructions you need. So please following along there.

Learning Checklist

By the end of this session you should be able to do the following in Google Collaboratory:

- Create a new notebook.
- Rename a notebook.
- Write code in a cell and execute that cell.
- Add new text or code cells.
- Move cells up and down.
- Save your notebook to Google Drive.
- Download a notebook to your local computer.
- Load new notebooks and save them in your Colab environment, so you can interact with them.

Useful Links

1. Python 3.8 API [documentation](#).
2. A simple W3Schools [Python tutorial](#).
3. A Python [programming course](#) provided by the creators of the language.
4. A [guide](#) to writing Python code and comments.

Reading Materials

Here are some example texts that we recommend to our students when learning Python. Maybe these will be of some help to you too.

Beazley, D. M. & Jones, B. K., 2013, "Python Cookbook", 3rd Edition, [available in the university library](#), last accessed 30/12/2019.

Dawson, M., 2009, "Python Programming for the Absolute Beginner", 3rd Edition, [available in the university library](#), last accessed 30/12/2019.

Downey, A., 2015, "Think Python: How to Think Like a Computer Scientist", 2nd Edition, Version 2.4.0, [available on-line](#), last accessed 09/06/2020. An interactive version is available [here](#).

Dusty, P., 2010, "Python 3 Object Oriented Programming", 1st Edition, [available on-line via university library sign-in](#), last accessed 30/12/2019.

Lutz, M., 2011, "Programming Python", 4th Edition, [available in the university library](#), last accessed 30/12/2019.