



El futuro digital  
es de todos

MinTIC TIC



UNIVERSIDAD  
DE ANTIOQUIA

Facultad de Ingeniería

«Misión  
TIC2022»

«Misión  
TIC2022»

SEMANA 1

Luisa Restrepo



UNIVERSIDAD  
DE ANTIOQUIA

Facultad de Ingeniería

Luisa Fernanda Restrepo.



El futuro digital  
es de todos

MinTIC



UNIVERSIDAD  
DE ANTIOQUIA

Facultad de Ingeniería



# Luisa Fernanda Restrepo G.

Formadora en cursos de  
programación.

Ingeniera de Sistemas.

Magister en Ingeniería.

Estudiante de Doctorado en  
Ingeniería.





El futuro digital  
es de todos

MinTIC



UNIVERSIDAD  
DE ANTIOQUIA  
Facultad de Ingeniería



# Presentación del curso

## Metodología

## Contenido

## Evaluaciones

## Tips





El futuro digital  
es de todos

MinTIC



UNIVERSIDAD  
DE ANTIOQUIA


Facultad de Ingeniería



# Presentación beneficiarios

slido

Tienes experiencia en programación?

 Start presenting to display the poll results on this slide.



El futuro digital  
es de todos

MinTIC




UNIVERSIDAD  
DE ANTIOQUIA  
Facultad de Ingeniería



¿Qué crees que se necesita  
para ser un buen  
programador?

slido

¿Qué crees que se necesita para ser un buen programador?

 Start presenting to display the poll results on this slide.





# Objetivo

Aunar esfuerzos técnicos administrativos y financieros para llevar a cabo procesos de formación académica en programación, mediante el desarrollo de las rutas de aprendizaje definidas por MinTIC, de acuerdo con el contenido curricular y demás características contempladas para el proyecto Misión TIC2022.





# Metodología - Recursos

- MoodleVPL
- Recursos del docente
- Carpeta compartida  
(Programas desarrollados por  
los beneficiarios)
- Correos de contacto

<https://lms.misiontic2022udea.com/login>

<https://github.com/LuisaRestrepo/MisionTIC2022-Ciclo1>

<https://bit.ly/3htvj5J>

[soportemisiontic@udea.edu.co](mailto:soportemisiontic@udea.edu.co)  
[permanenciamisiontic@udea.edu.co](mailto:permanenciamisiontic@udea.edu.co)



# Metodología - Recursos

*Por cada hora de clase un estudiante debe dedicar 2 horas de estudio fuera del aula.*



# Metodología – Reglas generales

- Pedimos la palabra a través de la herramienta “levantar la mano”
- Nos escuchamos “atentos a los micrófonos”
- Participamos activamente de las sesiones
- Nos respetamos nuestras perspectivas
- Nos enfocamos y concentramos en la sesión
- ¿Alguna otra regla?



# Contenido

- Semana 1 – Inducción a la algoritmia, Python y operaciones
- Semana 2 – Condicionales-ciclos-definición de requisitos funcionales
- Semana 3 – Subprogramas, métodos y funciones (parámetros, retornos) vectores, programación orientada a objetos.
- Semana 4 – Clases derivadas-arreglos de dos dimensiones.
- Semana 5 – Listas, pilas y colas.
- Semana 6 – Librería de Python-Interfaces gráficas.
- Semana 7 – Archivos JSON y CSV, Pruebas unitarias.



# Evaluaciones

SEMANA	RETO
SEMANA 3	RETO 1
SEMANA 4	RETO 2
SEMANA 5	RETO 3
SEMANA 6	RETO 4
SEMANA 7	RETO 5

## Evaluación

- Reto 1 y 2 - 10%
- Reto 3, 4 y 5 – 20%
- Inglés – 20%



# Tips

- Antes de iniciar clases, haber revisado los contenidos en Moodle.
- Repasar la clase anterior antes de cada clase.
- Aprovechar a los tutores.



## Correo electrónico

permanenciamisiontic@udea.edu.co



## Moodle

<https://lms.misiontic2022udea.com/>



## Facebook

Facultad de Ingeniería Universidad de Antioquia



## Instagram

Facultad de Ingeniería UdeA





El futuro digital  
es de todos

MinTIC



UNIVERSIDAD  
DE ANTIOQUIA  
Facultad de Ingeniería

‘Mision  
<TIC2022>’

