Environment Management on JupyterHub

On JupyterHub, your base environment is independent from the other users. Hence you can install packages and libraries based on your own needs. However, you may encounter some issues if you use `conda` for installation. Here are a few points to note.

- For the same library, we recommend using `pip` instead of `conda` for installation. `conda` may lead to some permission-related issues, while `pip` can bypass this problem and complete installation efficiently.
- In some cases, you may have specific requirements for the version of Python libraries and cannot easily install them using `pip`. As a solution, you can create a virtual environment that is independent of the base environment. Because it is created under your working directory, you can use `conda` to install packages and libraries. It would be a good practice to create a separate virtual environment for each research project.

  You can create and use a virtual environment by following these steps:

  1. Open the terminal in JupyterHub
  2. Run the following command to create a new virtual environment:

  ```
  conda create -n your_virtual_environment python=3.7
  ```

  Replace "your_virtual_environment" with the desired name, and “3.7” with the desired Python version for your virtual environment.

  3. Activate the virtual environment with `conda`

  ```
  conda activate your_virtual_environment
  ```

4. Once the virtual environment is activated, you can install libraries or packages specific to your project using `pip` or `conda` commands.
5. When you are done working with the virtual environment, you can deactivate it using the following command:

```
conda deactivate
```

- Please keep in mind that `pip` and `conda` cannot be called when you are using `ssh`, because in this case the base environment is shared by all users.