

WARNING: This is not the final version of
this file, more updates are coming

Module 0: Data Science Hacking Tools

CSCI1360: FOUNDATIONS FOR INFORMATICS AND DATA
ANALYTICSSCI

Objectives

- Use Command prompt\Terminal for OS basic operations
- Use Conda to install coding tools
- Use Jupyter Notebook to write Python code
- Create GitHub account
- Use Git to interact with GitHub repositories

Command Prompt\Terminal

- Windows: Command Line Prompt or CMD
- Linux-based or Mac OS: Terminal
- Used to interact with the OS
 - To perform file \ folder operations
 - To start a program
 - Some programs are developed without Graphical User Interface (GUI)
 - They run in the terminal
 - Interaction happens through entering commands

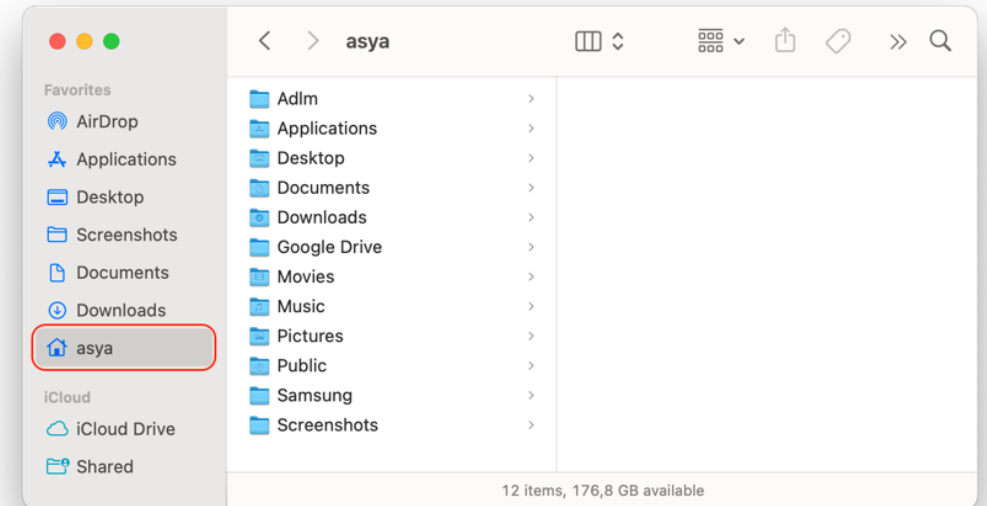


Important CMD\Terminal Commands

LINUX	WINDOWS	DESCRIPTION
ls	dir	List files and folders in the current working directory.
cd	cd	Change the current working directory.
cp	copy	Copy a file or folder to a new location.
mv	move / rename	Move or rename a file or folder..
mkdir	md	Make a new folder or subdirectory.
rm	del or rmdir	Delete a file or folder.
echo	echo	Send some text to the console screen.
cat	type	Show the contents of a text-based file
pwd	cd	Prints current path

Home Folder

It is a personal space where users can store files, create folders, and manage data without worrying about other users on the same system.

