CENTRE FOR
DEVELOPMENT OF
ADVANCED COMPUTING

OCTOBER 2024

MINICONDA

COMPLETE GUIDE

PRESENTED BY

KISHOR Y D

HPC TECHNOLOGIES GROUP

C-DAC PUNE

AGENDA

Introduction	
Advantages and Disadvantages	
Miniconda Installation Guide	
Managing Conda	
Managing Environments	
Managing Channels	
Managing Packages	

Introduction

1 : An open source package and environment management system that runs on Windows, Mac OS and Linux.

2 : Miniconda is a minimalistic distribution of the Conda package manager, specifically designed for users who need a lightweight, flexible tool to manage software environments and dependencies.

3 : Conda, the core of Miniconda, is a powerful cross-platform package and environment manager.

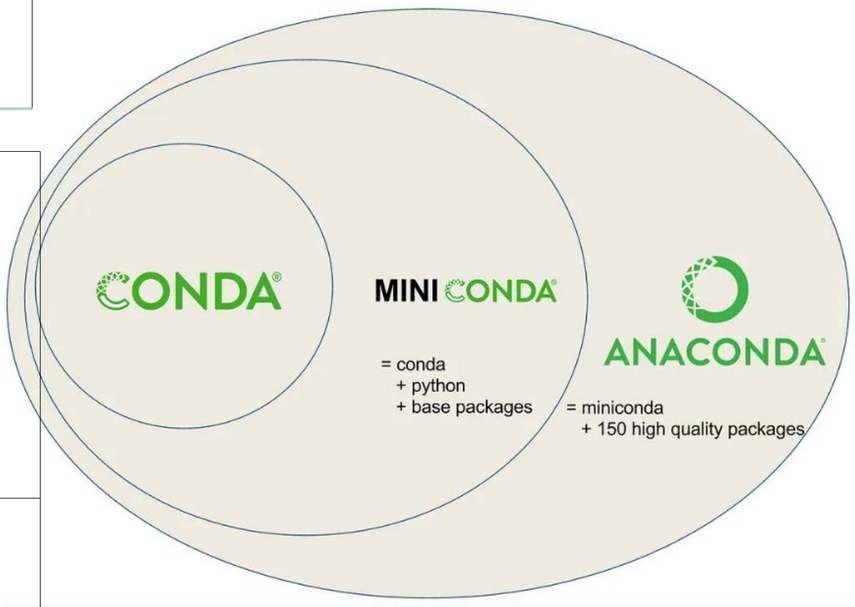
4 : Allowing users to build custom environments by installing only the packages they need.

5 : Supports the installation of GPU-optimized versions of libraries like TensorFlow, Pytorch and others with CUDA and cuDNN integration.

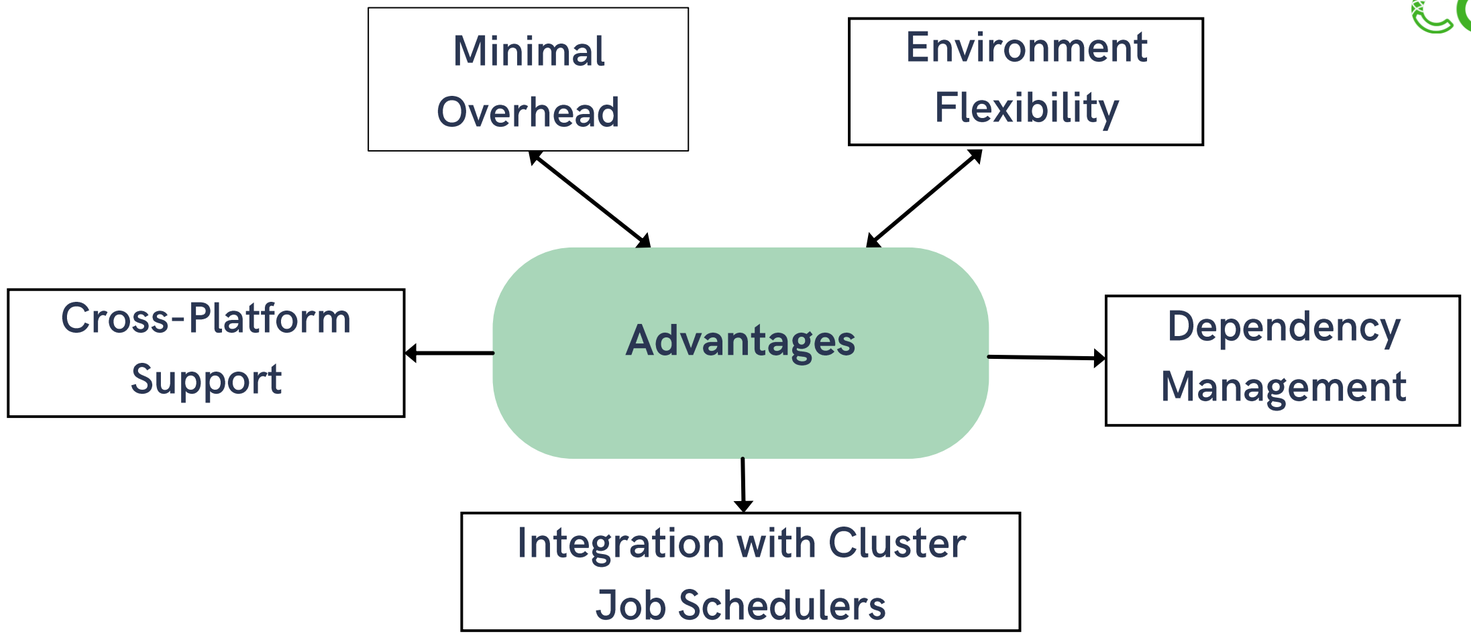
6 : System requirements: 400MB disk space and OS

