



Pre-Event Instructions

Owner	Denisse Fierro Arcos
Tags	Training
Last edited time	@November 16, 2023 9:26 AM

During this event, we will go through some `R` and `Python` based notebooks showing how to access dataset available in the RIMReP DMS. These notebooks show how to perform simple analyses and create a variety of graphs, including maps, time series and bar plots.

To make the most out of this work, we recommend that you follow the instructions below **before** you attend the workshop. This will ensure you have all the necessary software installed in your computer so you are able to run these notebooks during the workshop.

NOTE: Installation of necessary software may take some time. To avoid delays, we highly recommend that you install all necessary software **before** the workshop.

Items to complete before workshop

- ☐ [Install conda/miniconda](#)
- ☐ [Check conda/miniconda installation](#)
- ☐ [Install `R`](#)
- ☐ [Install RStudio](#)
- ☐ [Install RTools \(Windows users only\)](#)
- ☐ [Check `R` and RStudio have been successfully installed](#)
- ☐ [Download/clone example notebooks](#)
- ☐ [Install `Python` packages](#)
- ☐ [Install `R` packages](#)

1. Installation instructions

The notebooks developed for this training event show how to access datasets available in the RIMReP DMS using two popular programming languages: `R` and `Python`.

You can copy these notebooks in your local machine by downloading or cloning our GitHub repository: <https://github.com/aodn/rimrep-training>. To run these notebooks, you will need to have installed `R`, `Python` and all libraries used in our example notebooks. In this section, we provide instructions on how to install everything you need.

Remember, you will need to make sure that you have installed all necessary software prior to the training as this may take some time.

1.1 Install conda/miniconda (includes **Python** installation)

Anaconda is a software distributor from where you can download **Python** and other programming languages, as well as hundreds of data science packages. Anaconda has two versions available: conda and miniconda, and their main difference is their size.

Conda will install Python and over 100 Python packages for the data science. The advantage of this is that you will likely have all the packages you need for a data science project and will not need to install additional packages. However, a large number of packages installed required a large space on your disk, so you will need to have at least 3 GB of disk space available. If you do not have that much space available in your disk, do not worry, you can install miniconda instead. Miniconda will still install Python and a small proportion of the packages included in conda.

If you are unsure about what option to choose, you can check out the articles [here](#) and [here](#). But we recommend you install miniconda, and you can download the latest version from this [link](#). If you prefer to download conda, download it from [here](#). Regardless of which option you choose, you will need to select the installer that matches your operating system (e.g., Windows, Linux, MacOS).

If you need any additional instructions on how to complete the installation, this [website](#) has installation instructions for every operating system.

[Back to checklist](#)

1.1.1 Checking conda/miniconda installation

The [Anaconda documentation](#) provides information on how to check your installation has been successful. Note that instructions vary slightly depending on your operating system.

To verify the installation, we will use the *Anaconda Prompt* (a command line interface for Anaconda) on Windows and macOS, or the *Terminal* on Linux.

To open Anaconda Prompt:

- **Windows:** Click **Start**, search for *Anaconda Prompt*, and click to open.
- **macOS:** Use **Cmd+Space** to open Spotlight Search and type “*Navigator*” to open the program.

To open Terminal:

- **Linux–CentOS:** Open **Applications > System Tools > terminal**.
- **Linux–Ubuntu:** Open the Dash by clicking the Ubuntu icon, then type “terminal”.

After opening Anaconda Prompt or the terminal, choose any of the following methods to verify:

- Enter `conda list`. If Anaconda is installed and working, this will display a list of installed packages and their versions.
- Enter the command `python`. This command runs the Python shell, also known as the REPL. If Anaconda is installed and working, the version information it displays when it starts up will include “Anaconda”. To exit the Python shell, enter the command `quit()`.

- Open Anaconda Navigator by typing the command `anaconda-navigator`. If Anaconda is installed properly, Anaconda Navigator will open.

[Back to checklist](#)

2.1 R

R is distributed by CRAN and is available for Windows, Linux, and MacOS. Installers are available [here](#). Make sure you download the installer that matches the operating system in your machine.

NOTE: R must be installed **before** you attempt to install RStudio or RTools.

2.2 RStudio

Once you install R in your local machine, you will need to install a program that allows you to interact with R. This program is known as an integrated development environment (IDE). There are several IDEs that allow you to interact with R, but RStudio is by far the most popular. We will use RStudio in our workshop, so we recommend you install it, especially if you are not an experienced R user. However, if you are more comfortable using a different IDE, you do not need to install RStudio.

You can download RStudio Desktop [here](#) for free. Once again, ensure you select the installer that matches the operating system in your machine.