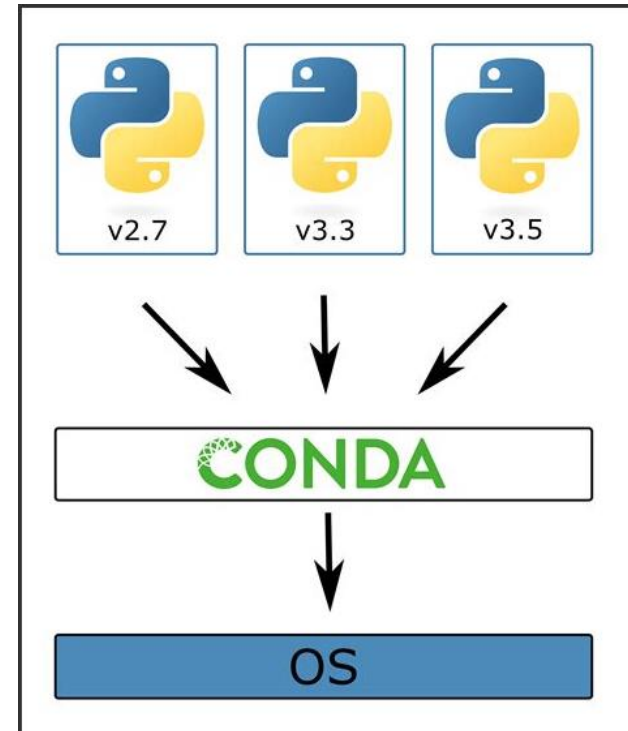


Create Course Folder

LINUX	WINDOWS	DESCRIPTION
cd ~	cd c:\Users\<user_name>	Change/Point the current working directory to Home
ls	dir	List files and folders in home directory
cd Desktop	cd Desktop	Point to your Desktop folder
mkdir csci1360	md csci1360	Create csci1360 in Desktop
cd csci1360	cd csci1360	Point to csci1360
mkdir assignments	md assignments	Create (assignments) folder
mkdir notebooks	md notebooks	Create (notebooks) folder
cd ..	cd ..	Point to the parent folder (Desktop)
cd csci1360	cd csci1360	Point to csci1360
ls	dir	Should see the two new folders

Conda

- Conda is an open-source, cross-platform, language-agnostic package manager and environment management system. [Wikipedia]
- A Conda environment is a self-contained directory that contains a specific collection of software packages, along with their dependencies.
- Conda allows users to install different versions of software packages in different environments.



Conda

- Regular coding environment (No Conda)
 - Packages depend on system libraries
 - Install packages with all the required steps by the vendor
 - One environment for all packages
- Conda environment
 - Isolated multiple environments
 - Packages depend on Conda libraries
 - One step package installation
- Consider using Miniconda for this course

Miniconda Installation

- Link: <https://docs.conda.io/en/latest/miniconda.html>
- Mac OS
 - 64-bit bash package (.sh package)
 - Open terminal
 - Navigate to the package download folder
 - `bash <package-name.sh>`
- Windows
 - Download and install (Miniconda3 Windows 64-bit)
 - Use the default installation settings
 - Search (Anaconda Prompt)

Conda Basic Commands

Using environments

Create a new environment named py35, install Python 3.5

```
conda create --name py35 python=3.5
```

Activate the new environment to use it

```
WINDOWS: activate py35  
LINUX, macOS: source activate py35
```

Get a list of all my environments, active environment is shown with *

```
conda env list
```

Make exact copy of an environment

```
conda create --clone py35 --name py35-2
```

List all packages and versions installed in active environment

```
conda list
```

Installing and updating packages

Install a new package (Jupyter Notebook) in the active environment

```
conda install jupyter
```

Run an installed package (Jupyter Notebook)

```
jupyter-notebook
```

Deactivate the current environment

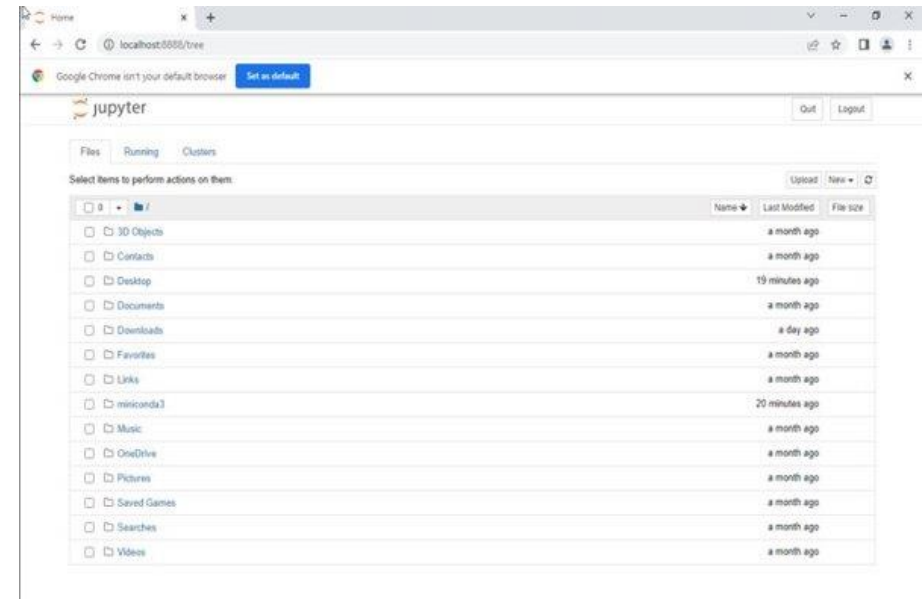
```
WINDOWS: deactivate  
macOS, LINUX: source deactivate
```

