

Pontificia Universidad Católica de Chile  
Escuela de Ingeniería  
Departamento de Ciencia de la Computación



# IIC2115 - Programación como herramienta para la ingeniería

## Introducción

**Profesores:** Hans Löbel y Francisco Garrido Valenzuela

El curso busca **preparar y especializar** a los ingenieros en herramientas computacionales de desarrollo avanzado

# Motivación

Evaluar y utilizar de manera efectiva distintos lenguajes y herramientas de programación, para resolver problemas asociados a sus áreas de especialización, en base a los requerimientos de estos.

**Que lata (miedo) programar**



**Excelente, si lo programo sale mejor/más fácil**

Proponer y desarrollar soluciones novedosas utilizando la programación, ya sea para problemas tradicionales o para nuevos problemas en ingeniería.

**¿Es difícil/nuevo? Si no hay librería/software que lo haga, busco otra cosa**



**Ningún problema, voy a implementarlo yo mismo**

# Contenidos

El curso cubre múltiples temas divididos en **7 capítulos**

Introducción

Programación orientada a objetos

Estructuras de datos

Técnicas y algoritmos

Uso de bases de datos y archivos

Análisis de datos en Python

Tópicos avanzados

Todo el desarrollo estará basado en Python más algunas otras herramientas o librerías dependiendo del tema.

# Metodología de clase invertida obliga ser más responsable

- Cada capítulo se desarrollará durante dos o tres semanas, mediante 3 instancias: consolidación de materia, actividad práctica, y laboratorio, siendo estos últimos individuales y evaluados.
- La consolidación de materia se realiza el primer módulo de la primera clase, y se focaliza en resumir los aspectos más relevantes del capítulo.
- La actividad práctica se realiza el segundo módulo de la primera clase, y consiste en la resolución de un problema de programación corto, que involucre los contenidos del capítulo. Cabe destacar que el módulo para la actividad práctica tiene formato de taller, con el fin de facilitar la resolución de dudas.

# Metodología de clase invertida obliga ser más responsable

- Una vez finalizada la primera clase, se publica el enunciado del laboratorio. Este tiene una duración aproximada de dos semanas, y deberá ser entregado antes de las 23.59 del día indicado en cada enunciado.
- Para contribuir al desarrollo del laboratorio, la segunda sesión de cada capítulo, y la sesión de ayudantía, tendrán formato de taller.
- Entregas atrasadas de laboratorio deben ser avisadas hasta 24 horas posteriores a la fecha de entrega oficial, de lo contrario serán considerados como no entregado.

# Cronograma

	L	M	W	J	V	S	D
Marzo	9	10	11	<u>12</u>	13	14	15
	<u>16</u>	17	18	<u>19</u>	20	21	22
	23	24	25	<u>26</u>	27	28	29
	<u>30</u>	31	1	2	3	4	5
Abril	6	7	8	<u>9</u>	10	11	12
	<u>13</u>	14	15	<u>16</u>	17	18	19
	20	21	22	<u>23</u>	24	25	26
	27	28	29	<u>30</u>	1	2	3
Mayo	<u>4</u>	5	6	<u>7</u>	8	9	10
	11	12	13	<u>14</u>	15	16	17
	<u>18</u>	19	20	<u>21</u>	22	23	24
	25	26	27	<u>28</u>	29	30	31
Junio	1	2	3	<u>4</u>	5	6	7
	<u>8</u>	9	10	<u>11</u>	12	13	14
	15	16	17	<u>18</u>	19	20	21
	22	23	24	<u>25</u>	26	27	28
	29	30	1	2	3	4	5



Consolidación de contenidos  
Actividad de participación  
Publicación de laboratorio



Ayudantía en formato taller

# Evaluaciones

## **Laboratorios (L) → 60%**

Promedio de los 6 laboratorios.

## **Participación (P) → 30%**

Promedio de las 6 actividades evaluadas.

## **Asistencia (A) → 10 %**

7.0: Si asiste a las dos clases de cada laboratorio.

4.0: Si asiste a solo una de las clases.

1.0: Si no asiste a ninguna de las dos.



# El Syllabus

Todo lo relacionado con el curso se encuentra en el *Syllabus*

<http://github.com/IIC2115/Syllabus>

Es importante tenerlo continuamente actualizado (más sobre esto en un rato).