# Tips - I

Mantenerse actualizado, y no quedarse solo con lo que brinda la universidad

vive digital  
Colombia  
Apps.co



amazon books  
edX

Organiza tu calendario académico



Google Calendar

Trabajar fuerte en un segundo idioma





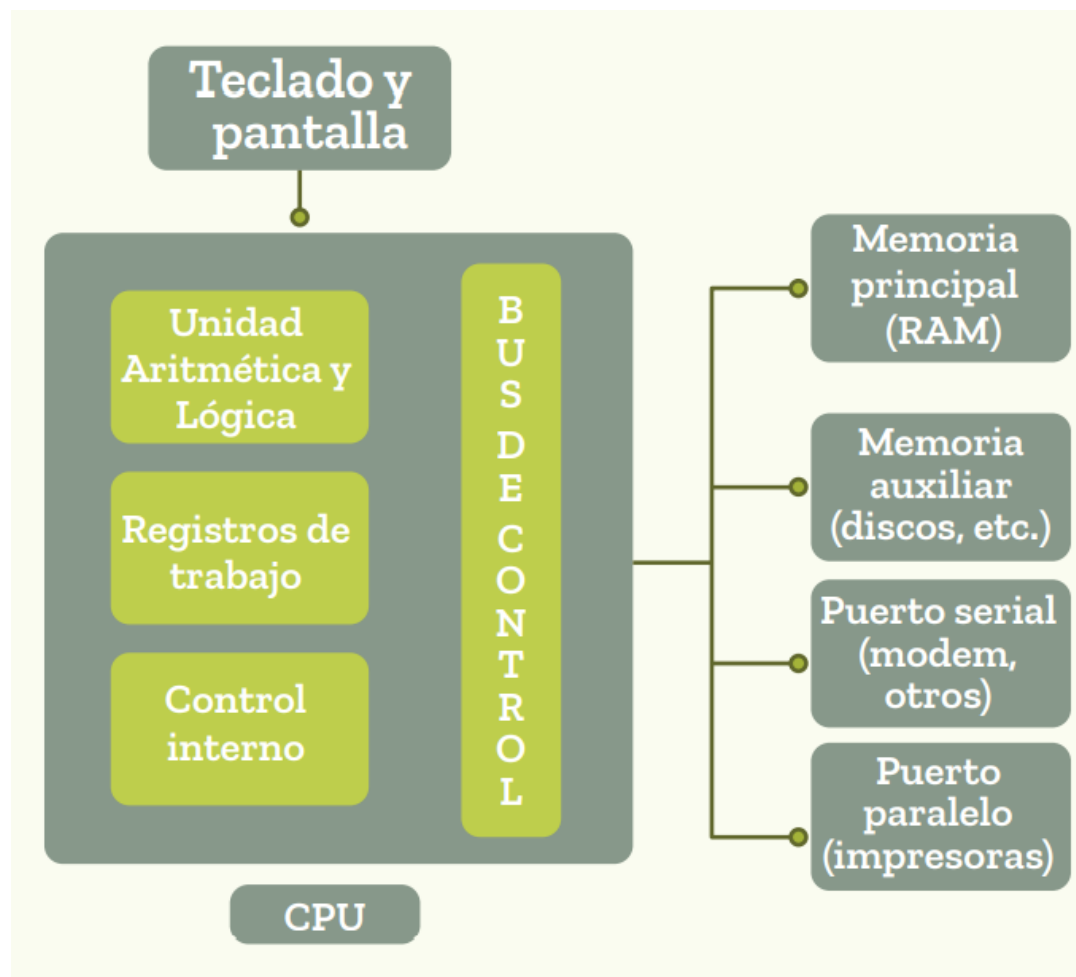


# Agenda

- Inducción a la algoritmia
  - Esquema de un computador
  - Algoritmos
- Introducción a Python
- Operaciones



# Esquema general de un computador





# Computadores

## ¿Qué es un computador?

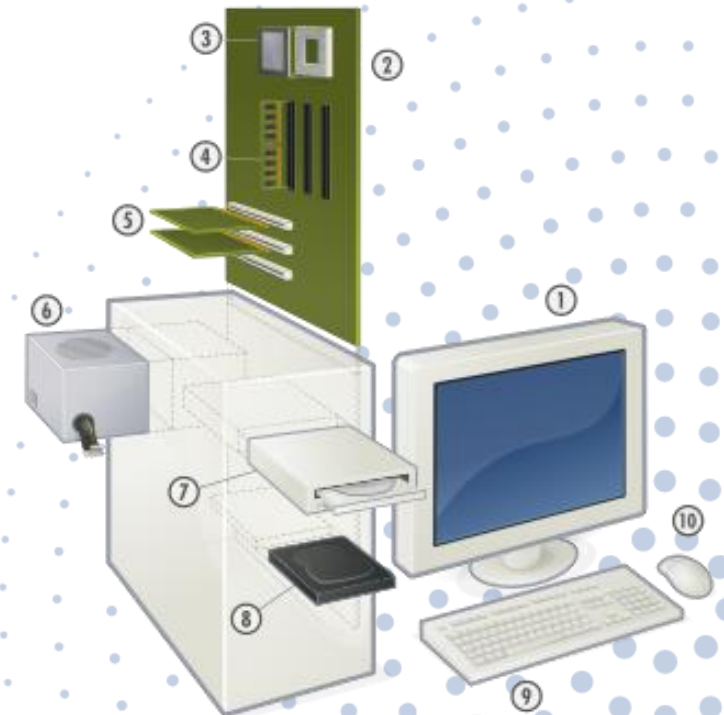
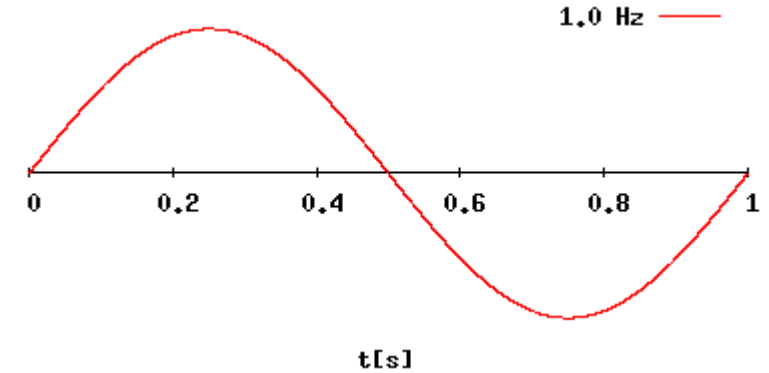
- Un **computador** es un dispositivo electrónico que almacena y procesa datos.
- Un computador incluye tanto *hardware* como *software*. En general, el **hardware** comprende los elementos físicos visibles de la computadora, y el **software** proporciona las instrucciones invisibles que controlan el hardware y lo hacen realizar tareas específicas.



# Hardware

**En general un computador un computador contiene los siguientes componentes de hardware:**

- Una unidad central de procesamiento (CPU)
- Memoria
- Dispositivos de almacenamiento (como discos y CD)
- Dispositivos de entrada (como el mouse y el teclado)
- Dispositivos de salida (como monitores e impresoras)
- Dispositivos de comunicación (como módems y tarjetas de red).