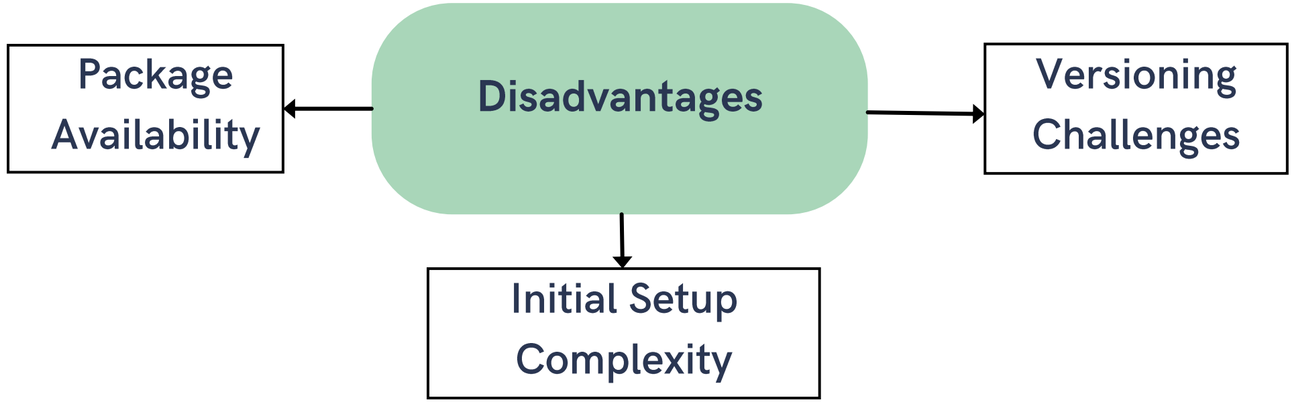
Installing Conda

1. **Miniconda (Minimal Installer)**
2. **Anaconda Distribution (Full Package)**
3. **Miniforge (Minimal Installer)**



Advantages and Disadvantages



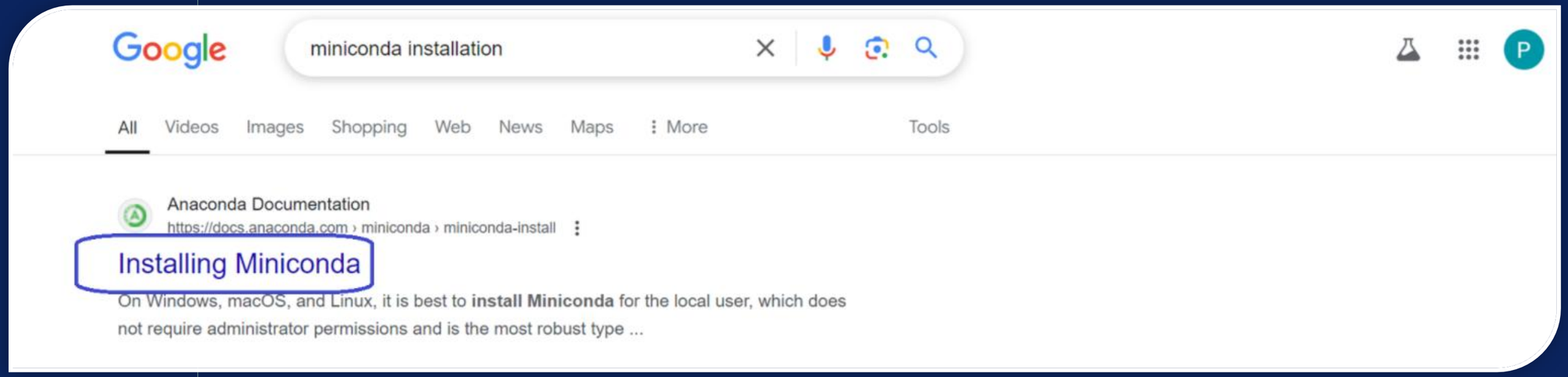


INSTALLATION GUIDE

WEB SEARCH FOR INSTALLATION LINK

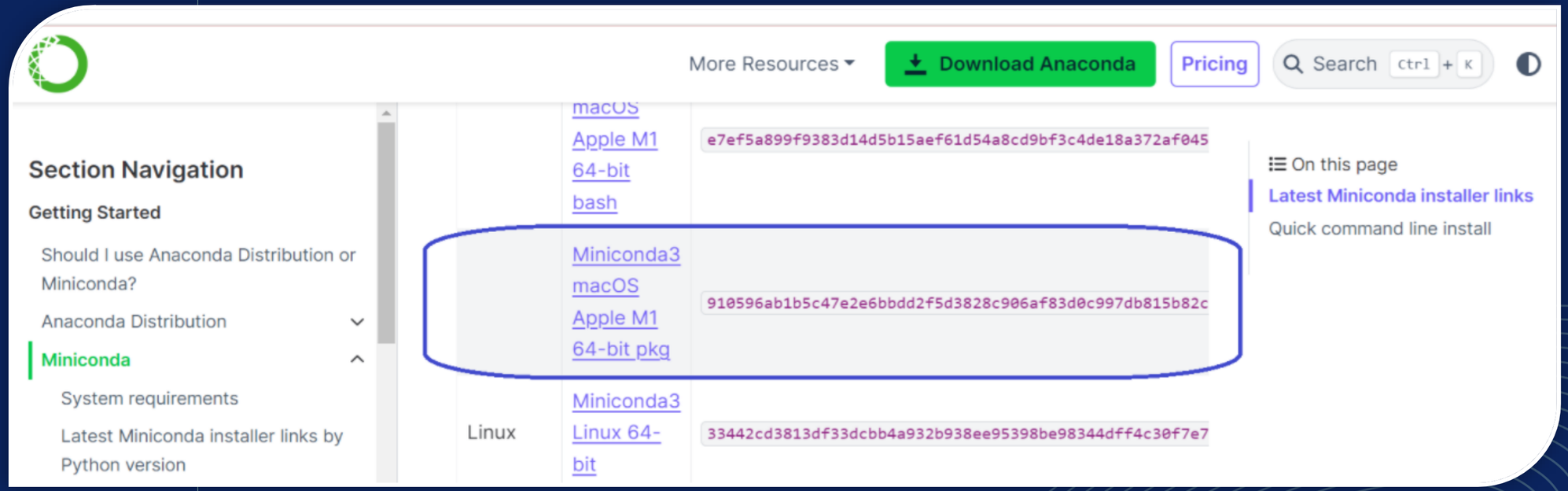
STEP 1:

Search for Miniconda installation



STEP 2:

Scroll down to copy Miniconda distribution link



INSTALLATION GUIDE

OPEN CLUSTER TERMINAL

STEP 3:

Create a directory structure to store executable : $ mkdir <directory_name>

```
2. RUDRA luac -> samritm
[samritm@login09 ~]$ mkdir Miniconda
[samritm@login09 ~]$ ls
Miniconda  test
[samritm@login09 ~]$
```

STEP 4:

Download the .sh using link from STEP 2 : $ wget <web-download_link>

```
[samritm@login09 ~]$ wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
--2024-10-08 18:53:16--  https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
Resolving repo.anaconda.com (repo.anaconda.com)... 104.16.32.241, 104.16.191.158, 2606:4700::6810:20f1, ...
Connecting to repo.anaconda.com (repo.anaconda.com)|104.16.32.241|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 148981743 (142M) [application/octet-stream]
Saving to: 'Miniconda3-latest-Linux-x86_64.sh'

Miniconda3-latest-Linux-x86_64. 100%[=====================================================>] 142.08M  31.1MB/s    in 4.3s

2024-10-08 18:53:21 (33.2 MB/s) - 'Miniconda3-latest-Linux-x86_64.sh' saved [148981743/148981743]
```

STEP 5:

Extract the executable into Miniconda Directory : $ sh <bash_script> -b –p <directory_name -u>

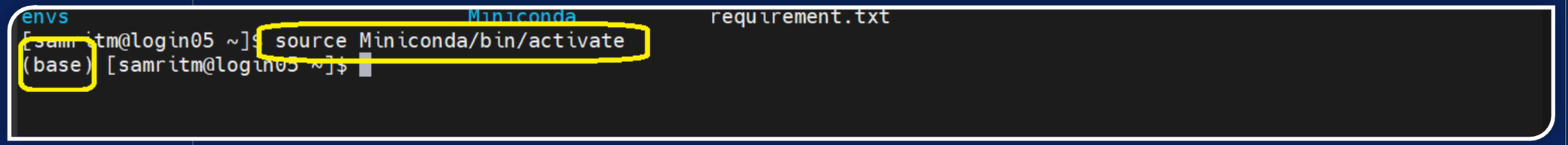
```
[samritm@login09 ~]$ sh Miniconda3-latest-Linux-x86_64.sh -b -p Miniconda/ -u
PREFIX=/home/samritm/Miniconda
Unpacking payload ...

Installing base environment...

Preparing transaction: ...working... done
Executing transaction: ...working... done
installation finished.
```

INSTALLATION GUIDE (Optional)

OPEN CLUSTER TERMINAL

STEP 6:

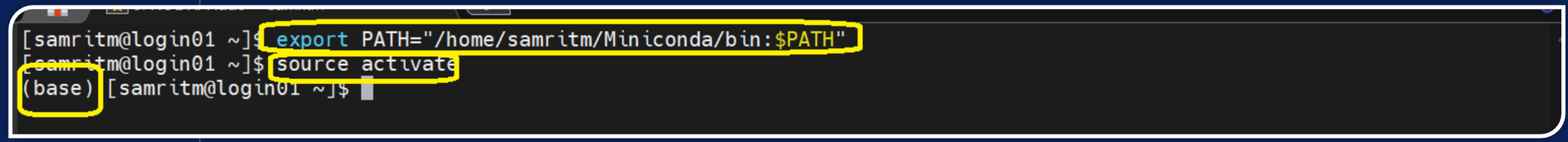
Use source to apply changes to the current session to use Miniconda : $ source <directory_name>/bin/activate



STEP 7:

Use export to create or modify environment variables : export PATH="<Absolute_path_directory>/bin:$PATH"

Hint: use either STEP 6 or STEP 7



INSTALLATION GUIDE (Optional)

OPEN CLUSTER TERMINAL

STEP 8:

Run conda init to configure the shell : $ <directory_name>/bin/conda init

```
[samritm@login06 ~]$ Miniconda/bin/conda init
no change     /home/samritm/Miniconda/condabin/conda
no change     /home/samritm/Miniconda/bin/conda
no change     /home/samritm/Miniconda/bin/conda-env
no change     /home/samritm/Miniconda/bin/activate
no change     /home/samritm/Miniconda/bin/deactivate
no change     /home/samritm/Miniconda/etc/profile.d/conda.sh
no change     /home/samritm/Miniconda/etc/fish/conf.d/conda.fish
no change     /home/samritm/Miniconda/shell/condabin/Conda.psm1
no change     /home/samritm/Miniconda/shell/condabin/conda-hook.ps1
no change     /home/samritm/Miniconda/lib/python3.12/site-packages/xontrib/conda.xsh
no change     /home/samritm/Miniconda/etc/profile.d/conda.csh
modified      /home/samritm/.bashrc

==> For changes to take effect, close and re-open your current shell. <==
```