Dudas de materia o administrativas se pueden plantear como *Issues*, que serán respondidas oportunamente por el cuerpo docente del curso.

El *Syllabus* está montado sobre la plataforma GitHub, que usaremos de manera constante durante todo el semestre, tanto para el material de clases como para la entrega de laboratorios.

# Python y Git

Descargar Python 3.6

<http://www.python.org>

Descargar Git o instalar Xcode

<https://git-scm.com/>

# Jupyter Notebook / Colab

Jupyter Notebook es un entorno de desarrollo que permite crear y compartir documentos que contienen código fuente, ecuaciones, visualizaciones y texto explicativo.

Su interfaz se presenta como una página web, que nos permite, dentro de otras cosas, interactuar con código Python.

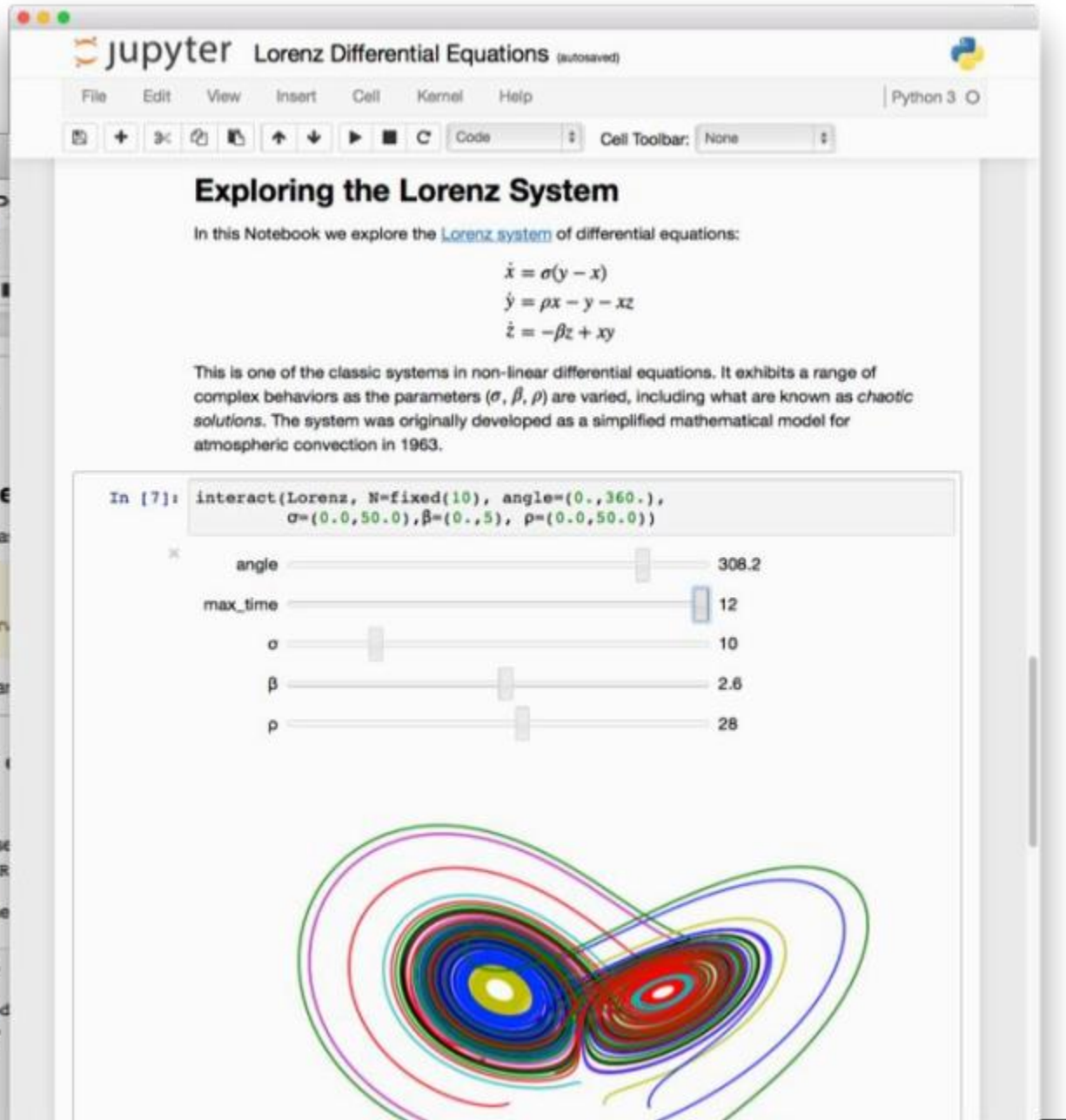
También pueden utilizar la plataforma Colab de Google:  
[colab.research.google.com](https://colab.research.google.com)

