[Conda] [cheatsheet]

1. Installation and Setup

- **Install Miniconda or Anaconda**: Anaconda documentation

2. Managing Conda

- **Update Conda**: conda update conda
- **List Conda version**: conda --version

3. Managing Environments

- **Create a new environment**: conda create --name myenv
- **Create an environment with a specific Python version**: conda create -n myenv python=3.8
- **Activate an environment**: conda activate myenv
- **Deactivate current environment**: conda deactivate
- **List all environments**: conda env list
- **Delete an environment**: conda env remove --name myenv
- **Clone an environment**: conda create --clone myenv --name myclone
- **Export environment to a YAML file**: conda env export > environment.yml
- **Create environment from a YAML file**: conda env create -f environment.yml
- **List packages in an environment**: conda list -n myenv
- **Update all packages in an environment**: conda update --all -n myenv
- **Check for environment compatibility issues**: conda doctor

4. Managing Packages

- **Install a package**: conda install numpy
- **Install a specific version of a package**: conda install numpy=1.18
- **Install multiple packages**: conda install numpy scipy pandas
- **Install packages from a specific channel**: conda install -c conda-forge numpy
- **Update a package**: conda update numpy

By: Waleed Mousa

- **Uninstall a package**: conda remove numpy
- **List installed packages**: conda list
- **Search for a package**: conda search numpy
- **List packages with updates available**: conda search --outdated
- **Install a package directly from a URL or local file**: conda install ./package.tar.bz2
- **Freeze installed packages to a file**: conda list --export > package-list.txt
- **Install packages from a list file**: conda install --file package-list.txt

5. Managing Channels

- **Add a channel**: conda config --add channels conda-forge
- **List configured channels**: conda config --show channels
- **Remove a channel**: conda config --remove channels conda-forge
- **Set channel priority**: conda config --set channel_priority strict

6. Miscellaneous

- **Clean unused packages and caches**: conda clean --all
- **Find package dependencies before installing**: conda search numpy --info
- **Update Conda and all packages**: conda update conda && conda update --all
- **Use Conda environment in Jupyter Notebook**: conda install nb_conda
- **Set up Conda-forge as the default channel**: conda config --add channels conda-forge --force
- **List all Conda commands and their options**: conda --help
- **Generate a list of licenses for installed packages**: conda list --explicit
- **Export the active environment's package list to a YAML file**: conda env export --name myenv > myenv.yml
- **Create a custom Python environment for data science**: conda create -n ds python=3.8 numpy scipy pandas matplotlib seaborn scikit-learn
- **Resolve package conflicts with a comprehensive search**: conda search <package> --info --reverse-dependency

By: Waleed Mousa

- **Backup and restore Conda environments**: conda create --name myenv_backup --clone myenv & conda env create -f myenv_backup.yml
- **Integrate Conda with Bash (Linux/Mac)**: echo ". /path/to/anaconda3/etc/profile.d/conda.sh" >> ~/.bashrc && source ~/.bashrc

7. Advanced Package Management

- **Pin a package to a specific version**: echo "numpy=1.18" > /path/to/env/conda-meta/pinned
- **Exclude a package from being updated**: conda update --all --freeze-installed
- **Search for a package in all channels**: conda search --override-channels -c defaults -c conda-forge numpy
- **List installed packages and their origins**: conda list --show-channel-urls
- **Reinstall a package (force install)**: conda install --force-reinstall numpy

8. Environment Variables Management

- **Set an environment variable in an active Conda environment**: conda env config vars set MY_VAR=my_value
- **Unset an environment variable**: conda env config vars unset MY_VAR
- **List environment variables set within Conda**: conda env config vars list
- **Activate changes to environment variables**: conda activate myenv

9. Managing Python Versions

- **Upgrade Python version in an environment**: conda install python=3.9 --update-deps
- **Downgrade Python version safely**: conda install python=3.7 --update-deps --force-reinstall

10. Working with Conda Environments at Scale

- **Share an environment with others by exporting to YAML**: conda env export --no-builds > environment.yml

By: Waleed Mousa

- **Replicate an environment on another machine**: conda env create -f environment.yml
- **Create environments from a requirements.txt file (Pip)**: conda create --name myenv --file requirements.txt
- **Generate a requirements.txt file from an active Conda environment**: conda list --export > requirements.txt

11. Debugging and Solving Environment Issues

- **Check for issues in the current environment**: conda info --envs
- **Solve package conflicts**: conda install --update-deps --force-reinstall problematic_package
- **List dependency tree for a package**: conda list --revisions
- **Restore an environment to a previous revision**: conda install --revision 1

12. Automation and Scripting

- **Automate environment setup with a script**:

```
#!/bin/bash

conda create --name myproject python=3.8 -y

conda activate myproject

conda install numpy pandas matplotlib -y
```

- **Use Conda environments in batch jobs or scripts**:

```
#!/bin/bash

source activate myenv

python myscript.py
```

13. Integrations and Extensions

- **Integrate Conda with Docker**: Use FROM continuumio/miniconda3 in a Dockerfile to base your Docker container on Miniconda.
- **Use Conda environments with VS Code**: Install the Python extension for Visual Studio Code and select the interpreter from a Conda environment.

By: Waleed Mousa

- **Manage Conda environments in PyCharm**: In PyCharm settings, add a new Python interpreter and choose Conda Environment.

14. Performance Optimization

- **Optimize Conda's solver with Mamba**: Install Mamba with `conda install mamba -n base -c conda-forge` and use it as a drop-in replacement for Conda (e.g., `mamba install numpy`).

15. Miscellaneous Tips

- **Use Conda's `--dry-run` option for safe testing**: `conda install --dry-run package_name`
- **Export installed packages in an environment to JSON**: `conda list --json > packages.json`
- **Bulk remove packages not dependent on others**: `conda remove --force-remove package_name`