Start 16/03/20201

By

Jedsada LUENGARAMSUK

Table of contents

# Operating system

* OS
  + Kernel
    - Manage system resources
  + User space
    - We interact with user space
      * System program and user interface

## Usage

* Managing files on the OS (Basics)
* Major operating system
  + Windows
  + MAC OS
  + Linux
* Unix > BSD
  + MacOS and Linux evolves from this.

Operating system and you! Course

python --version



Pypl

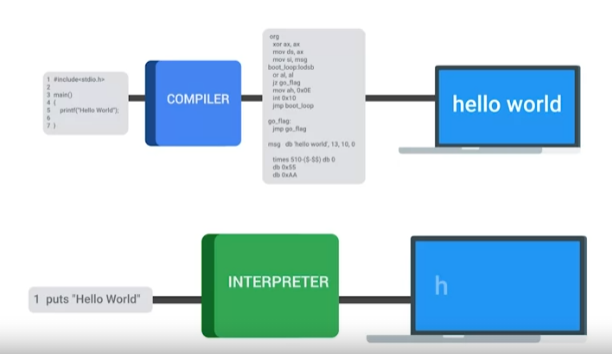
* Thousands of project
* Manage with command line tool called PIP
* Pip = python package manager

Install modules using pip

* pip install {module\_name}

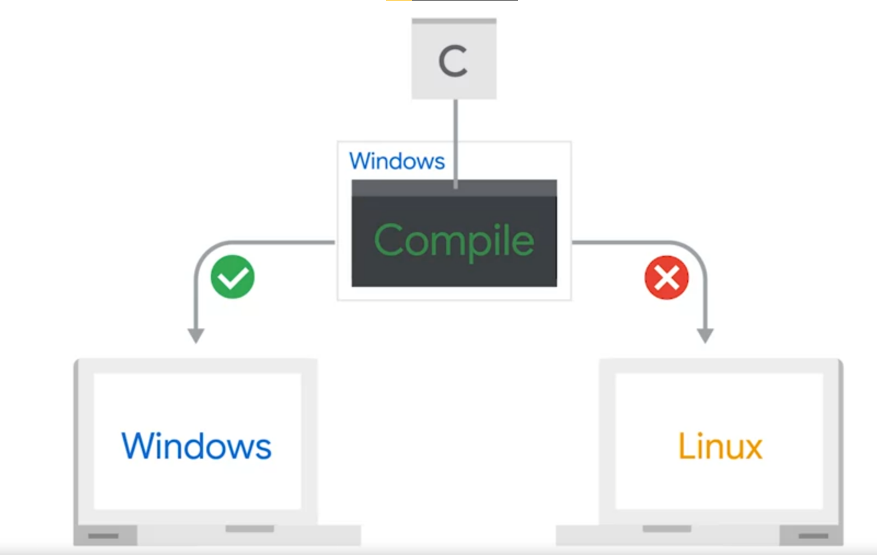
# Running Python Locally

## Interpreted vs. Compiled Languages

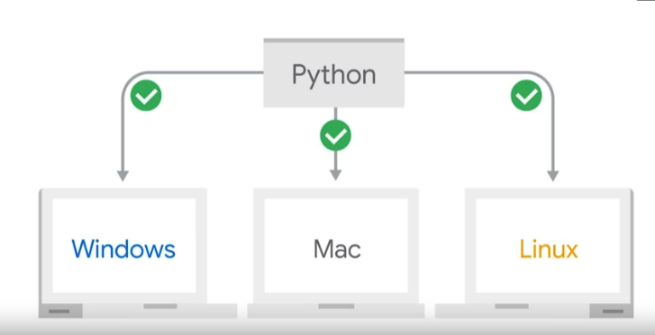


Interpreted language.

* compiler
  + Translate code to the machine language.
  + Super-fast to run but hard to write
  + C C++ GO RUST

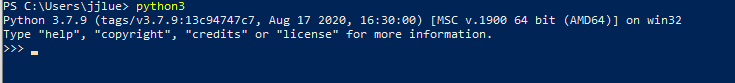


* Interpreter
  + Run without compiler.
  + Develop faster
  + Run in different systems.
  + Run slower.
  + Python, Ruby, Bash and PowerShell



Compiler need to complier into different operating system but interpreter doesn’t

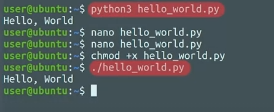
# Running a python Script



* We can run everything on command prompt
* Exit() to exit python in the interpreter
* Use vim to edit python text
* Use shebang line to make a default program run