Create Conda Environment

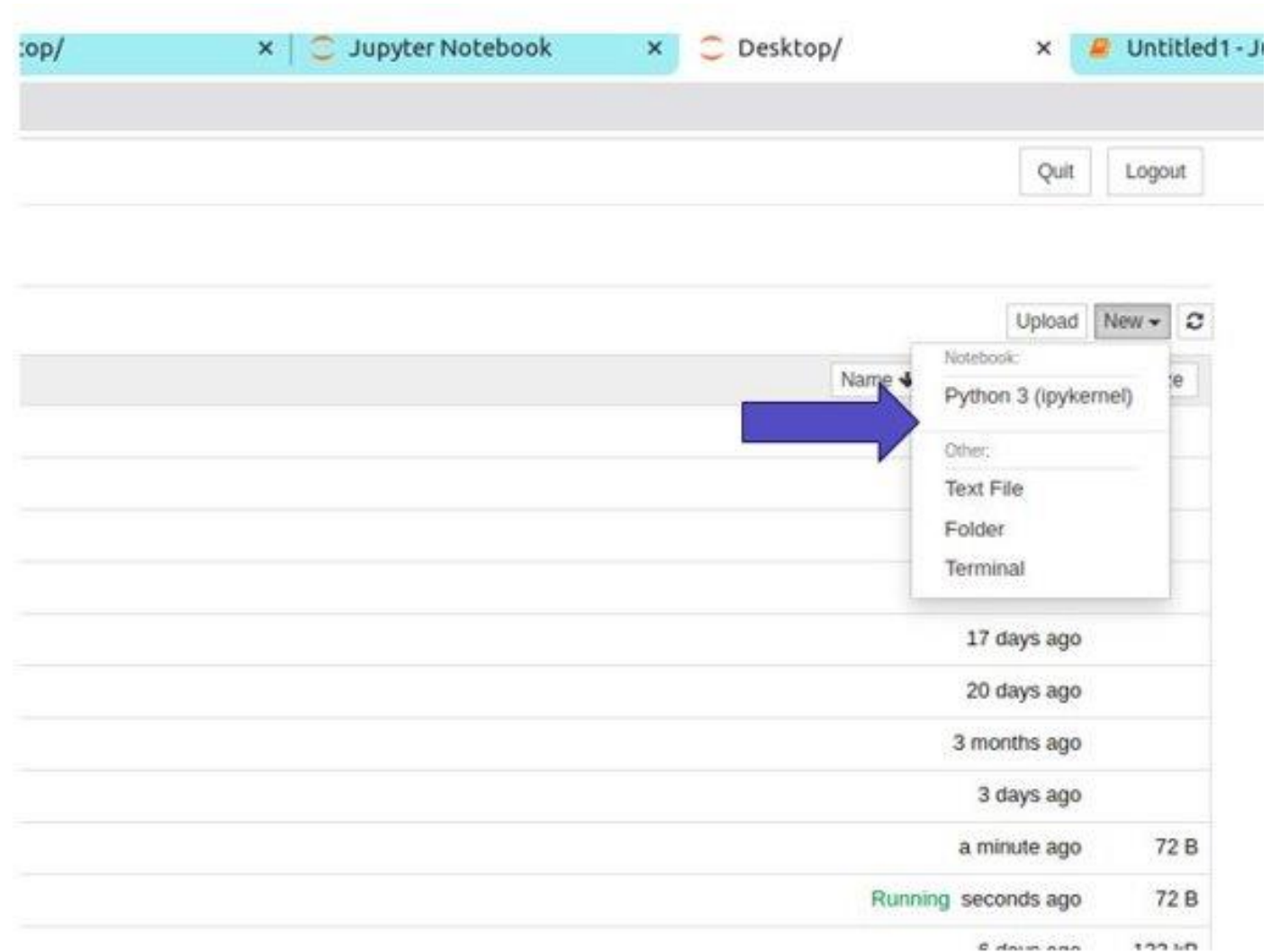
- Conda creates (base) environment by default
 - You may use it for this course or create another one
- To create another environment
 - `conda create --name csci1360 python=3.9`
- Activate the environment
 - `conda activate <environment-name>`
- Packages to install for this course in conda
 - Notebook: `conda install notebook`
 - Git: `conda install git`