# Memoria

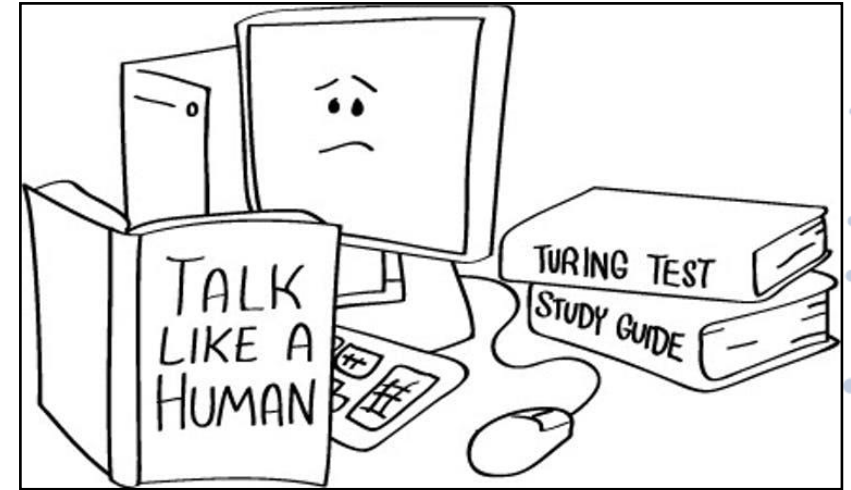
- La memoria de una computadora consiste en una secuencia ordenada de bytes para almacenar programas, así como los datos con los que está trabajando el programa.
- Puede pensar en la memoria como el área de trabajo de la computadora para ejecutar un programa.
- Un programa y sus datos deben trasladarse a la memoria de la computadora antes de que la CPU pueda ejecutarlos.

Memory address	Memory content	
.	.	
.	.	
.	.	
2000	01000011	Encoding for character 'C'
2001	01110010	Encoding for character 'r'
2002	01100101	Encoding for character 'e'
2003	01110111	Encoding for character 'w'
2004	00000011	Decimal number 3
.	.	

# Lenguajes de programación

¿Qué son los lenguajes de programación?  
¿Por qué son necesarios?

- Las computadoras no entienden los idiomas humanos, por lo cual, los programas deben estar escritos en un idioma que una computadora pueda usar. Hay cientos de lenguajes de programación, y fueron desarrollados para facilitar el proceso de programación para las personas. Sin embargo, todos los programas al final deben convertirse en instrucciones que los computadores puedan ejecutar.







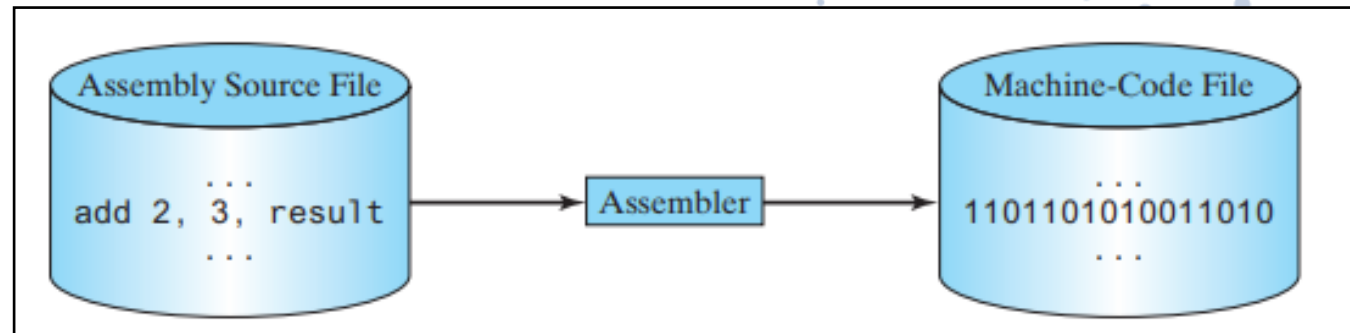
# Tipos de lenguaje - I

## Lenguaje de maquina

Es el idioma nativo de un computador. El cual esta hecho en forma de código binario.

## Lenguaje de ensamblador

Programar a punta de unos y ceros es muy tedioso, por lo tanto a inicios de la era de la computación, se inventó el lenguaje de ensamblador. El lenguaje ensamblador utiliza palabras descriptivas cortas para representar un conjunto de instrucciones de maquina