[Back to checklist](#)

1.3 RTools (Windows users ONLY)

If you are running R in a Windows machine, you will also need to install RTools. This program will allow you to build some R packages. RTools installers can be downloaded from [here](#).


The version of RTools you need to install will depend on the version of R you have installed your computer. If you installed R while following these instructions, then you will need to download RTools 4.3. Otherwise, check the version of R you have installed in your machine to identify the correct RTools version. If you are unsure about the R version installed in your machine, you can simply type `version` in the RStudio console.

NOTE: Non-Windows users are **NOT** required to install RTools.

1.4 Checking and RStudio installation


Open RStudio in your machine. If you cannot find RStudio, follow the instructions below.

- **Windows:** Click **Start**, search for “*rstudio*”, and click to open.
- **macOS:** Use **Cmd+Space** to open Spotlight Search and type “*rstudio*” to open the program.
- **Linux–CentOS:** Open **Applications > System Tools > *rstudio***.
- **Linux–Ubuntu:** Open the Dash by clicking the Ubuntu icon, then type “*rstudio*”.

Once RStudio is opened, type  in the console (panel on the lower left) and press ENTER/RETURN. This should print output similar to the one shown below.

```
> version

platform      x86_64-w64-mingw32
arch           x86_64
os             mingw32
crt            ucrt
system        x86_64, mingw32
status
major          4
minor          2.3
year           2023
month          03
day            15
svn rev        83980
language       R
version.string R version 4.2.3 (2023-03-15 ucrt)
nickname       Shortstop Beagle
```

If you were able to open RStudio and the console printed something similar to what is shown above, then you have successfully installed  and RStudio in your machine.

[Back to checklist](#)

3. Accessing example notebooks

All notebooks created for this workshop are publicly available through our GitHub [repository](#).

Note that although the repository is publicly available in GitHub, you will need to have been granted access to the API services used in the notebooks. You don't need special access to connect directly with public datasets. If the training exercises require the use of API calls, you will receive access credentials in advance.

3.1 Making example notebooks accessible in your local machine

As mentioned above, all notebooks are in public repository in GitHub. You can use  to clone this repository to your local machine. If you do not have  installed in your machine, but would like to install

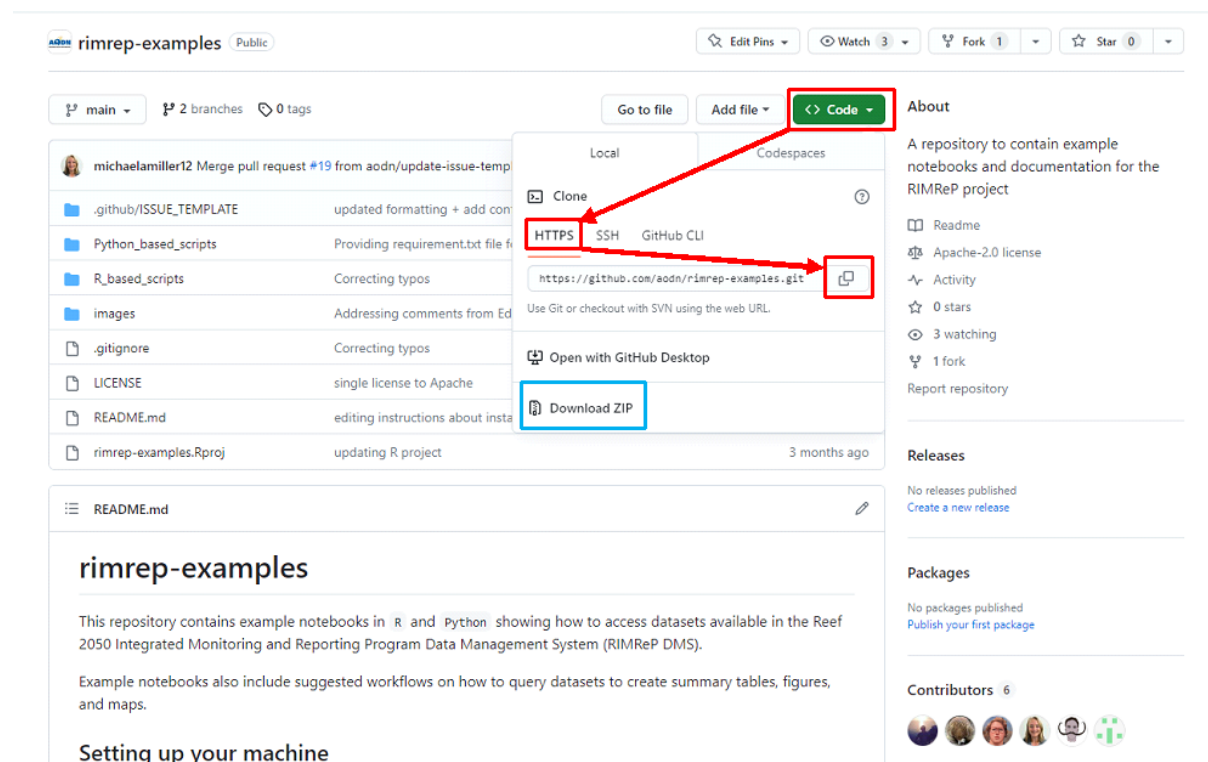
it, head over to their [website](#) for instructions and to download installers that match the operating system in your machine.