STEP 9:

Restart the shell or reload the configuration : $ source ~/.bashrc

```
(base) [samritm@login06 ~]$ source ~/.bashrc
(base) [samritm@login06 ~]$ cat ~/.bashrc
# .bashrc

# >>> conda initialize >>>
# !! Contents within this block are managed by 'conda init' !!
__conda_setup="$('/home/samritm/Miniconda/bin/conda' 'shell.bash' 'hook' 2> /dev/null)"
if [ $? -eq 0 ]; then
    eval "$__conda_setup"
else
    if [ -f "/home/samritm/Miniconda/etc/profile.d/conda.sh" ]; then
        . "/home/samritm/Miniconda/etc/profile.d/conda.sh"
    else
        export PATH="/home/samritm/Miniconda/bin:$PATH"
    fi
fi
unset __conda_setup
# <<< conda initialize <<<
```

Hint : use *conda init –reverse –all* to undo changes

MANAGING CONDA

FOLLOW THE INSTALLATION GUIDE

Verify that conda is installed : $ conda --version

```
anaconda  anaconda1  Anaconda3-2024.06-1-Linux-x86_64.sh  Miniconda  Miniconda3-latest-Linux-x86_64.sh  test
(base) [samritm@login06 ~]$ conda --version
conda 24.7.1
(base) [samritm@login06 ~]$
```

Updating conda to the current version : $ conda update –n base conda

```
(base) [samritm@login06 ~]$ conda update -n base conda
Channels:
 - defaults
Platform: linux-64
Collecting package metadata (repodata.json): done
Solving environment: done

# All requested packages already installed.
```

If newer version is available, type y to update :

```
The following packages will be UPDATED:

  ca-certificates                    2024.7.2-h06a4308_0 --> 2024.9.24-h06a4308_0
  certifi                        2024.7.4-py312h06a4308_0 --> 2024.8.30-py312h06a4308_0
  conda                            24.7.1-py312h06a4308_0 --> 24.9.1-py312h06a4308_0
  openssl                             3.0.14-h5eee18b_0 --> 3.0.15-h5eee18b_0

Proceed ([y]/n)? y

Downloading and Extracting Packages:

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
```

Recommended: Always create new environment before installing any packages.

MANAGING ENVIRONMENTS

CREATING ENVIRONMENTS WITH COMMANDS

To create an environment : $ conda create –name <my-env>

```
(base) [samritm@login06 ~]$ conda create --name lab_conda
Channels:
 - defaults
Platform: linux-64
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /home/samritm/Miniconda/envs/lab_conda
```

To create an environment with a specific version of Python: $ conda create –name <my-env> python=<version>

```
(base) [samritm@login06 ~]$ conda create --name lab_conda_2 python=3.9
Channels:
 - defaults
Platform: linux-64
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /home/samritm/Miniconda/envs/lab_conda_2

added / updated specs:
  - python=3.9
```

To create an environment with a specific version of Python and multiple packages: $ conda create –name python=<version> numpy=<version>

```
(base) [samritm@login06 ~]$ conda create --name lab_conda_3 python=3.8 numpy=1.14.6
Channels:
 - defaults
Platform: linux-64
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /home/samritm/Miniconda/envs/lab_conda_3

added / updated specs:
  - numpy=1.14.6
  - python=3.8
```

MANAGING ENVIRONMENTS

CREATING ENVIRONMENTS WITH COMMANDS

For detailed information : $ conda create –help

```
(base) [samritm@login06 ~]$ conda create --help
usage: conda create [-h] [--clone ENV] [-n ENVIRONMENT | -p PATH] [-c CHANNEL] [--use-local] [--override-channels]
                    [--repodata-fn REPODATA_FNS] [--experimental {jlap,lock}] [--no-lock] [--repodata-use-zst | --no-repodata-use-zst]
                    [--strict-channel-priority] [--no-channel-priority] [--no-deps | --only-deps] [--no-pin] [--copy] [--no-shortcuts]
                    [--shortcuts-only SHORTCUTS_ONLY] [-C] [-k] [--offline] [--json] [-v] [-q] [-d] [-y] [--download-only]
                    [--show-channel-urls] [--file FILE] [--no-default-packages] [--subdir SUBDIR] [--solver {classic,libmamba}] [-m]
                    [--dev]
                    [package_spec ...]
```

```
Additional arguments:
  package_spec          List of packages to install or update in the conda environment.

options:
  -h, --help            Show this help message and exit.
  --clone ENV           Create a new environment as a copy of an existing local environment.
  --file FILE           Read package versions from the given file. Repeated file specifications can be passed (e.g. --file=file1
                        --file=file2).
  -m, --mkdir           `--mkdir` is deprecated and will be removed in 25.3. Redundant argument.
  --dev                 Use `sys.executable -m conda` in wrapper scripts instead of CONDA_EXE. This is mainly for use during tests where
                        we test new conda sources against old Python versions.

Target Environment Specification:
  -n ENVIRONMENT, --name ENVIRONMENT
                        Name of environment.
  -p PATH, --prefix PATH
                        Full path to environment location (i.e. prefix).

Channel Customization:
  -c CHANNEL, --channel CHANNEL
                        Additional channel to search for packages. These are URLs searched in the order they are given (including local
                        directories using the 'file://' syntax or simply a path like '/home/conda/mychan' or '../mychan'). Then, the
                        defaults or channels from .condarc are searched (unless --override-channels is given). You can use 'defaults' to
                        get the default packages for conda. You can also use any name and the .condarc channel_alias value will be
                        prepended. The default channel_alias is https://conda.anaconda.org/.
  --use-local           Use locally built packages. Identical to '-c local'.
  --override-channels   Do not search default or .condarc channels. Requires --channel.
```