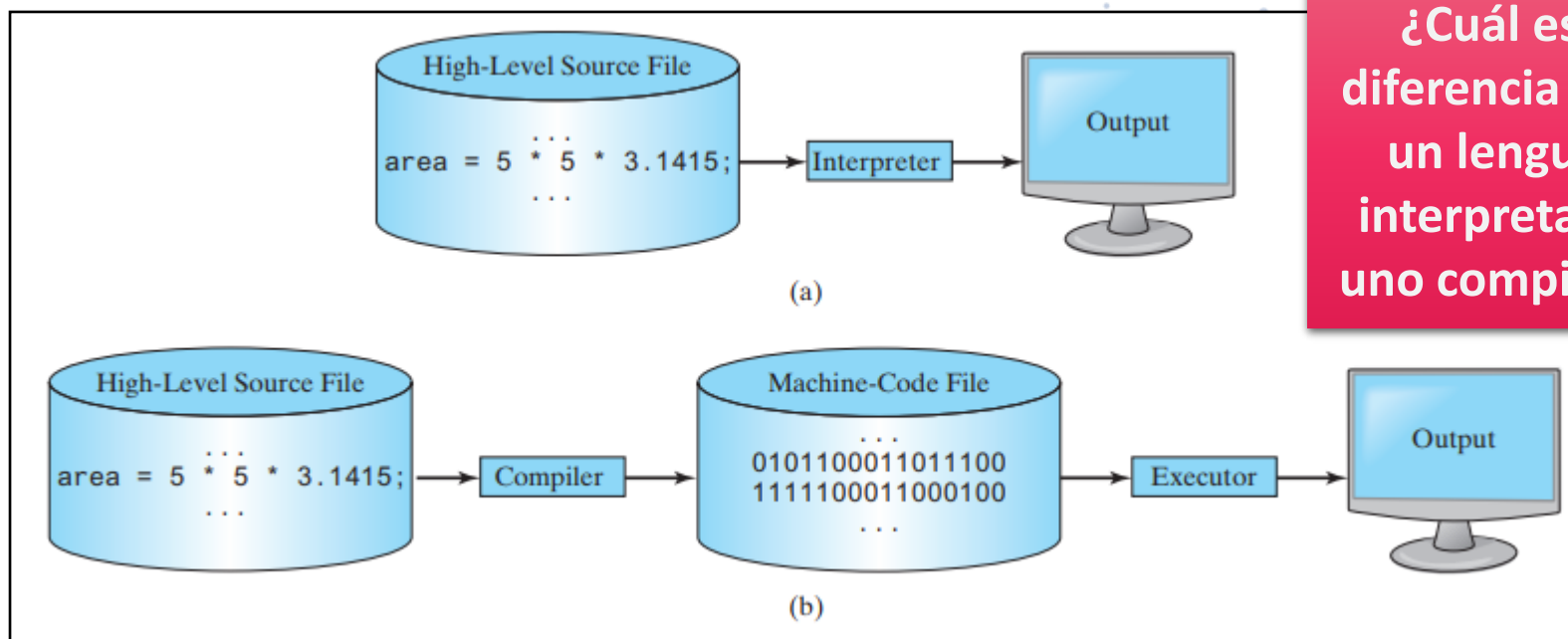




# Tipos de lenguaje - II

## Lenguaje de alto nivel

En la década de 1950, surgió una nueva generación de lenguajes de programación conocidos como “lenguajes de alto nivel”. Estos lenguajes son independientes de la plataforma, lo que significa que se puede escribir un programa en un lenguaje de alto nivel y ejecutarlo en diferentes tipos de máquinas. Los lenguajes de alto nivel son similares al idioma “Inglés”, y son más fáciles de aprender y de usar.



¿Cuál es la  
diferencia entre  
un lenguaje  
interpretado y  
uno compilado?



# Metodología de desarrollo de aplicaciones

## Pasos en la construcción de una aplicación por computador

1. Análisis del problema
2. Diseño de la solución
3. Implementación de la solución diseñada
  - 3.1 Elaboración de algoritmos
    - 3.1.1 Análisis
    - 3.1.2 Diseño
    - 3.1.3 Construcción del algoritmo
    - 3.1.4 Pruebas de escritorio
  - 3.2 Codificación en algún lenguaje de programación
  - 3.3 Compilación
  - 3.4 Pruebas sistematizadas
4. Pruebas con el usuario
5. Puesta en marcha



# Algoritmos

## ¿Qué es un algoritmo?

- En matemáticas y ciencias de la computación, un algoritmo es una secuencia finita de instrucciones, las cuales están bien definidas y se pueden implementar en una computadora. Típicamente, estas instrucciones se utilizan para resolver un tipo de problema o realizar un cálculo.

# Ejemplo algoritmo

**Objetivo:** calcular el promedio de edad de los trabajadores de una oficina.

## Secuencia de pasos:

- **Paso 1:** Recoger la edad del integrante 1
- **Paso 2:** Recoger la edad del integrante 2
- **Paso 3:** Recoger la edad del integrante 3
- **Paso 4:** Recoger la edad del integrante 4
- **Paso 5:** Recoger la edad del integrante 5
- **Paso 6:** Recoger la edad del integrante 6
- **Paso 7:** Sumar la edad de los integrantes 1, 2, 3, 4, 5 y 6.
- **Paso 8:** Dividir el total de la suma anterior entre 6.



12

18

23

40

20

23

$$12 + 18 + 23 + 40 + 20 + 23 = 136$$

$$136 / 6 = 22.66$$



**1. Análisis del  
problema**

**2. Diseño de  
la solución**

**3.  
Construcción  
del algoritmo**

**3. Validación  
de la  
solución/  
Prueba de  
escritorio**

**4.  
Codificación  
en un  
lenguaje de  
programación**



# 1. Análisis para diseñar un algoritmo

**Realizar las siguientes preguntas:**

- ¿Cuál es el objetivo buscado?
- ¿Cuáles son los datos de entrada?
- ¿Qué cálculos/procesos deben llevarse a cabo?
- ¿Cuáles son los datos de salida?



## 2. Diseño de la solución

- **Objetivo:** calcular el promedio de edad de los trabajadores de una oficina.
- **Datos de entrada:** edad integrante 1, edad integrante 2, edad integrante 3, edad integrante 4, edad integrante 5, y edad integrante 6.
- **Procesos / cálculos:** sumar las edades de los integrantes 1, 2, 3, 4, 5, y 6. Y dividir el total por 6.
- **Datos de salida:** promedio de edad de los 6 integrantes.

## 3. Construcción del algoritmo

**Pseudocódigo**  
(P-código)

inicio

acción 1

acción 2

acción n

...

fin





## 3. Validación de la solución

- Para validar la solución de un algoritmo, se pueden realizar pruebas de escritorio (las cuales no requieren el uso de computadores). O también se puedan realizar pruebas en computadores, para lo cual se necesita tener codificado el algoritmo en un lenguaje de programación.

