There's a lot of vocabulary involved in software development. This document is meant to be a community resource to keep track of vocab words/definitions. Please add to the definitions of any words you know and/or add words (without definitions) that you'd like to see defined. TAs & instructors will also work on this document.

Abstraction

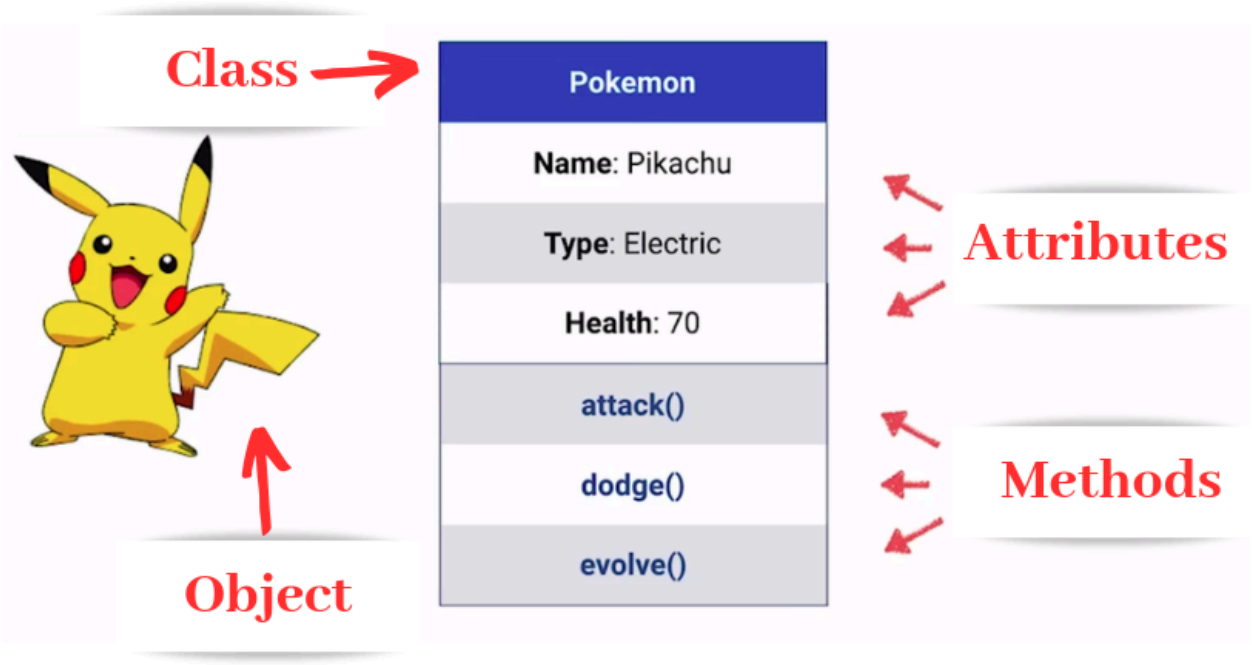
When you kind of hand-wave the unimportant details, usually to focus on the problem at hand. For example when you import a package you abstract away the implementation details of the package and silently thank the people who made it.

Branch

A "branch" diverges from the main line of development in a version control system. In git this is a pointer that points to the latest commit in that diverged line of development.

Class

An encapsulation of variables and functions that operate together.



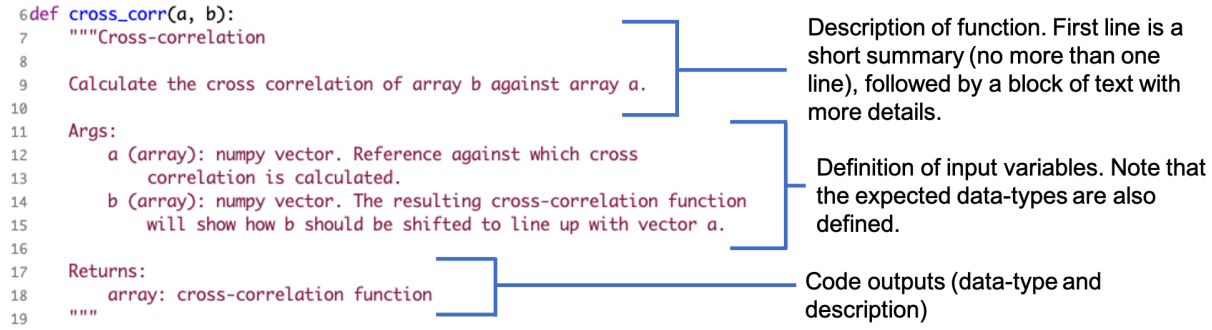
Commit

(definition modified from wikipedia): In version control systems, a *commit* is an operation which sends the latest changes to the source code to the repository, making these changes a permanent part of the repository. Several changes to the code can be packaged together in one "commit."