MANAGING ENVIRONMENTS

CREATING ENVIRONMENTS WITH COMMANDS

For detailed information : $ conda create –help

```
Solver Mode Modifiers:
  --strict-channel-priority
                        Packages in lower priority channels are not considered if a package with the same name appears in a higher
                        priority channel.
  --no-channel-priority
                        Package version takes precedence over channel priority. Overrides the value given by `conda config --show
                        channel_priority`.
  --no-deps             Do not install, update, remove, or change dependencies. This WILL lead to broken environments and inconsistent
                        behavior. Use at your own risk.
  --only-deps           Only install dependencies.
  --no-pin              Ignore pinned file.
  --no-default-packages
                        Ignore create_default_packages in the .condarc file.
```

```
Networking Options:
  -C, --use-index-cache
                        Use cache of channel index files, even if it has expired. This is useful if you don't want conda to check
                        whether a new version of the repodata file exists, which will save bandwidth.
  -k, --insecure        Allow conda to perform "insecure" SSL connections and transfers. Equivalent to setting 'ssl_verify' to 'false'.
  --offline             Offline mode. Don't connect to the Internet.

Output, Prompt, and Flow Control Options:
  --json                Report all output as json. Suitable for using conda programmatically.
  -v, --verbose         Can be used multiple times. Once for detailed output, twice for INFO logging, thrice for DEBUG logging, four
                        times for TRACE logging.
  -q, --quiet           Do not display progress bar.
  -d, --dry-run         Only display what would have been done.
  -y, --yes             Sets any confirmation values to 'yes' automatically. Users will not be asked to confirm any adding, deleting,
                        backups, etc.
  --download-only       Solve an environment and ensure package caches are populated, but exit prior to unlinking and linking packages
                        into the prefix.
  --show-channel-urls   Show channel urls. Overrides the value given by `conda config --show show_channel_urls`.
```

MANAGING ENVIRONMENTS

CREATING ENVIRONMENTS WITH COMMANDS

Exporting existing environment packages to local file : $ conda env export –name <env_name> > <environment>.yml

```
(base) [samritm@login06 ~]$ conda env export --name lab_conda_3 > lab_3_packages.yml
(base) [samritm@login06 ~]$ ls
anaconda    Anaconda3-2023.09-0-Linux-x86_64.sh  Miniconda                          test
anaconda1   lab_3_packages.yml                   Miniconda3-latest-Linux-x86_64.sh
```

Creating new environment from .yml file : $ conda env create –file <environment>.yml –name <env_name>

```
(base) [samritm@login06 ~]$ conda env create --file lab_3_packages.yml --name lab_conda_4
/home/samritm/Miniconda/lib/python3.12/argparse.py:2005: FutureWarning: `remote_definition` is deprecated and will be removed
 in 25.9. Use `conda env create --file=URL` instead.
  action(self, namespace, argument_values, option_string)
Channels:
 - defaults
Platform: linux-64
Collecting package metadata (repodata.json): done
Solving environment: done
```

Creating new environment from requirement file : $ conda env create –file <requirement>.txt –name <env_name>

```
(base) [samritm@login06 ~]$ conda env create --file requirement.txt --name lab_conda_5
/home/samritm/Miniconda/lib/python3.12/argparse.py:2005: FutureWarning: `remote_definition` is deprecated and will be removed
 in 25.9. Use `conda env create --file=URL` instead.
  action(self, namespace, argument_values, option_string)
Channels:
 - conda-forge
 - defaults
Platform: linux-64
Collecting package metadata (repodata.json): done
Solving environment: done
```

MANAGING ENVIRONMENTS

CREATING ENVIRONMENTS WITH COMMANDS

Cloning an existing environment : $ conda create –name <new_env_name> –clone <old_env_name>

```
(base) [samritm@login06 lab_conda_5]$ conda create --name lab_conda_6 --clone lab_conda_5
Source:      /home/samritm/Miniconda/envs/lab_conda_5
Destination: /home/samritm/Miniconda/envs/lab_conda_6
Packages: 162
Files: 1

Downloading and Extracting Packages:
```

Creating new environment to particular directory : $ conda env create –prefix <path> <packages>

```
(base) [samritm@login06 lab_conda_5]$ conda create --prefix /home/samritm/envs jupyterlab=3.2 matplotlib=3.5 numpy=1.21
Channels:
 - conda-forge
 - defaults
Platform: linux-64
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /home/samritm/envs

  added / updated specs:
    - jupyterlab=3.2
    - matplotlib=3.5
    - numpy=1.21
```

MANAGING ENVIRONMENTS

VIEWING LIST OF ENVIRONMENTS

To see list of all environments: $ conda info –envs / conda env list

```
(base) [samritm@login06 lab_conda_5]$ conda env list
# conda environments:
#
farseq                   /home/samritm/.conda/envs/farseq
base                  *  /home/samritm/Miniconda
lab_conda                /home/samritm/Miniconda/envs/lab_conda
lab_conda_2              /home/samritm/Miniconda/envs/lab_conda_2
lab_conda_3              /home/samritm/Miniconda/envs/lab_conda_3
lab_conda_4              /home/samritm/Miniconda/envs/lab_conda_4
lab_conda_5              /home/samritm/Miniconda/envs/lab_conda_5
lab_conda_6              /home/samritm/Miniconda/envs/lab_conda_6
```