### Validación manual (prueba de escritorio):

- Determinar los valores finales de las variables suponiendo que se tienen las siguientes instrucciones:

- $i = 9$
- $j = 4$
- $i = i + j$
- $i = j - 2$

Línea	i	J
1	9	
2	9	4
3	13	4
4	2	4



## 4. Codificación en un lenguaje de programación

```
main.py  saved
1  edad1 = 12
2  edad2 = 18
3  edad3 = 23
4  edad4 = 40
5  edad5 = 20
6  edad6 = 23
7
8  suma = edad1+edad2+edad3+edad4+
9  promedio = suma/6
10
11 print(promedio)
```

```
public class Welcome {
    public static void main(String[] args) {
        // Display message Welcome to Java! on the console
        System.out.println("Welcome to Java!");
    }
}
```

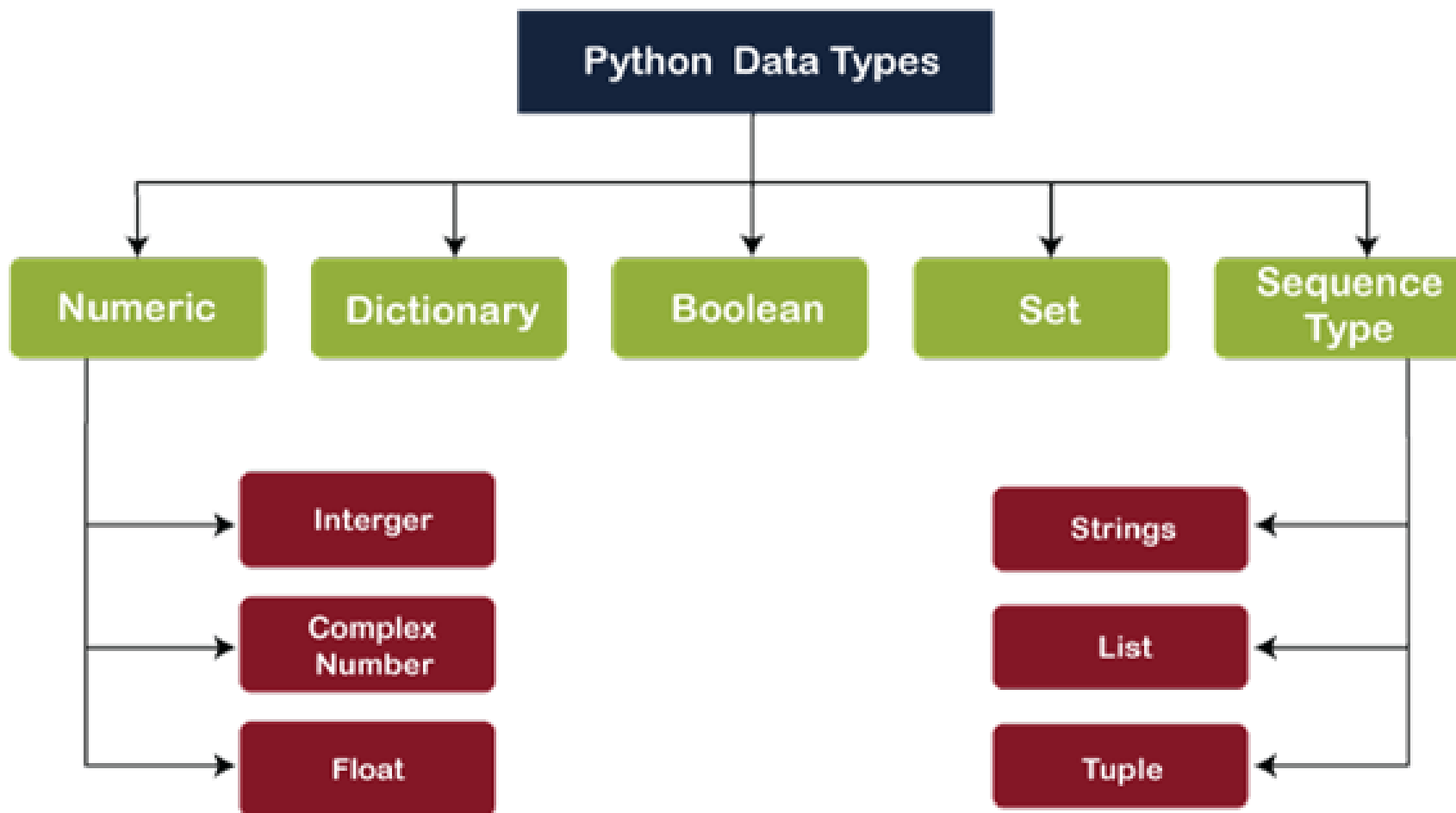
```
https://Holamundo.o
22.666666666666666
>
```



# Algoritmo, Datos, Constantes y Variables

## Datos

- Los datos son la información con la que trabaja un algoritmo.
- Pueden ser numéricos o no numéricos.
- Los datos numéricos pueden ser:
  - Enteros (int)
  - Reales (tienen parte decimal) (float)
- Los datos no numéricos son de tipo string (hilera o cadena) (str)
- Datos lógicos: pueden ser verdadero (True) o falso (False) (bool)





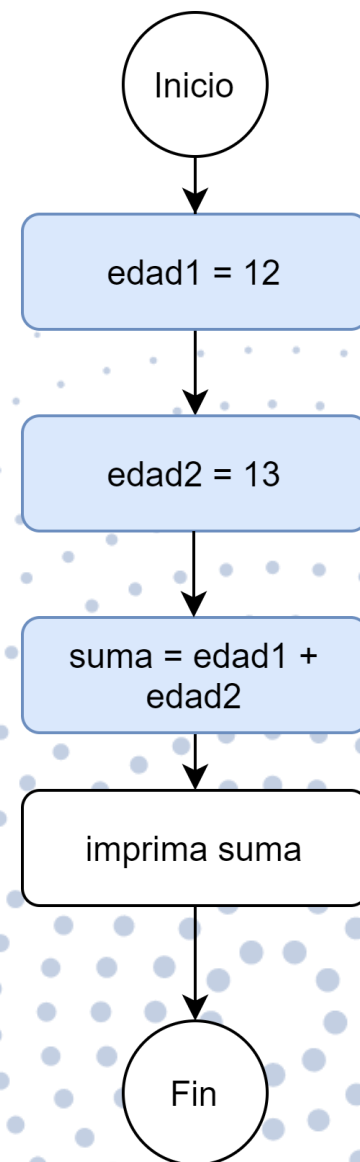
## Estructura de un algoritmo

- Entrada de datos
- Procesamiento de datos
- Salida de resultados

## Estructuras para la construcción de un algoritmo

- Estructura secuencia
- Estructura decisión
- Estructura ciclo

```
main.py  saved
1  edad1 = 12
2  edad2 = 18
3  edad3 = 23
4  edad4 = 40
5  edad5 = 20
6  edad6 = 23
7
8  suma = edad1+edad2+edad3+edad4+edad5+edad6
9  promedio = suma/6
10
11 print(promedio)
```





El futuro digital  
es de todos

MinTIC



UNIVERSIDAD  
DE ANTIOQUIA  
Facultad de Ingeniería





# Python

- Es un lenguaje de programación de alto nivel.
- Es interpretado.
- Creado a finales de los 80  
inicios de los 90.