Directory

A filesystem cataloging structure that contains files. Also known as “Folder”.

Docstring
a comment at the beginning of a function or method, in a standardized form. “Google-style” docstrings, which we use in this workshop, look like this:



Environment
Any place where code is run. Examples of environments: the terminal, a jupyter notebook, the iPython console, the matlab IDE, etc.

Forked Repository
A forked repository is a remote that you, the forker, can modify. It is common for open-source projects, when the official remote can only be edited by its maintainers and contributors make changes to their own forks before submitting a pull request to the official remote.

Git
Version control program invoked from the command line. Interfaces with online hosting tools like GitHub & Bitbucket.

GitHub
Website used for facilitating collaborative use of the git command line tool

Handler
A software routine that performs a task. It “handles” it for you.

IDE
Stands for Integrated Development Environment; a software application that allows programmers to edit code and comes with additional tools (debuggers, etc.)

Inheritance
If an object inherits from a parent object, it has all of the functionality of the parent object, plus anything you want to add specifically for that object. This is great for situations where you need

to do the same thing over and over again slightly differently each time. One example of classes inheriting from a common base class is the priors.py module in orbitize.

Instance
An occurrence of an object at runtime.
For example the number “1” is an instance of the immutable object of type int (which stands for integer).

Immutable
An object that can not be changed after it has been instantiated.

Memory
The short-term data storage for your computer, it is very fast and typically where information for immediate use gets stored. Where runtime often happens.

Merge
Merging two branches consolidates the changes from both into one. More details: https://www.atlassian.com/git/tutorials/using-branches/git-merge (helpful visuals in this article!)

Module
Any Python file. More generally, a set of related functions, classes, and/or other Python objects.

Mutable
An object that can be changed after it is created.

Object
A specific instance of a class. Objects of the same class typically have the same variables and functions, but those variables and functions do not necessarily return the same value.

Origin
Shortcut for git remote branch can be anything you like but origin is standard.

Path
A string that specifies the location of a file or directory in a file system. For example, “/usr/bin/local/example.txt” is a path that describes the location of the file “example.txt”