Viewing list of packages in an environment : $ conda list –name <env_name>

Or Environment is activated, in the terminal run : $ conda list

```
(base) [samritm@login06 lab_conda_5]$ conda list --name lab_conda_6
# packages in environment at /home/samritm/Miniconda/envs/lab_conda_6:
#
# Name                    Version                   Build  Channel
_libgcc_mutex             0.1                 conda_forge    conda-forge
_openmp_mutex             4.5                       2_gnu    conda-forge
alsa-lib                  1.2.12               h4ab18f5_0    conda-forge
attr                      2.5.1                h166bdaf_1    conda-forge
brotli                    1.1.0                hb9d3cd8_2    conda-forge
brotli-bin                1.1.0                hb9d3cd8_2    conda-forge
bzip2                     1.0.8                h4bc722e_7    conda-forge
ca-certificates           2024.8.30            hbcca054_0    conda-forge
cairo                     1.18.0               hebfffa5_3    conda-forge
certifi                   2024.8.30          pyhd8ed1ab_0    conda-forge
contourpy                 1.2.1            py39h7633fee_0    conda-forge
cycler                    0.12.1             pyhd8ed1ab_0    conda-forge
dbus                      1.13.6               h5008d03_3    conda-forge
expat                     2.6.3                h5888daf_0    conda-forge
```

MANAGING ENVIRONMENTS

RESTORING AND RENAME AN ENVIRONMENT

To rollback to previous revision : $ conda list --revisions

```
(base) [samritm@login06 lab_conda_5]$ conda list --revisions
2024-08-16 20:09:27  (rev 0)
    +_libgcc_mutex-0.1
    +_openmp_mutex-5.1
    +anaconda-anon-usage-0.4.4
    +archspec-0.2.3
    +boltons-23.0.0
    +brotli-python-1.0.9
    +bzip2-1.0.8
    +c-ares-1.19.1
    +ca-certificates-2024.7.2
```

To restore environment installation of packages to previous revision: $ conda install –rev <REVNUM>

```
(base) [samritm@login06 lab_conda_5]$ conda install --rev 01
Collecting package metadata (current_repodata.json): done
Reverting to revision 1: done

# All requested packages already installed.
```

To rename existing environment: $ conda rename –name <old_env_name> <new_env_name>

```
(base) [samritm@login02 ~]$ conda rename --name lab_conda_6 lab_conda_7
Source:      /home/samritm/Miniconda/envs/lab_conda_6
Destination: /home/samritm/Miniconda/envs/lab_conda_7
Packages: 38
Files: 1
```

MANAGING ENVIRONMENTS

ACTIVATE / DEACTIVATE ENVIRONMENTS

To activate particular environment : $ conda activate <env_name>

```
(base) [samritm@login06 lab_conda_5]$ conda activate lab_conda_6
(lab_conda_6) [samritm@login06 lab_conda_5]$
```

To deactivate the existing environment: $ conda deactivate

```
(lab_conda_6) [samritm@login06 lab_conda_5]$ conda deactivate
(base) [samritm@login06 lab_conda_5]$
```

For detailed information : $ conda remove --help

```
(base) [samritm@login06 lab_conda_5]$ conda remove --help
usage: conda remove [-h] [-n ENVIRONMENT | -p PATH] [-c CHANNEL] [--use-local] [--override-channels]
                    [--repodata-fn REPODATA_FNS] [--experimental {jlap,lock}] [--no-lock]
                    [--repodata-use-zst | --no-repodata-use-zst] [--features] [--force-remove] [--no-pin]
                    [--solver {classic,libmamba}] [-C] [-k] [--offline] [--json] [-v] [-q] [-d] [-y] [--all] [--keep-env]
                    [--dev]
                    [package_name ...]

Options:
  -h, --help            Show this help message and exit.
  --all                 Remove all packages, i.e., the entire environment.
  --keep-env            Used with `--all`, delete all packages but keep the environment.
  --dev                 Use `sys.executable -m conda` in wrapper scripts instead of CONDA_EXE. This is mainly for use
                        during tests where we test new conda sources against old Python versions.

Target Environment Specification:
  -n ENVIRONMENT, --name ENVIRONMENT
                        Name of environment.
  -p PATH, --prefix PATH
                        Full path to environment location (i.e. prefix).
```

MANAGING ENVIRONMENTS

REMOVE PACKAGES / ENVIRONMENT

To remove particular package in environment : $ conda remove <package_name>

```
executing transaction: done
(lab_conda_6) [samritm@login10 ~]$ conda remove numpy
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /home/samritm/Miniconda/envs/lab_conda_6

  removed specs:
    - numpy
```

To remove a package from a environment ‘lab_conda_6’: $ conda remove –name <env_name> <package_name>

```
(base) [samritm@login10 ~]$ conda remove --name lab_conda_6 numpy
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /home/samritm/Miniconda/envs/lab_conda_6

  removed specs:
    - numpy
```

To remove environment ‘lab_conda_6’ from list of environments: $ conda env remove –name <env_name>

```
(base) [samritm@login02 ~]$ conda env remove --name lab_conda_6

Remove all packages in environment /home/samritm/Miniconda/envs/lab_conda_6:
```