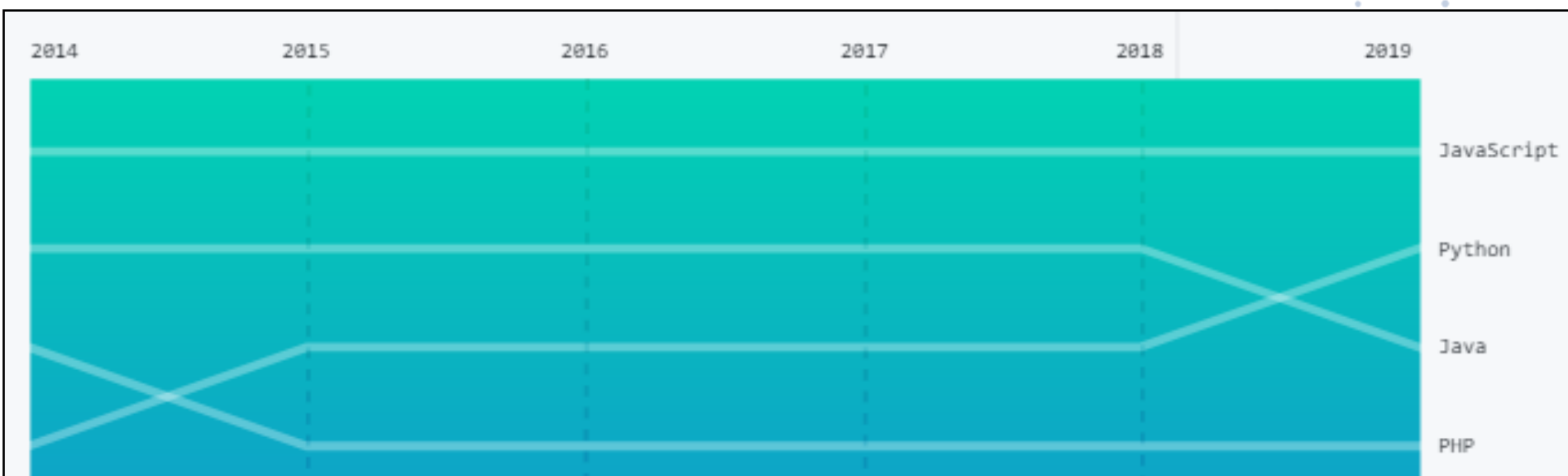






# Top lenguajes

Top lenguajes de programación mas populares  
según registro de proyectos en GitHub





# Tutorial de manejo e instalación Python

Integrated development environment (**IDE**)

<https://www.spyder-ide.org/>

Entorno digital utilizado para desarrollar software, juegos o cualquier cosa relacionada con la codificación. Un IDE ofrece integración desde los pasos más básicos del desarrollo de software, como escribir su código, depurar o incluso compilar sus aplicaciones en un lenguaje que las computadoras puedan entender.



<https://replit.com/>



<https://www.jetbrains.com/pycharm/>





El futuro digital  
es de todos

MinTIC



UNIVERSIDAD  
DE ANTIOQUIA

Facultad de Ingeniería

‘Mision  
<TIC2022>’



repl.it



El futuro digital  
es de todos

MinTIC



UNIVERSIDAD  
DE ANTIOQUIA




Facultad de Ingeniería

Mision  
<TIC2022>

Repl.it - Log In

repl.it/login

repl.it Log In




  


email or username


password


[Forgot password?](#) [or sign up](#)


Log in

  @LuisaR... 

 Home


 **My Repls**

 Talk

 Notifications

Search your repls

create a repl

 Python >

all languages



El futuro digital  
es de todos

MinTIC



UNIVERSIDAD  
DE ANTIOQUIA

Facultad de Ingeniería

Mision  
<TIC2022>



@LuisaR... ▾



Search your repls



upgrade



Home



My Repls



Talk



Notifications

(home)/



new folder



Multiplayer repls

New Folder

MisionTIC2022-Ciclo1

Cancel

Create

@LuisaRestrepo1

Search and run commands

Ctrl .

+

+ New repl

Upgrade

Apps BETA

Home

My repls

Talk

Notifications

Languages

Templates

Tutorials

(home)/MisionTIC2022-Ciclo1/

(up 1 level)

You don't seem to have anything in here. Let's change that!