3.1.1 Cloning a repository

If you have `git` installed in your machine, follow these instructions.

1. Head over our [public repository](#) containing the example notebooks. You will see the page shown in the figure below.
2. Click on the green **Code** button, located to the left *About* section.
3. This will open a small tab, click on **HTTPS**, and you will see a URL ending in `.git`.
4. Click on the two boxes to copy the link ending in `.git`. Alternatively, you can highlight the link and copy it.
5. Open Git Bash in your machine.
6. Navigate to the folder where you would like to clone the repository. You can use the `cd` command to do this. For example, if you would like to navigate to your documents, you could type: `cd /c/Users/your_user_name/Documents/` and press ENTER/RETURN.
7. Finally type: `git clone https://github.com/aodn/rimrep-training.git` and press ENTER/RETURN. This will create an exact copy of our repository in your machine.

The red boxes in the figure below, show steps 2-4.



3.1.2 Downloading a repository

If you do not have `git` installed in your machine, you can simply download the repository as a compressed folder (`zip` format). Follow the instructions below:

1. Head over our [public repository](#) containing the example notebooks. You will see the page shown in the figure above.
2. Click on the green **Code** button, located to the left *About* section.
3. This will open a small tab showing **Download ZIP** as the last item in the list (see blue box in figure above). Click **Download ZIP** and a compressed (`zip`) folder will be downloaded.
4. When the download is completed, un-compress this folder, and place it in the folder of your choice.

[Back to checklist](#)

4. Installing Python packages

We will use the *requirements.txt* file available under the *Python_based_scripts* folder contained inside the repository you downloaded in the previous step. You can use this file to create a [conda environment](#) with all the packages used in the `Python` scripts.

To do this, you will need to open either the Anaconda Prompt (Windows) or the Terminal (MacOS, Linux). You will see a black screen with the path to the folder where you currently are.



```
Anaconda Prompt (miniconda3)
(base) C:\Users\username>
```

You will now need to change the directory to the folder where you saved the repository files you downloaded in the previous step. Assuming you cloned or unzipped the example folder in your Documents, the path should look something like this **C:/Users/username/Documents/rimrep-training**. Note that **username** will need to be replaced by the username shown in your computer. We will use this path when changing the directory.

Back in the *Anaconda Prompt* type: **cd C:/Users/username/Documents/rimrep-training/Python_based_scripts**. The command **cd** stands for changing directory. The path shown in the Anaconda should be the same that you included next to the **cd** command. You can check the contents of the folder by typing **dir**. Here, you should see the *requirements.txt* file.

If you can see the *requirements.txt* file, type the following line in the *Anaconda prompt* or *Terminal*:

```
conda env create -f requirements.txt -n rimrep
```



Note: If you are not in the directory where the `requirements.txt` file is located, the code above will not work. You will need to specify the path to the file. For example, if your terminal window is in the `rimrep-examples` folder, you will need to specify the full path to the file as follows: `conda env create -f Python_based_Scripts/requirements.txt -n rimrep`

You can check that everything was successfully installed by typing the following command:

```
conda activate rimrep
```

[Back to checklist](#)

5. Installing **R** packages

In our repository, we included an **R** script that automatically checks that all libraries used in this notebook are installed in your machine. If there are any missing libraries, then the script will install them for you.

This script can be found in the `R_based_scripts` folder of our repository and it is called

`Installing_R_libraries.R`.

To run this script, use RStudio to open the repository that you cloned/downloaded in the previous step.

You can do this by double clicking on the `rimrep-examples.Rproj` file included in the repository. This will open the repository as project in RStudio.

Once you have open this project in RStudio. Head over to the console and type the following lines:

```
source("R_based_scripts/Installing_R_libraries.R")
checking_libraries()
```

This will start the process of checking all packages are installed in your machine and install any missing packages.

[Back to checklist](#)