# ¿Por qué Jupyter Notebook?

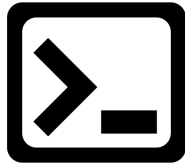
El uso de notebooks facilita el desarrollo rápido y la claridad del código.

Además, funciona en todas los sistemas operativos usados regularmente.

Será uno de los estándares para este curso (hay otras opciones si esta no les gusta).



# Terminal o CMD



Interfaz que permite interactuar con el computador

## Comandos

`cd [path]`

`cd ..`

`dir (windows)`

`ls (mac)`

*Last but not least...*

*Como miembro de la comunidad de la Pontificia Universidad Católica de Chile me comprometo a respetar los principios y normativas que la rigen. **Asimismo, prometo actuar con rectitud y honestidad en las relaciones con los demás integrantes de la comunidad y en la realización de todo trabajo,** particularmente en aquellas actividades vinculadas a la docencia, el aprendizaje y la creación, difusión y transferencia del conocimiento. Además, velaré por la integridad de las personas y cuidaré los bienes de la Universidad.*

# Introducción a git

Programación como  
herramienta para la ingeniería

Tomado casi en su totalidad de material desarrollado por Ivania Donoso y Antonio Ossa



# ¿Qué es git?

Git es un sistema distribuido de control de versiones, gratuito y open source, diseñado para manejar de pequeños a enormes proyectos de forma rápida y eficiente <sup>1</sup>

<sup>1</sup> <https://git-scm.com/>



# ¿Qué es GitHub?

Es una plataforma para alojar proyectos usando el sistema de control de versiones git



# Ventajas

- Versiones disponibles en cualquier momento
- Control de cambios
- Programar versiones en paralelo y luego juntarlas
- Múltiples backup de sus programas
- Trabajo en equipo fluido (No hay problemas como en Dropbox)

# Conceptos

1. **Working directory:** lugar en donde están los archivos que luego serán parte del repositorio
2. **Repositorio local (o repo):** lugar que contiene todos los archivos que han sido “commiteados”
3. **Repositorio remoto:** lugar que contiene todos los archivos que han sido “commiteados” en un servidor de GitHub

# Setup



# Obtener dirección

IIC2115 / Syllabus

Watch

2

Star

0

Fork

0

<> Code

Issues0

Pull requests0

Projects0

Wiki

Insights

No description, website, or topics provided.

3 commits

1 branch

0 releases

1 contributor

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

Hernan4444

Clase 00

Actividades	Clase 00	
Ayudantias	Carpetas	
Clases	Clase 00	
Controles	Carpetas	
Tareas	Carpetas	3 days ago
.gitignore	Carpetas	3 days ago
README.md	Carpetas	3 days ago

README.md

IIC2115 - Programación como Herramienta para la Ingeniería

Clone with HTTPS

Use SSH

Use Git or checkout with SVN using the web URL.

https://github.com/IIC2115/Syllabus.git

Open in Desktop

Download ZIP

# Clonar el repositorio

En la consola escribir

```
git clone https://github.com/IIC2115/Syllabus.git
```

