CHAPTER
ONE

IN BRIEF

The e-learning Python for Ocean Mapping (ePOM) project provides collections of Jupyter notebooks to learn Python with a focus on Ocean Mapping.

1.1 Rationale

Given the variety of backgrounds of Ocean Mapping students and the fact that some of them have limited programming experience, there is a need to ensure a minimum common level of programming skills. Python is a natural choice as a means to provide those skills. The language is simple to learn for inexperienced programmers, increasing popular, and freely available.

Based on such considerations, the ePOM project was established.

The teaching goals are to familiarize Ocean Mapping students with key programming concepts and, at the same time, to teach them programming habits and skills that can be potentially translated to other programming languages. Given the popularity of Python, there is a lot of online material about the language. This can be overwhelming. Thus, ePOM aims to help the students to navigate their way through the huge amount of help and online resources. Last but not least teaching goal, ePOM aims to provide enough basic Python skills to ease the first lab assignments.
HOW TO ACCESS THE NOTEBOOKS

The e-learning Python for Ocean Mapping (ePOM) notebooks can be accessed in the three alternative ways:

- By accessing the CCOM/JHC servers.
- By installing NOAA Pydro (Windows-only).
- By cloning the ePOM’s GitHub repositories.

2.1 Access to the CCOM/JHC ePOM Servers

This option relies on the CCOM/JHC ePOM servers. It is the easiest way to obtain access to the ePOM notebooks. You only need:

- An updated, modern browser (e.g., Mozilla Firefox, Google Chrome, Apple Safari).
- An Internet connection.

The CCOM/JHC ePOM servers are configured with a Python environment that has JupyterHub and all the required packages to run the notebooks. In order to access the CCOM/JHC ePOM servers, you need an username. For obtaining an username, write to epom@ccom.unh.edu.

Once obtained the username, read the following instructions:

- Programming Basics with Python - Quick Start
- Introduction to Ocean Data Science - Quick Start

2.2 Installation of NOAA Pydro

NOAA Pydro is a free suite of open-source software tools used to support hydrography and cartography. You can download the NOAA Pydro installers from the NOAA Pydro download page. Pydro only works on Windows.

To have a full NOAA Pydro installation, download the following two files into the same folder, then run the PydroSetup installer. It will automatically find and install the PydroSupplemental installer.

The ePOM notebooks are part of NOAA Pydro. Once that you have installed NOAA Pydro, you can access the notebooks through a menu entry in Pydro Explorer (Fig. 2.1).

Pydro Explorer also provides the option to remove all the applied changes and thus restore the initial notebooks.
Fig. 2.1: The **Programming Basics with Python** menu entry in Pydro Explorer.
2.3 Cloning the GitHub Repositories

This third option assumes some previous knowledge on how to use git and create a NOAA Pydro environment. A common use case for this option is the willingness to contribute to the ePOM project by proposing improvements. You can access (and clone) the ePOM’s GitHub repositories at the following urls:

- Programming Basics with Python
- Introduction to Ocean Data Science
3.1 Introduction

Programming Basics with Python is a collection of notebooks that aims to teach you programming basics in Python, assuming zero knowledge on the subject, in preparation for the labs and exercises in the Ocean Mapping courses.

You will not need to install Python on your computer, since the notebooks run on a CCOM/JHC server that has Python already installed.

Thus, what you need for this course is just:

- An updated, modern browser (e.g., Mozilla Firefox, Google Chrome, Apple Safari).
- An Internet connection.

To avoid visualization issues, please do not use Microsoft Edge, Microsoft Internet Explorer, or an outdated browser (e.g., Apple Safari is no longer updated for Windows). If you need to use one of these browsers, please contact us for specific instructions.

If you do not have access to a computer, you may use a tablet or a smartphone. In such a case, your user experience might be not optimal (due to the limited screen size and a few visualization issues).