New folder

+ New repl

Create new repl

Import from GitHub

Python

HolaMundo

Your repls will appear on your profile

Upgrade your account for private repls

Public

Private

Cancel

Create repl



El futuro digital  
es de todos

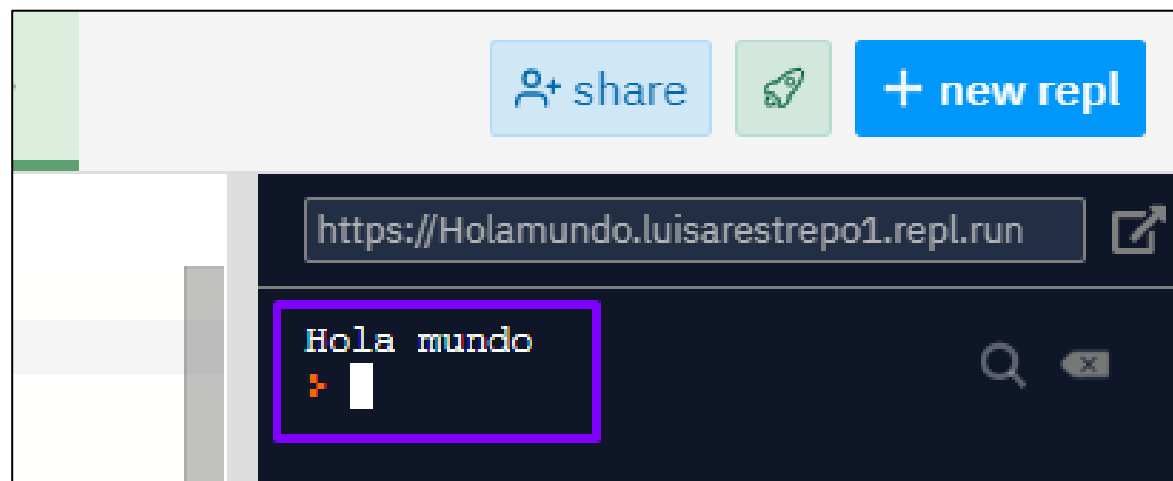
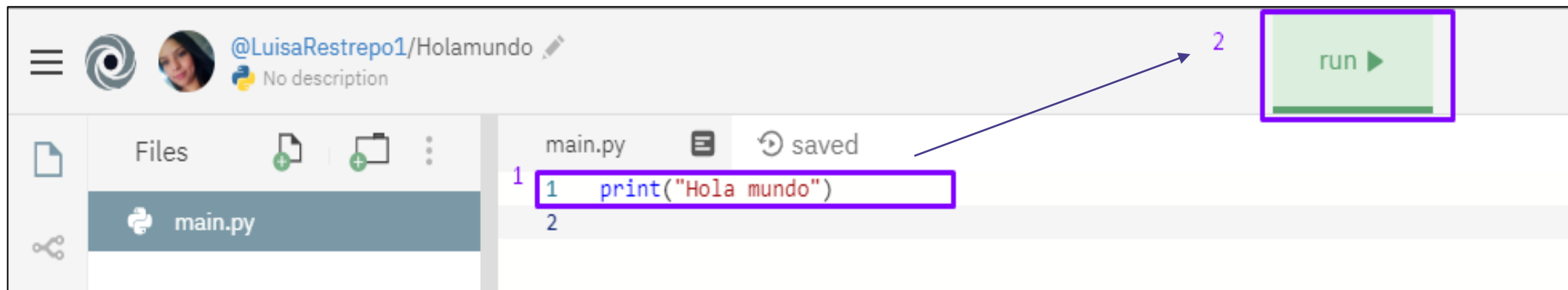
MinTIC



UNIVERSIDAD  
DE ANTIOQUIA

Facultad de Ingeniería

Mision  
<TIC2022>



```
public class Welcome {  
    public static void main(String[] args) {  
        // Display message Welcome to Java! on the console  
        System.out.println("Welcome to Java!");  
    }  
}
```





El futuro digital  
es de todos

MinTIC



UNIVERSIDAD  
DE ANTIOQUIA

Facultad de Ingeniería

‘Mision  
<TIC2022>’



@LuisaRestrepo1



+ New repl

Upgrade



Apps

BETA



Home



My repls



Talk



Notifications



Languages

> Search and run commands

Ctrl



(home)/MisionTIC2022-Ciclo1/

New folder



(up 1 level)



HolaMundo

Python

6 hours a... 3 KB





El futuro digital  
es de todos

MinTIC



UNIVERSIDAD  
DE ANTIOQUIA

Facultad de Ingeniería

‘Mision  
<TIC2022>’