Recuerda estar en la carpeta en la que quieren mantener el repo

**¿Cómo funciona git y GitHub?**







Git → Sistema de control de versiones

GitHub → Plataforma para alojar proyectos

## Comandos

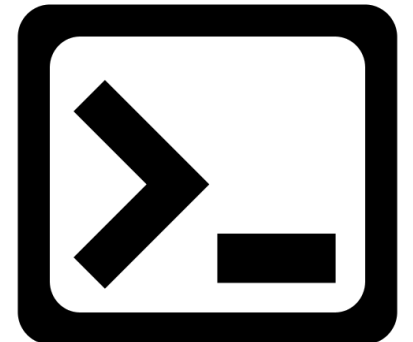
`git clone [link]`

`git add [path]`

`git commit -m "[msg]"`

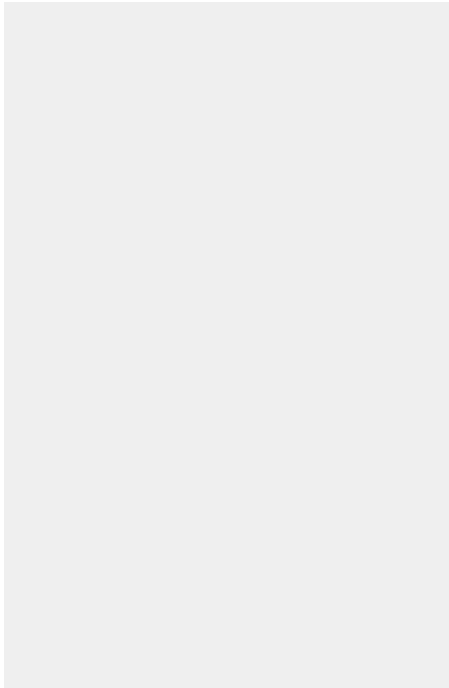
`git push`

`git pull`





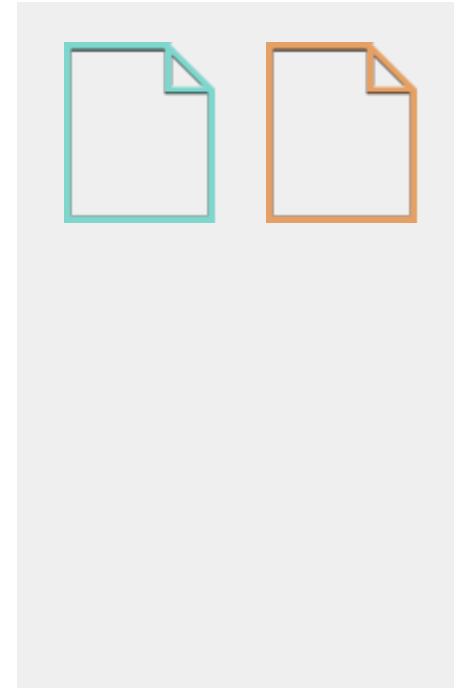
**Working directory**



`git clone [link]`



**Repositorio remoto**

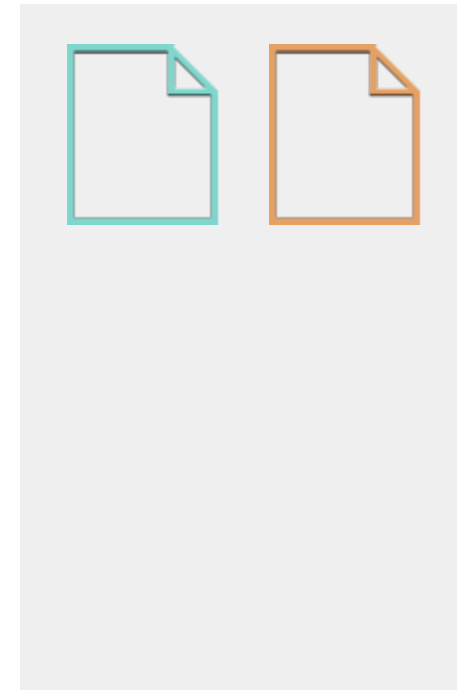
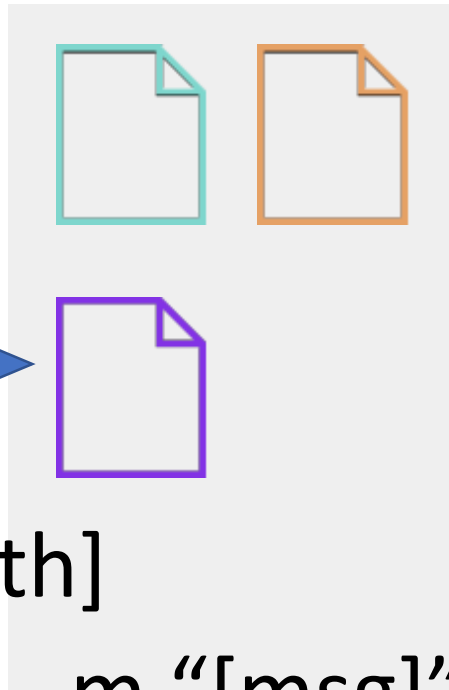