3.2 First-time Access

If you landed on this page, you have likely got an email from CCOM/JHC. With the same email you should have also received:

- The following quick-access url to access the notebooks collection on the server: quick-access url.
- Your personal username to access the CCOM/JHC server.
Each time that you use the **quick-access url** with your *browser*, you will be welcomed by a login prompt (Fig. 3.1). The very first time that you follow the **quick-access url**, you have to put in the login prompt:

- In the **Username** field, the username that you received by email.
- In the **Password** field, a **strong password** of your choice.

**You need to remember the chosen password** since it will be used any time that you want to access again the CCOM/JHC *server*.

For any following access, use the **quick-access url**, your assigned **username**, and the **chosen password**.

### 3.3 Welcome Notebook

After the login to the *server*, a welcome *notebook* will be open (Fig. 3.2).

This *notebook* will provide you with:

- The minimal set of notions required to begin to use a *notebook*.
- Directions on how to access the other available *notebooks*. 
3.4 Support

If you experience difficulties in completing the Programming Basics with Python training, it is important to seek help from all sources before your arrival at CCOM/JHC, including fellow students and instructors. We are always willing to help!

For issues or suggestions related to the above directions, write to epom@ccom.unh.edu.

3.5 Supplementary Material

- Cheat Sheets
- Past Evaluations

Now you should be ready to go!
4.1 Introduction

Introduction to Ocean Data Science is a collection of notebooks that aims to teach you basic concepts in ocean data science.

The notebooks assume the level of Python knowledge described in Programming Basics with Python notebooks. If you are not familiar with these latter notebooks, you should start from Programming Basics with Python - Quick Start.

4.2 Quick-Access Url

If you have already executed the Programming Basics with Python notebooks, you should know the mechanism on how to access the CCOM/JHC server. If you are not, see First-time Access.

The following quick-access url gives access to the notebooks collection on the server: quick-access url.

4.3 Support

For issues or suggestions related to the above directions, write to epom@ccom.unh.edu.
Fig. 4.1: The login prompt that provides access to the CCOM/JHC server.
CHAPTER
FIVE

INTRODUCTION TO OCEAN DATA SCIENCE - SLIDES

• Object-Oriented Programming
• Data Visualization
• Scientific Computing
• Raster and Vector Data Formats

For comments or suggestions related to the above slides, write to gmasetti@ccom.unh.edu.
FUNDAMENTALS OF RESEARCH SOFTWARE DEVELOPMENT - SLIDES

- Introduction
- Integrated Development Environment
- Code Version Control

For comments or suggestions related to the above slides, write to gmasetti@ccom.unh.edu.
CHAPTER
SEVEN

SIGNAL CONDITIONING AND FILTERING - QUICK START

7.1 Introduction

Signal Conditioning and Filtering is a collection of notebooks that aims to teach you basic concepts in digital signal processing.

The notebooks assume the level of Python knowledge described in Programming Basics with Python notebooks. If you are not familiar with these latter notebooks, you should start from Programming Basics with Python - Quick Start.

7.2 Quick-Access Url

If you have already executed the Programming Basics with Python notebooks, you should know the mechanism on how to access the CCOM/JHC server. If you are not, see First-time Access.

The following quick-access url gives access to the notebooks collection on the server: quick-access url.

7.3 Support

For issues or suggestions related to the above directions, write to epom@ccom.unh.edu.
Fig. 7.1: The login prompt that provides access to the CCOM/JHC server.
8.1 How to open a terminal

Once logged on the CCOM/JHC server, a preliminary step to use git is the opening of a terminal (Fig. 8.1).

![Fig. 8.1: How to open a terminal on the CCOM/JHC server.](image)

8.2 How to create a new local repository

To create a new local git repository (Fig. 8.2):

- Create an empty folder with a name of your choice (e.g., `mkdir my_repo`).
- From within the created folder (e.g., `cd my_repo`), execute `git init`.

