

ROS installation with Robostack on macOS

<https://robostack.github.io/>

V0.1

1st step: preparation of the tools

1/ Install iTerm2, a Terminal application in which you will be able to execute a Shell, see <https://iterm2.com/>

2/ Install Homebrew, cf. <https://brew.sh/>

- Open iTerm2
- In iTerm2, copy and past the command:

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

Once installed, we will use the brew command in a terminal to install various software.

3/ Install the bash shell with brew with

```
brew install bash bash-completion
```

4/ Configure iTerm2 to use the bash shell by creating a new profile:

- Open iTerm2 preferences
- Click on Profiles, and add a new profile by clicking on the + on the bottom.
- Give it a name (e.g. Bash), and in the general tab, in the Command section, select "Custom Shell" in the dropdown list, and type /opt/homebrew/bin/bash in the box on the right

Close the Preferences window. You can now select the Bash profile in the Profiles menu of iTerm2. Clicking on it opens a new tab, or a new window, and launches the bash shell.

3/ You need to have a base conda installation. Robostack documentation indicates: "Please do *not* use the Anaconda installer, but rather start with [miniforge](#) / [mambaforge](#), which are much more "minimal" installers (we recommend mambaforge)."

3.1/ If you already installed Anaconda, please uninstall it, see <https://docs.anaconda.com/anaconda/install/uninstall/>

3.2/ Then, install mambaforge, an alternative and minimalist conda installation. This is made very easy thanks to brew. Simply type in an iTerm2 window (**from a bash shell**):

```
brew install mambaforge
```

3.3/ Once installed, type in the shell

```
conda init bash
```

You can now use conda to manage python environments from the command line, and install the packages you might need (numpy, etc.) in a new environment. Please avoid to use the base environment to install packages.

2nd step: installation of ROS

In the bash shell:

4/ Create a new environment dedicated to ROS named `ros_env` :

```
mamba create -n ros_env python=3.9
```

and activate it

```
conda activate ros_env
```

From now on, all your ROS environment will be installed in the conda `ros_env` environment.

Configure the environment:

```
conda config --env --add channels robostack
```

```
conda config --env --add channels robostack-experimental
```

5/ Install ROS and other tools:

```
mamba install ros-noetic-desktop
```

```
mamba install compilers cmake pkg-config make ninja colcon-common-extensions
```

```
mamba install catkin_tools
```

```
mamba install ros-noetic-turtlebot3 ros-noetic-turtlebot3-simulations
```

6/ Reload the environment to activate required scripts before running anything

```
conda deactivate
```

```
conda activate ros_env
```

and finally

```
mamba install rosdep
```

```
rosdep init
```

```
rosdep update
```

7/ Download the two files `.bash_prompt` and `.bash_profile` in your home directory (they can be found on Moodle).

3rd step: configuration of ROS

8/ Create your catkin workspace from a bash shell:

```
mkdir -p ~/catkin_ws/src
```

```
cd ~/catkin_ws/src
```

```
catkin_init_workspace
```

That's it, you should be ready to work with ROS!