**Working directory**

**Repositorio local**

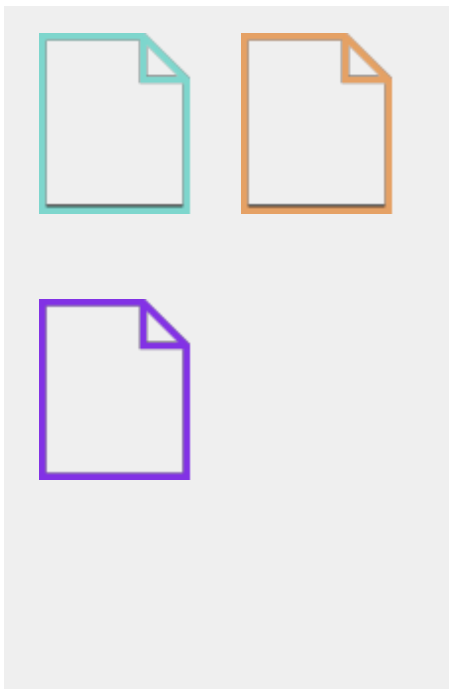
**Repositorio remoto**



`git add [path]`  
`git commit -m "[msg]"`



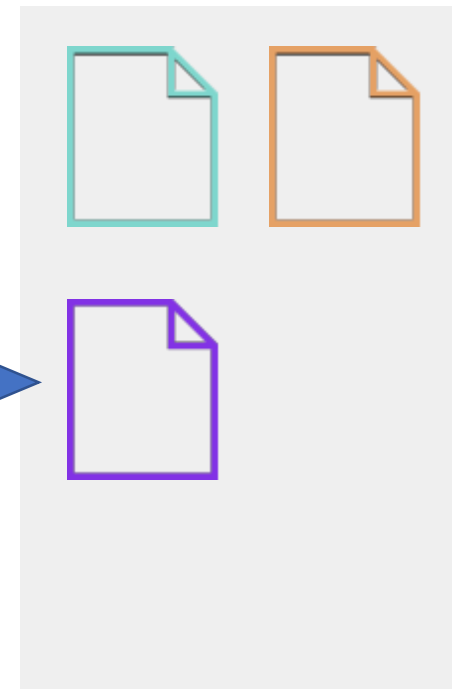
**Working directory**



**Repositorio local**



**Repositorio remoto**



`git push`



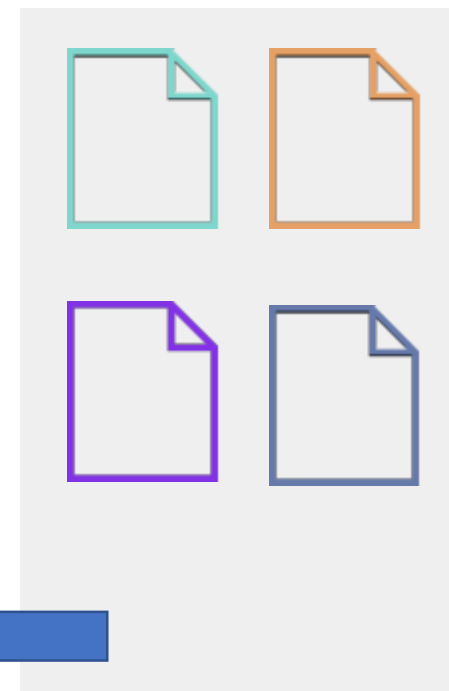
**Working directory**



**Repositorio local**



**Repositorio remoto**



`git pull`

# ¿Qué debo tener funcionando?

Git

Cuenta de GitHub → Página del curso y repositorio privado

Jupyter o Colab

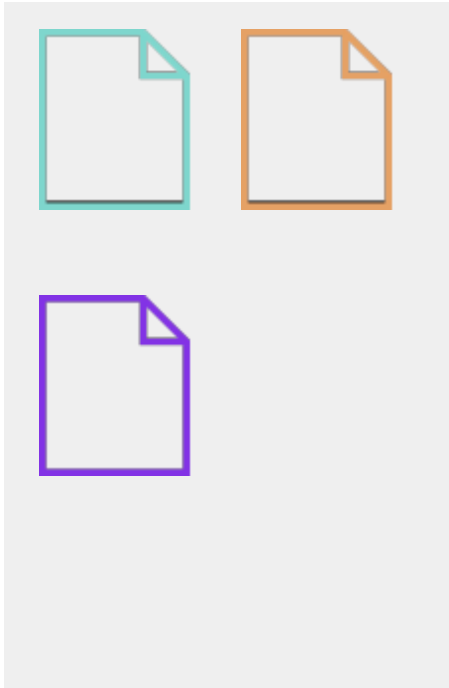
- <http://www.github.com/iic2115/syllabus> (materiales e issues)
- <http://www.github.com/iic2115/repositorio-2020-1-usuario>

**Click en WATCH al syllabus del curso**

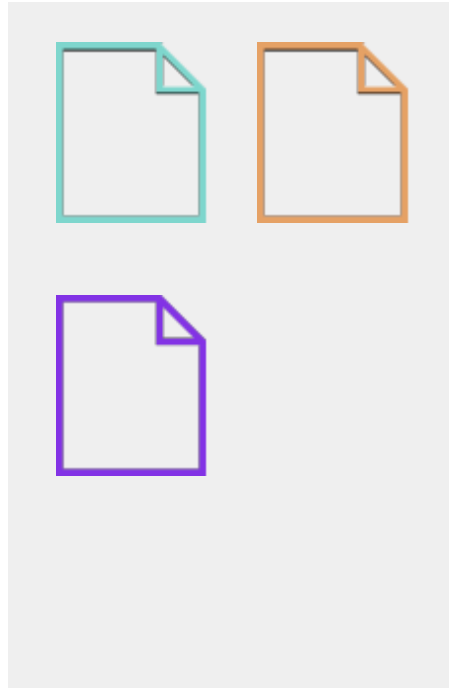
**Los mensajes son MUY importantes. Son una ayuda al ustedes del futuro.**