MANAGING ENVIRONMENTS

REMOVE PACKAGES

To remove all packages from environment ‘lab_conda_6’ and the environment itself: $ conda remove –name <env_name> --all

```
(base) [samritm@login10 ~]$ conda remove --name lab_conda_6 --all

Remove all packages in environment /home/samritm/Miniconda/envs/lab_conda_6:


## Package Plan ##

  environment location: /home/samritm/Miniconda/envs/lab_conda_6


The following packages will be REMOVED:
```

To remove all packages from the environment ‘lab_conda_6’ but retain the environment: $ conda remove –name <env_name> –all –keep-env

```
(base) [samritm@login05 ~]$ conda remove --name lab_conda_6 --all --keep-env
Remove all packages in environment /home/samritm/Miniconda/envs/lab_conda_6:


## Package Plan ##

  environment location: /home/samritm/Miniconda/envs/lab_conda_6
```

To remove all ununsed packages from environment: $ conda clean --all

```
(lab_conda_6) [samritm@login02 ~]$ conda clean --all
Will remove 320 (1.47 GB) tarball(s).
Proceed ([y]/n)? y
```

MANAGING CHANNELS

ADD CHANNELS

To list all channels currently configured for conda environment: $ conda config –show channels

1. Defaults
2. Conda-forge
3. Anaconda
4. Nvidia
5. Pytorch

```
(base) [samritm@login05 ~]$ conda config --show channels
channels:
  - defaults
  - conda-forge
(base) [samritm@login05 ~]$
```

To add a channel to the configurations: $ conda config –add channels <channel-name>

```
(base) [samritm@login05 ~]$ conda config --add channels nvidia
(base) [samritm@login05 ~]$ conda config --show channels
channels:
  - nvidia
  - defaults
  - conda-forge
(base) [samritm@login05 ~]$
```

MANAGING CHANNELS

TO REMOVE DEFAULT AND SET SPECIFIC CHANNEL

To remove a specific channel from configuration: $ conda config –remove channels <channel_name>

```
(base) [samritm@login05 ~]$ conda config --remove channels nvidia
(base) [samritm@login05 ~]$ conda config --show channels
channels:
  - defaults
  - conda-forge
```

When installing a package, can specify the channel to install without permanently adding channel : $ conda install <package_name> -c <channel_name>

```
(base) [samritm@login05 ~]$ conda install xgboost -c conda-forge
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /home/samritm/Miniconda

  added / updated specs:
    - xgboost
```