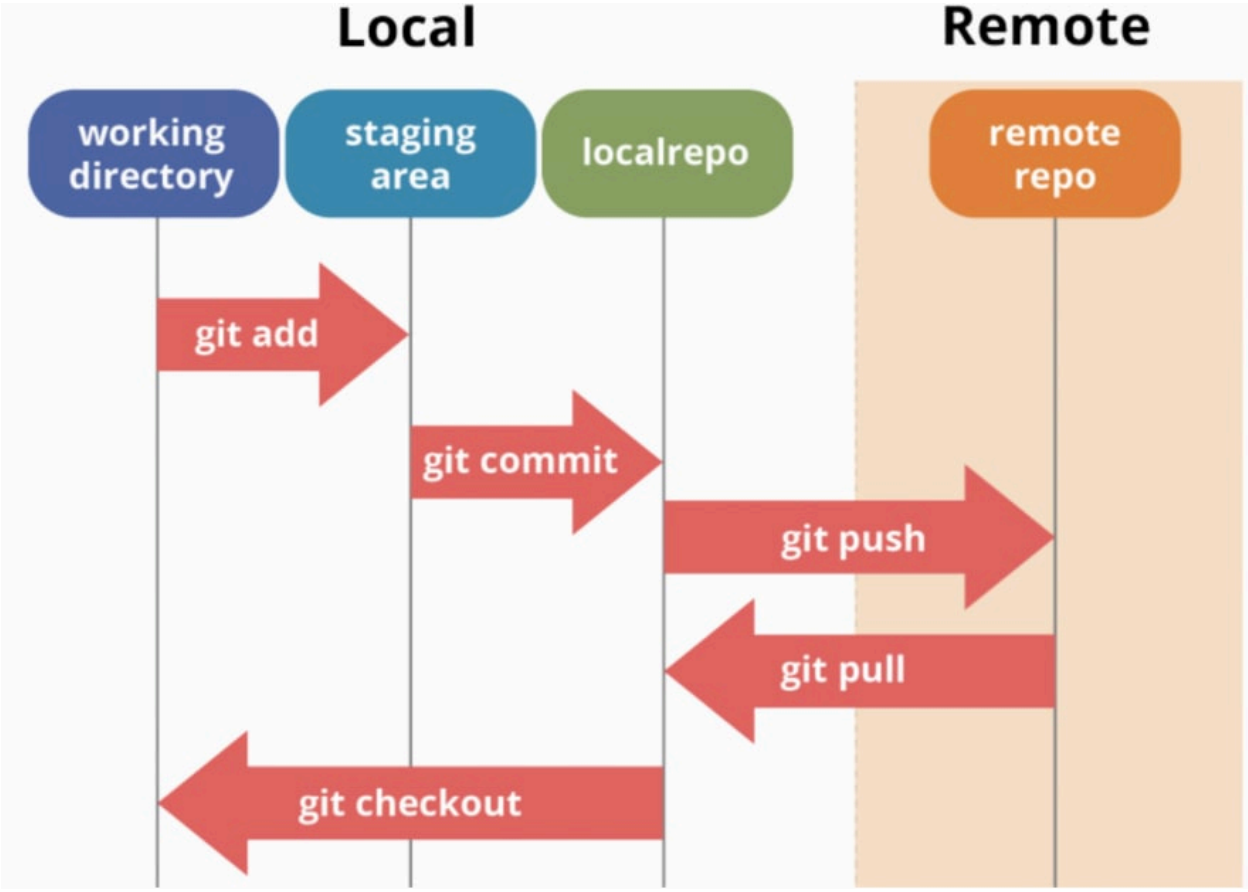
Plain Text Editor
A software application that lets you read/write/edit text. Does not come with any additional tools.

Pull
(also pull request, or PR): when changes to a repository from branch A are being merged into branch B, this is called “pulling” or “pulling in” changes from A. Also used in the context of pulling changes made on a remote server (e.g. GitHub) to the local copy of the repo.

Push

The opposite of pull. When changes to a repository from branch A are being merged into branch B, this is called “pushing” changes from A onto B

(Graphic from https://dilip-parmar.in/git-basics/)



Remote

A repository that lives on a server (e.g. Github) that keeps the common history of the project when people are collaborating. Any change pushed to the remote must not conflict with it (unless the push is forced).

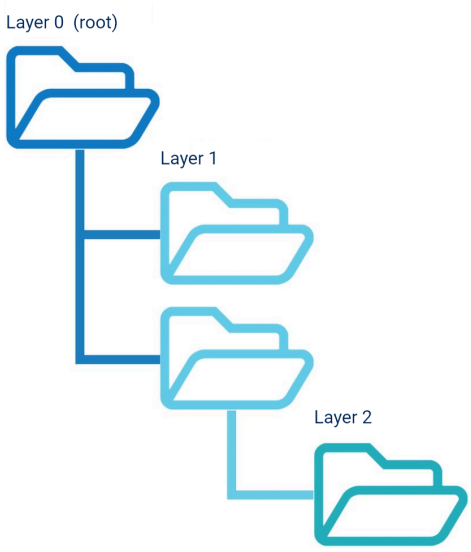
Repository (also called “repo” for short)

A *repository* for a version control system tracks all changes made to files in your project over time.

Root directory

The bottom-most directory in some context. For example, the “overall” root directory is “/” (the lowest, or base, directory). In the context of a package, the root directory is the lowest directory you can go to and still be inside the package.

In orbitize!, the file structure looks like: /path/to/orbitize/orbitize/kepler.py. The root directory would be /path/to/orbitize/



(Graphic from Sistrix)

Runtime
A phase in computing, particularly when the computer is executing (or running) lines of code. It refers to how long it takes the code to run, from start to finish.

Stack Trace
When you run a piece of Python code and it fails, the stack trace is the full error message printed out. It generally contains several lines, each of which is the line at which the code failed. For example, if my code fails on a line that invokes some numpy command, the last line of the stack trace will be the line of my code at which the failure occurred.

State
The state of any environment can be thought of as a list of pre-set variables that define the default behaviors. For example, when you open a terminal, several variables are “pre-loaded,” such as $PATH. In the terminal, these are called “environment variables.”