# Subir un archivo

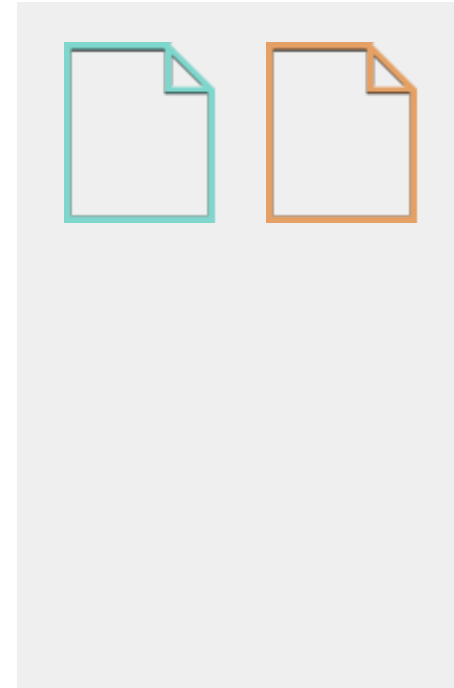
Working directory



Repositorio local



Repositorio remoto

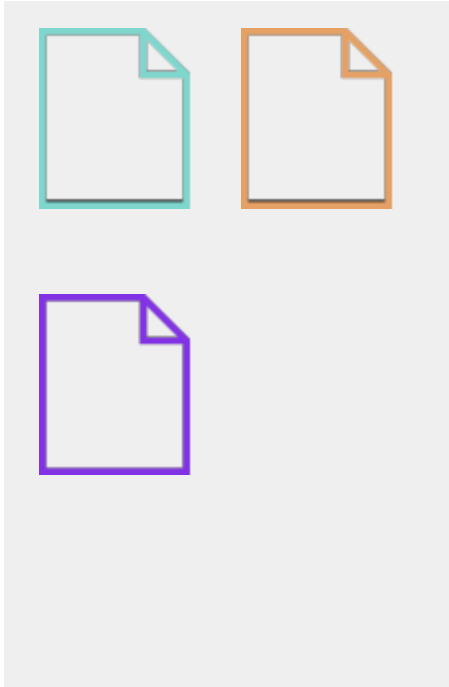


**git push**

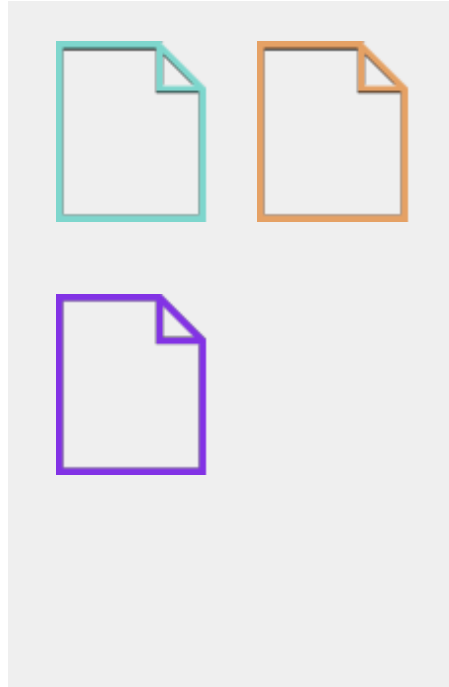


# Subir un archivo

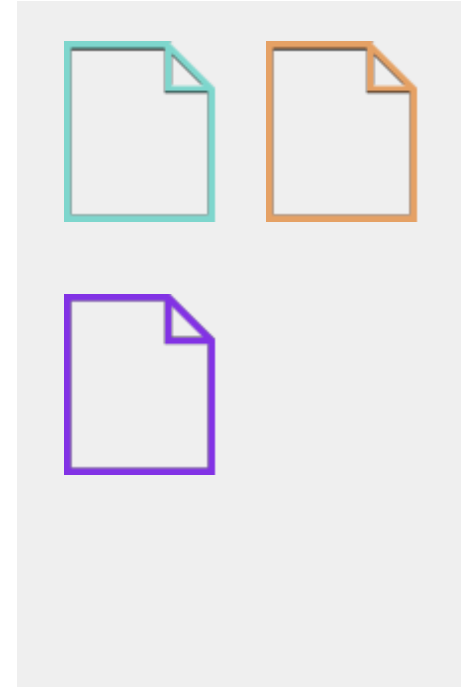
**Working directory**



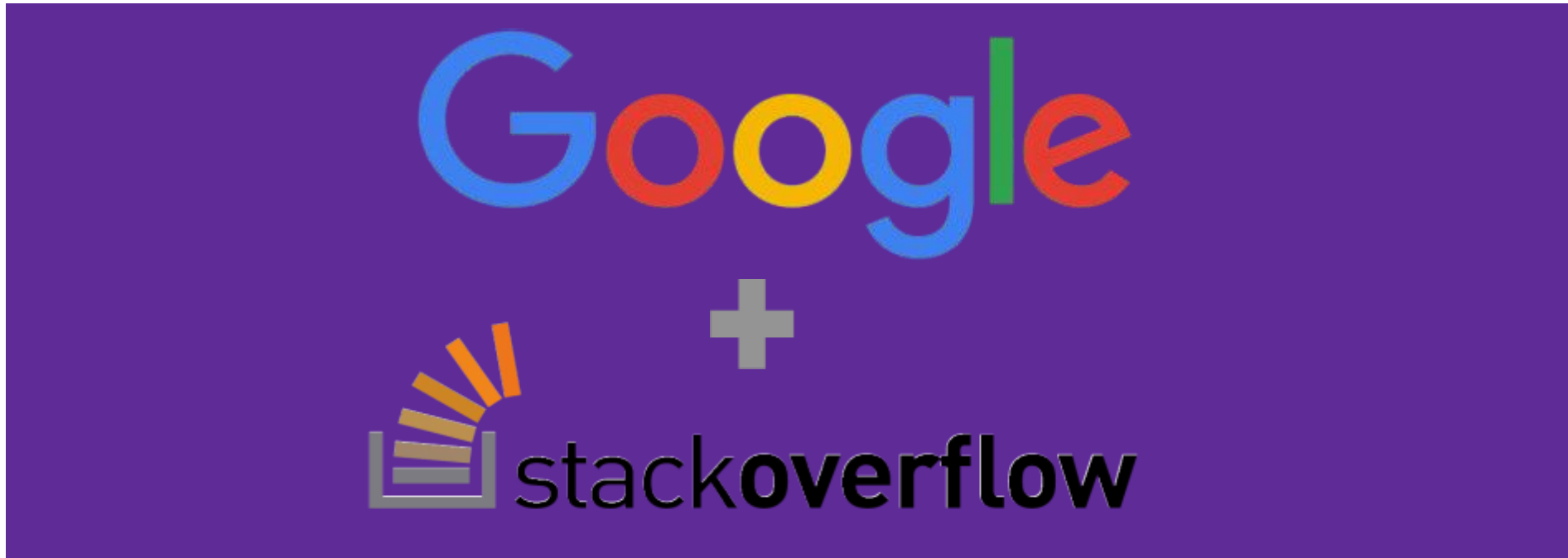
**Repositorio local**



**Repositorio remoto**



# ¿Cómo buscar soluciones?



# ¿Cómo buscar soluciones?

python [versión] [librería] [duda]



¡EN INGLÉS!

¿Cómo imprimir una cola con Python?



Python 3.5 collections print queue



# ¿Cómo buscar soluciones?

python [versión] [error]



¡EN INGLÉS!

NameError: name “MiVariable” is not defined



NameError: name \* is not defined





python3.5 NameError: name \* is not defined



Todos

Videos

Maps

Imágenes

Noticias

Más

Preferencias

Herramientas

Cerca de 95,800 resultados (0.50 segundos)

In Python3.5:NameError: name 'image\_to\_string' is not defined

<https://stackoverflow.com/.../in-python3-5nameerror-name-image-...> ▼ Traducir esta página