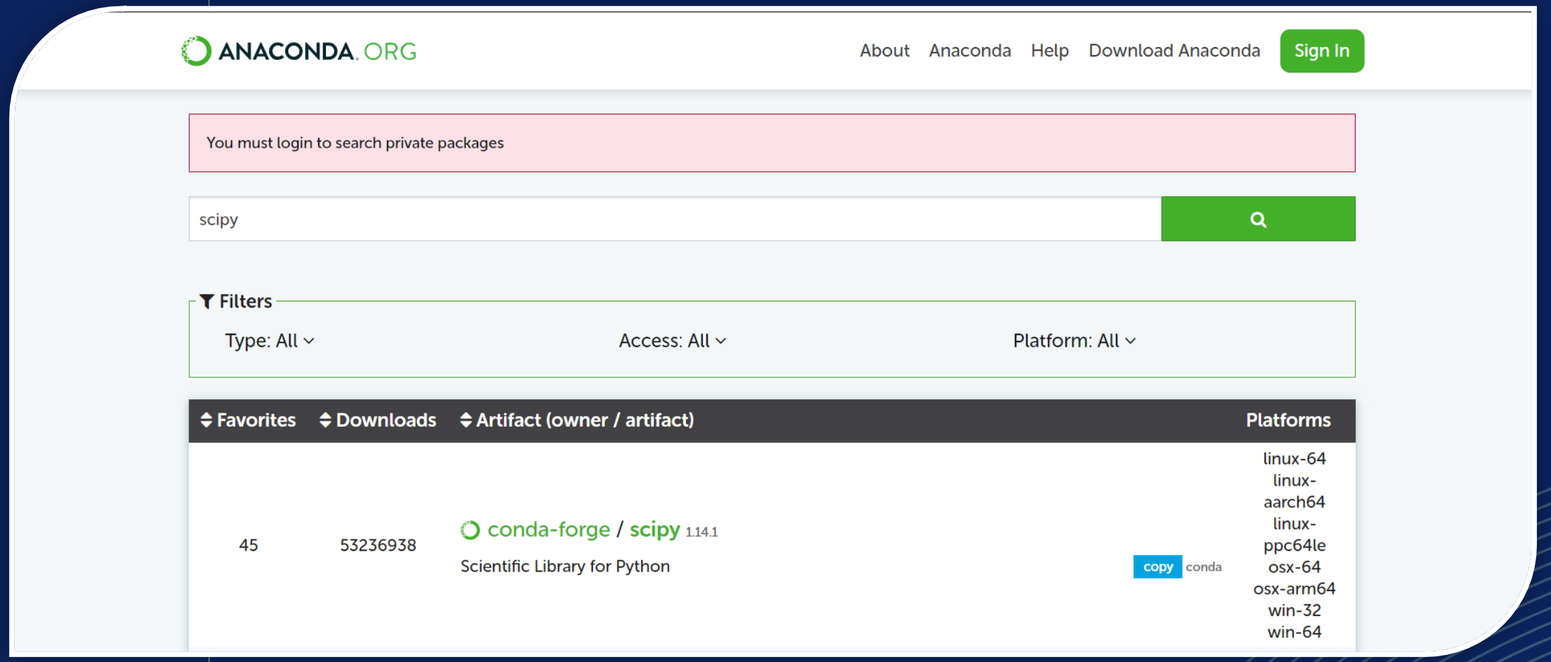
MANAGING PACKAGES

TO SEARCH FOR PACKAGES

To see if a specific packages are for available for installation : $ conda search <package_name>

```
(lab_conda_6) [samritm@login02 ~]$ conda search scipy
Loading channels: done
# Name                       Version           Build  Channel
scipy                         0.17.1 np110py27_blas_openblas_200  conda-forge
scipy                         0.17.1 np110py27_blas_openblas_201  conda-forge
scipy                         0.17.1 np110py27_blas_openblas_202  conda-forge
scipy                         0.17.1 np110py27_blas_openblas_203  conda-forge
scipy                         0.17.1 np110py27_blas_openblas_204  conda-forge
scipy                         0.17.1 np110py27_blas_openblas_205  conda-forge
scipy                         0.17.1 np110py34_blas_openblas_200  conda-forge
```

Alternatively can search on official website of anaconda for packages : https://anaconda.org/



MANAGING PACKAGES

TO SEARCH FOR NON-CONDA PACKAGES AND INSTALLATION

To install particular packages on environment : $ pip install <package_name>

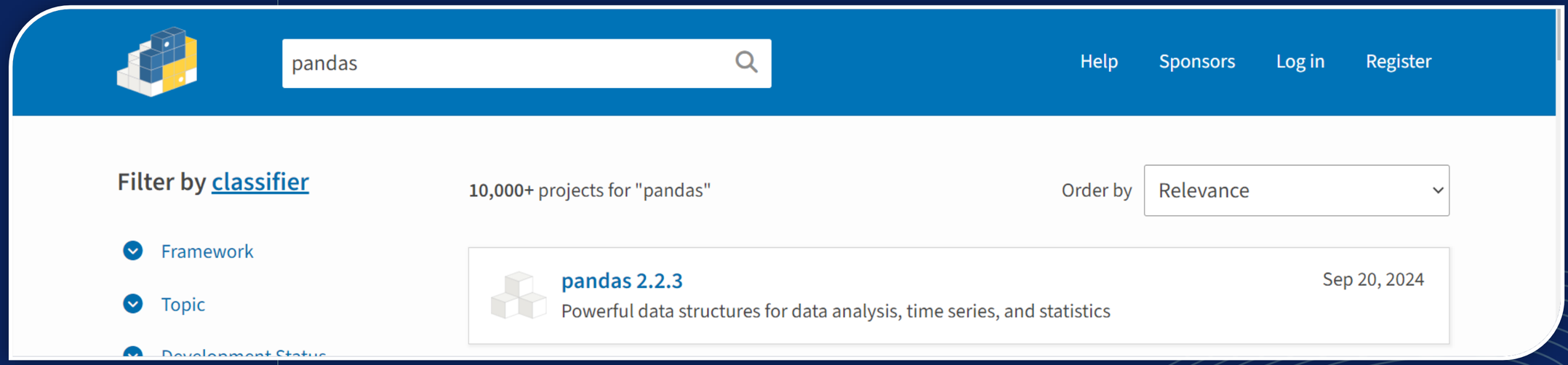
If a package is not available on conda or conda forge. We can install packages with another package manager like pip.

```
(lab_conda_6) [samritm@login02 ~]$ pip install pandas
Collecting pandas
  Downloading pandas-2.2.3-cp39-cp39-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (89 kB)
Requirement already satisfied: numpy>=1.22.4 in ./Miniconda/envs/lab_conda_6/lib/python3.9/site-packages (from pandas) (1.26.4
)
Collecting python-dateutil>=2.8.2 (from pandas)
  Using cached python_dateutil-2.9.0.post0-py2.py3-none-any.whl.metadata (8.4 kB)
Collecting pytz>=2020.1 (from pandas)
  Downloading pytz-2024.2-py2.py3-none-any.whl.metadata (22 kB)
Collecting tzdata>=2022.7 (from pandas)
  Downloading tzdata-2024.2-py2.py3-none-any.whl.metadata (1.4 kB)
Requirement already satisfied: six>=1.5 in ./Miniconda/envs/lab_conda_6/lib/python3.9/site-packages (from python-dateutil>=2.
```

Search on official website of python package Index (PyPI) for packages : https://pypi.org/search

Installation of pip:

$ conda install pip



MANAGING PACKAGES

LIST OUT NON-CONDA PACKAGES AND UN-INSTALLATION

Search on official website of python package Index (PyPI) for packages : https://pypi.org/search

To list all the packages installed through pip manager : pip list

```
(lab_conda_6) [samritm@login02 ~]$ pip list | grep pandas
pandas                        2.2.3
(lab_conda_6) [samritm@login02 ~]$
```

To uninstall packages which are installed through pip manager : pip uninstall <package_name>

```
(lab_conda_6) [samritm@login02 ~]$ pip uninstall pandas
Found existing installation: pandas 2.2.3
Uninstalling pandas-2.2.3:
  Would remove:
    /home/samritm/Miniconda/envs/lab_conda_6/lib/python3.9/site-packages/pandas-2.2.3.dist-info/*
    /home/samritm/Miniconda/envs/lab_conda_6/lib/python3.9/site-packages/pandas/*
Proceed (Y/n)? y
  Successfully uninstalled pandas-2.2.3
```

Cheat sheet : *https://docs.conda.io/projects/conda/en/latest/_downloads/e0795803d81d6d87b74abed339025237/conda-24.4.0.pdf*

THANK YOU