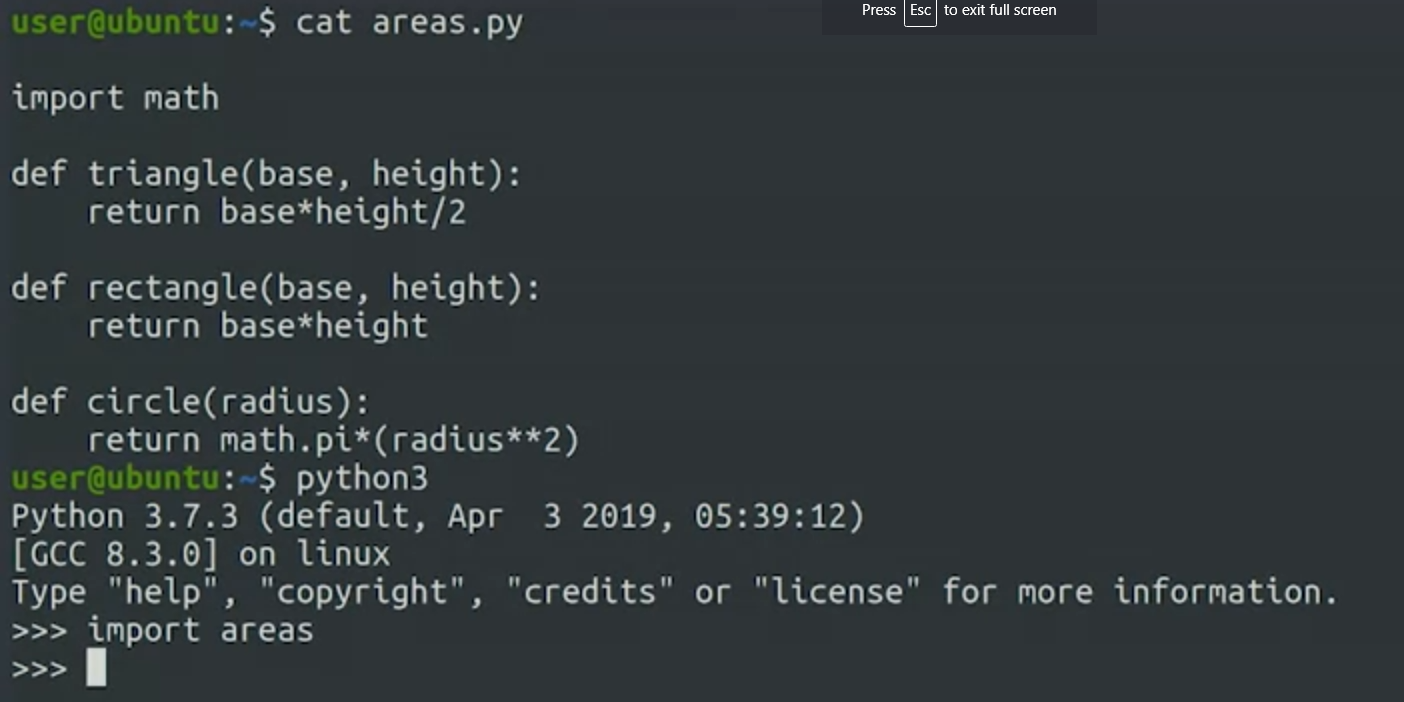
## Create executable sctrup

* Cdmod +x file <<< mark the script to be executeable
* ./hello\_world.py == python3 hello\_world.py /



# Code reuse (python modules)

* Modules help with tracking the reusing script.
* As script get more complicated modules is essential
* You can Import file as modules to use the functions



* Create modules by create functions
  + Create submodules
  + Requests

# IDEs

* Integrated Development environment
  + Features
    - Syntax highlighting
    - Code completion

# Automating Tasks

## Benefit of automation

* Increasing team member
* Can allow infrastructure to scale
* Scalability
  + When more work is added to a system, the system can do whatever it needs to complete the work
* If no IT, people can do each tasks manually.

## Pitfalls of automation

* Without thoughtful design, more problems created
* Is it time and effort worth it?
  + How many time will we do this task again?
  + 
* Pareto Principle
  + 20% of the system administration tasks that you perform are responsible for 80% of your work
* Bit-rot
  + Sometime the environment changed too fast and you cannot keep up with the script
* If automated system failed and you don’t realized it. This is the problem.
* Back up the wrong data (data corruption)
* System log can be useful log of information.

## Automation example

* shutil and psutil

# Qwikilabs

* virtual machine
  + If you don’t use it just delete it
  + Delete it once it done.
* Qwiklabs
  + Create virtual machine.
  + provisions resources backed by Google Cloud
  + Use linux operating system as if it were installing in the machine.

# To make python files execuable

1. Sherbang line
2. Use chmod +x filename
3. ./filename < to run the file