11 jun. 2017 - Please post your source code so we can look over the code and get more details. Also your error is caused by a variable declaration without a ...

oop - Python3 NameError: name 'method' is not defined - Stack Overflow

<https://stackoverflow.com/.../python3-nameerror-name-method-is-...> ▼ Traducir esta página

18 mar. 2016 - consider you have the function **defined** in the global scope: def recursive(x): if (x>5): print (x) recursive(x - 1). you would simply call this with ...

input() error - NameError: name '...' is not defined - Stack Overflow

<https://stackoverflow.com/.../input-error-nameerror-name-is-not-...> ▼ Traducir esta página

14 ene. 2014 - input\_variable = input ("Enter your name: ") print ("your name is" + input\_variable) ...  
input ("Enter your name: ") File "<string>", line 1, in <module> **NameError: name 'dude' is not defined**  
... I did what Kevin said and it is version 2.7.5! ... If you are using **Python 3.x**, raw\_input has been renamed to input .

python NameError: name 'file' is not defined in python 3.5 - Stack ...

<https://stackoverflow.com/.../python-nameerror-name-file-is-not-...> ▼ Traducir esta página

26 nov. 2015 - Traceback (most recent call last): File "c:\python3.5\lib\runpy.py", line .... python 3.x from this Q: python **NameError: name 'file' is not defined** But ...

python 3.x - NameError: name 'value' is not defined - Stack Overflow

<https://stackoverflow.com/.../nameerror-name-value-is-not-define-...> ▼ Traducir esta página

5 abr. 2014 - **NameError: name 'value' is not defined** ... variable defined in a function is not visible outside the function. ... answered Apr 5 '14 at 2:36

NameError: global name 'unicode' is not defined in Python 3 - Stack ...

<https://stackoverflow.com/.../nameerror-global-name-unicode-is-...> ▼ Traducir esta página

9 nov. 2013 - **Python 3** renamed the unicode type to str , the old str type has been replaced by bytes . if isinstance(unicode or str, str): text = unicode or str ...

Pontificia Universidad Católica de Chile  
Escuela de Ingeniería  
Departamento de Ciencia de la Computación



# IIC2115 - Programación como herramienta para la ingeniería

## Introducción

**Profesores:** Hans Löbel y Francisco Garrido Valenzuela