8.3 How to make the first commit

Before committing the code, you need to set name and email to identify who submitted a change:
Fig. 8.2: How to create a new git repository.

```
    git config --global user.email "my_email@example.com"
    git config --global user.name "My Name"
```

Now an example of commit (Fig. 8.3). We first create a file (i.e., a README file) with a basic welcome message, then:

- The created file is added to the tracked files with `git add`.
- The above change to the repository is committed with `git commit`.

```
    echo "Welcome to my repository!" >> README.rst
    git add README.rst
    git commit -m "First commit"
```

### 8.4 How to set and push to a remote repository

You first need to create a repository on a git hosting services such as GitHub (Fig. 8.4), GitLab, and BitBucket.

Once that a remote repository is created, an url similar to the following one will be available:

- `https://github.com/hydroffice/my_repo.git`

You set the above url as the remote repository (this operation usually only happens once, at the creation time):

```
    git remote add origin https://github.com/hydroffice/my_repo.git
```

Then, you push to the remote repository each time that you want to remotely store your changes:

```
    git push -u origin master
```

You will be asked to authenticate yourself on the hosting service of your choice (Fig. 8.5).

After that the push operation is completed, you should be able to visualize your updated code on the hosting service (Fig. 8.6).
8.4. How to set and push to a remote repository

![Example of first commit](image)

Fig. 8.3: Example of first commit.

![Example of remote repository created on GitHub](image)

Fig. 8.4: Example of remote repository created on GitHub.
Fig. 8.5: Example of how to push changes to a GitHub repository.

Fig. 8.6: Results on the GitHub repository after the push action for a first commit.
8.5 Other useful git operations

- To display all the changes for both tracked and un-tracked files:
  \[
  \text{git status}
  \]

- To display changes only to tracked files:
  \[
  \text{git diff}
  \]

- To add a specific file (e.g., “Lecture0.ipynb”) to the tracking for the next commit:
  \[
  \text{git add Lecture0.ipynb}
  \]

- To commit all local changes in tracked files (-a) with a message (-m "Latest changes")
  \[
  \text{git commit -a -m "Latest changes"}
  \]

- To download all changes from a remote repository (e.g., “origin”):
  \[
  \text{git pull origin}
  \]

- To discard all the local changes to tracked files. You will lose all the unpushed changes!!!
  \[
  \text{git reset -hard HEAD}
  \]

8.6 Useful references

- git Documentation
- GitHub’s git cheat sheet
- Tower’s git cheat sheet
• Programming Basics with Python - Cheat Sheet
• Matplotlib - Cheat Sheet (external link)
• Numpy - Cheat Sheet (external link)
PAST EVALUATIONS

- Programming Basics with Python
  - Evaluation 000 (see Solutions for Evaluation 000).
  - Evaluation 001 (see Solutions for Evaluation 001).
browser  Software that allows to retrieve and display web pages.

notebook  An interactive computing environment with text, code, and multimedia content in a single file.

programming  The process of creating a computer program to achieve one or more computing tasks.

Python  A very popular programming language with applications in widely different fields.

server  A computer which manages the access to network resources and/or services.
12.1 Authors

The e-learning Python for Ocean Mapping (ePOM) project is developed and maintained by:

- Giuseppe Masetti
- Rochelle Wigley
- Semme Dijkstra
- Jordan Chadwick

Feel free to contact us for comments and suggestions.
For bugs and feature requests: epom@ccom.unh.edu

12.2 Contributors

The following wonderful people contributed directly or indirectly to this project:

- Anne Hartwell
- Brandon Maingot
- Christos Kastrisios
- Dan Tauriello
- Jason Greenlaw
- Jeff Douglas
- Jenn Dijkstra
- Mike Smith
- Tomer Ketter
- Tyanne Faulkes
- Val Schmidt

**Note:** © University of New Hampshire, Center for Coastal and Ocean Mapping, 2019. All rights reserved.