Jupyter Notebook

- Open-source web-based coding platform
- Features
 - Easy to use
 - Live code and visualizations output in cells
 - Enable explanatory text in markdown cells
- Start Jupyter Notebook
 - Open the terminal
 - Activate the Conda environment
 - Run: `jupyter notebook`
 - Jupyter Notebook will start in the same terminal pointing folder



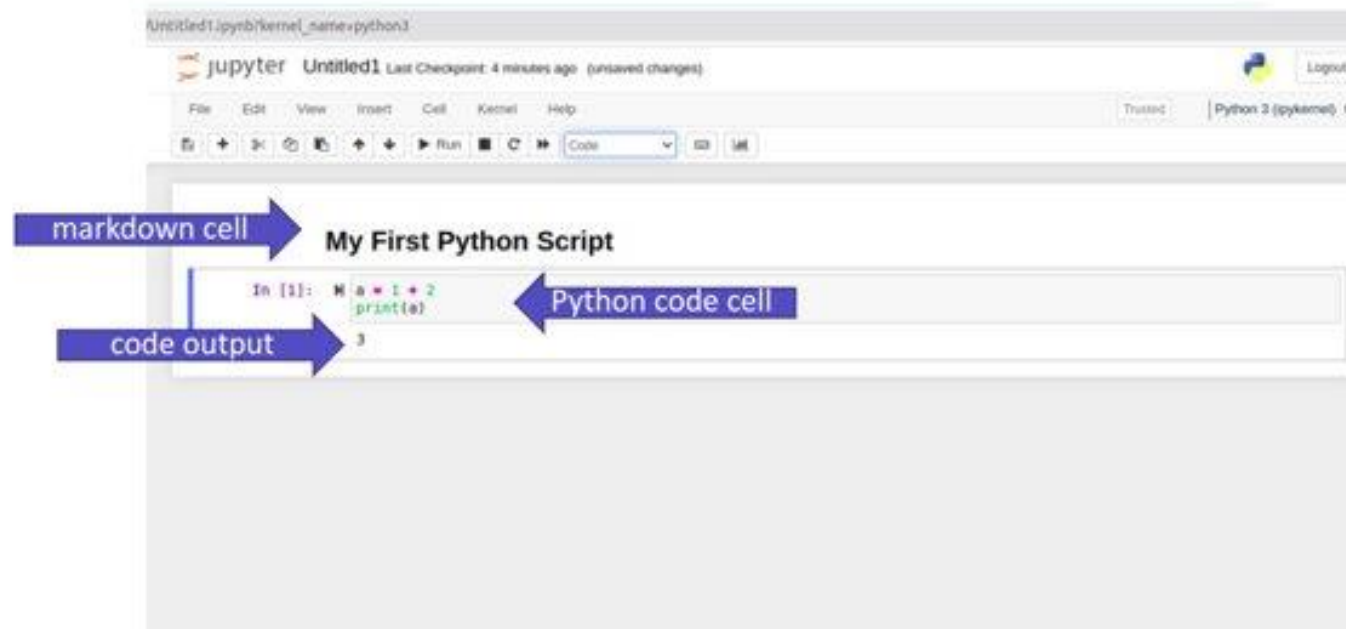
Create Your First Notebook



Create Your First Python Script

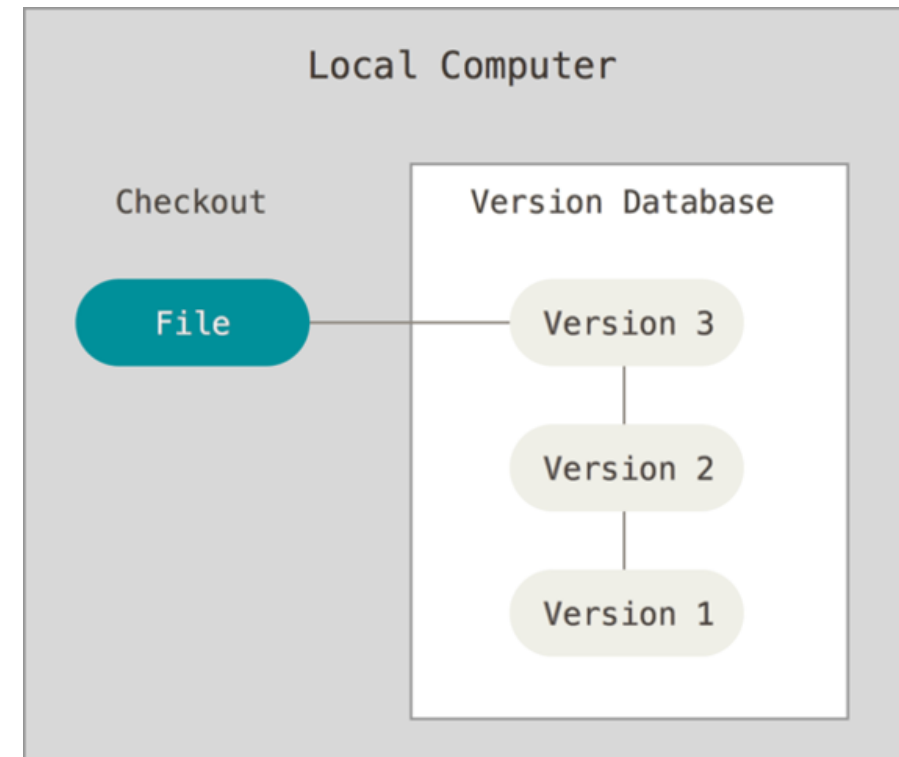
Click the Run button to execute the code.

Notebook files will be used for assignment solutions and class slides.



Git

- Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later
- Usages
 - Local tracking environment
 - Online collaboration environment
- Create a local repository
 - Navigate to the target folder using the terminal
 - Run: `git init`
- Or, clone an online repository



GitHub

- GitHub is a hosting service for software development and version control using Git
- Cloud based Git repository
- Web interface and commands interface through Git
- A repository contains all of your project's files and each file's revision history.



GitHub Account

- Use your UGA email for this account
- Generate private token to be used when submitting your assignment solutions
- Got to:
 - settings
 - Developer settings
 - Personal access tokens
 - Tokens
 - Generate new token (Classic)
 - Set Expiration
 - click "repo" checkbox
 - Generate token
- Do not make the token file publicly accessed

GitHub Account Information

Enter your UGA using the link below

○ <https://forms.office.com/r/31Cz7gwJnG>



GitHub Classroom

- A platform that helps teachers and students interact in courses that incorporate coding
- The course account:
 - <https://github.com/UGA-CSCI1360>
 - Hosts repositories for assignment and lectures
- Steps to work on an assignment
 - Get your assignment repository created through the assignment link
 - Clone the repository to your local course folder
 - Add your solution to the notebook template
 - Commit and push your changes using Git

Git Basics

- `git clone <url>`
 - To clone an online repository to your local environment
- `git add <file>`
 - Start tracking a file
 - The file will be a part of the repository
- `git commit -m "my commit note"`
 - Commits changes, new version of your code will be stored in Git database
 - Other changes after last `add` command won't be considered
 - Unless you run: `git commit -a -m "my commit note"`

Git Basics

- `git status`
 - to determine which files are in which state
- `git push`
 - Updates the online repository with your local changes
 - Submits your assignment solution
- `git config --global user.email "you@example.com"`
 - to set your GitHub account for push commands
- `git config --global user.name "Your Name"`
 - to set your name for push commands