CO  
the collaboratory



El futuro digital  
es de todos

MinTIC

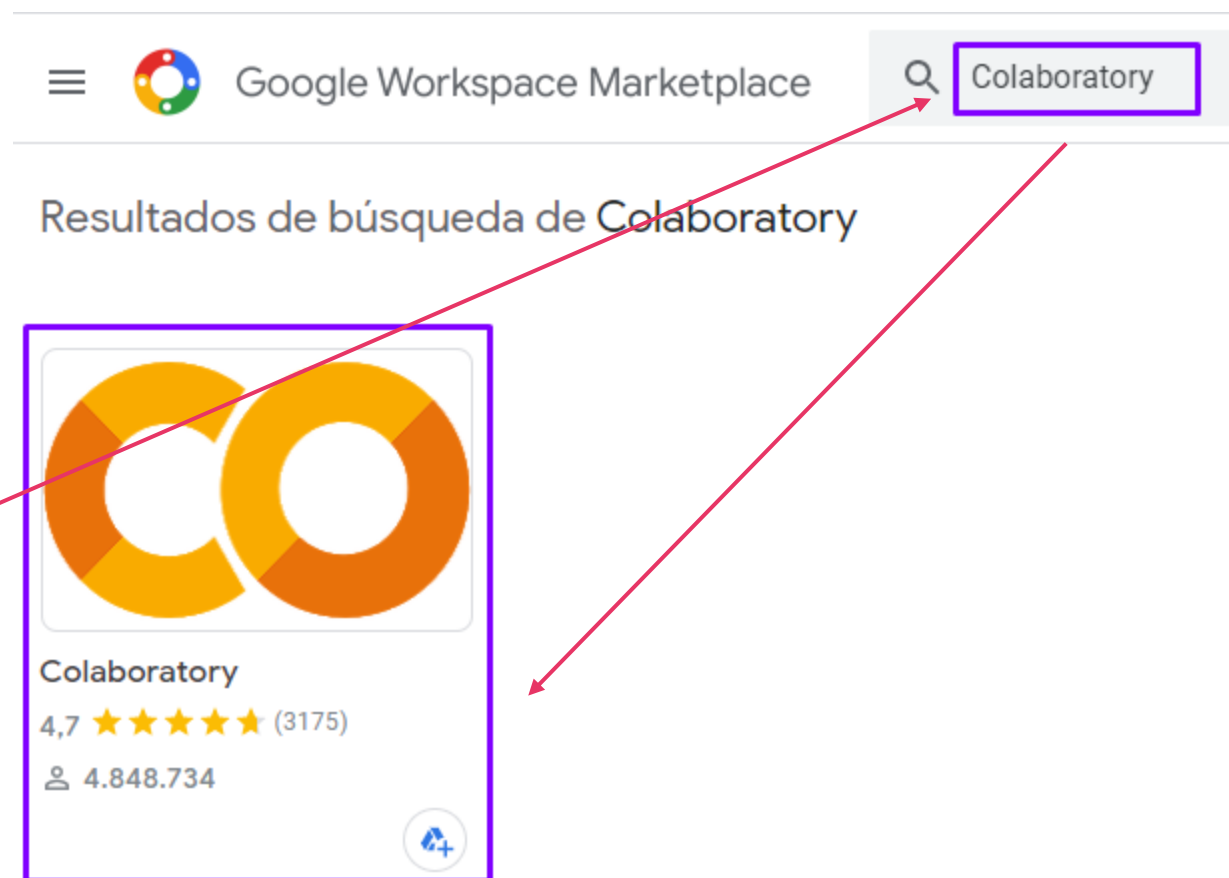
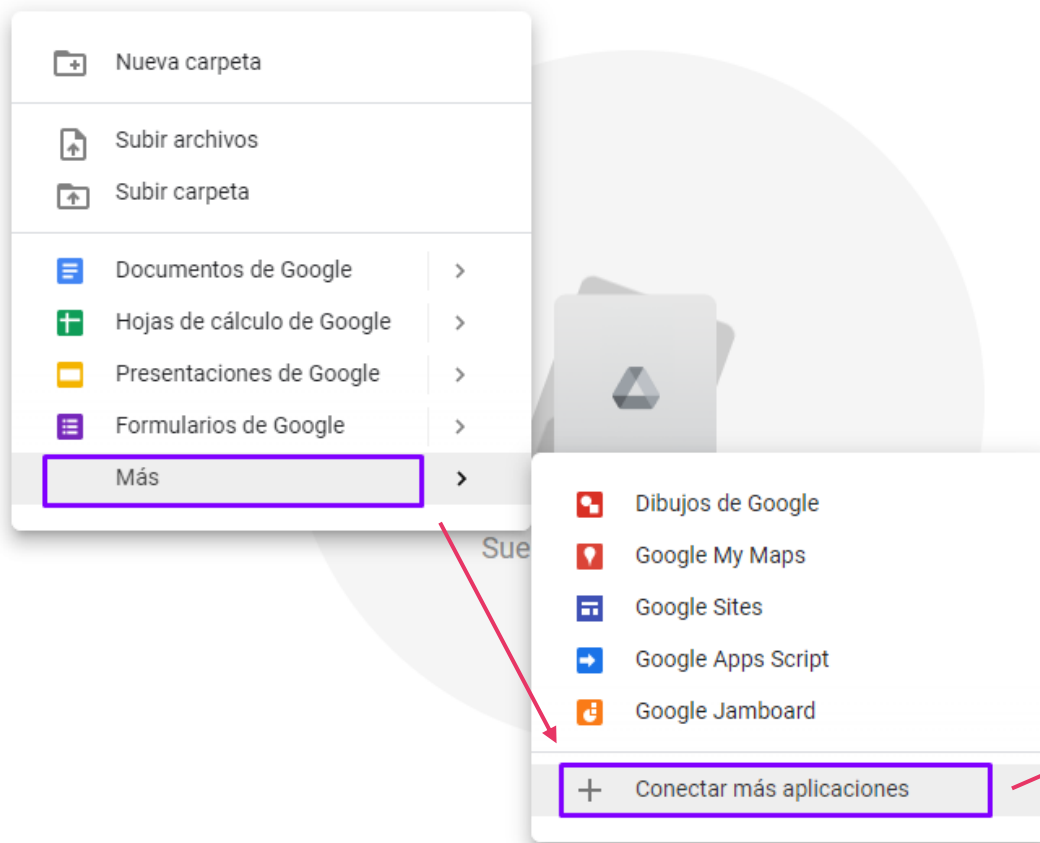


UNIVERSIDAD  
DE ANTIOQUIA

Facultad de Ingeniería



## Abrir Drive de Google





El futuro digital  
es de todos

MinTIC



UNIVERSIDAD  
DE ANTIOQUIA  
Facultad de Ingeniería

Mision  
<TIC2022>

← Google Workspace Marketplace

🔍 Buscar en apps



Colaboratory  
colab-team

★★★★★ (3175) · 👤 4.848.734

🔗 Complemento de Drive

Instalar

🔗 Prepárate para la instalación

Para iniciar la instalación de **Colaboratory**, antes debes dar permiso.

Al hacer clic en Continuar, aceptas que tu información se utilice de conformidad con las [condiciones de servicio](#) y la [política de privacidad](#) de esta aplicación.



Welcome To Colaboratory

File Edit View Insert Runtime Tools Help



+ Code + Text 📁 Copy to Drive

🔗 Share ⚙️

Sign in

Connect ▾

Editing



🔗 What is Colaboratory?

Colaboratory, or "Colab" for short, allows you to write and execute Python in your t

- Zero configuration required
- Free access to GPUs
- Easy sharing



Google Colaboratory se ha conectado a Google Drive.

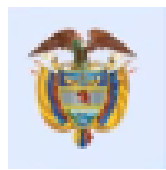


Convierte a **Google Colaboratory** en la aplicación predeterminada de los archivos que puede abrir.

CANCELAR

CONTINUAR

ACEPTAR



El futuro digital  
es de todos

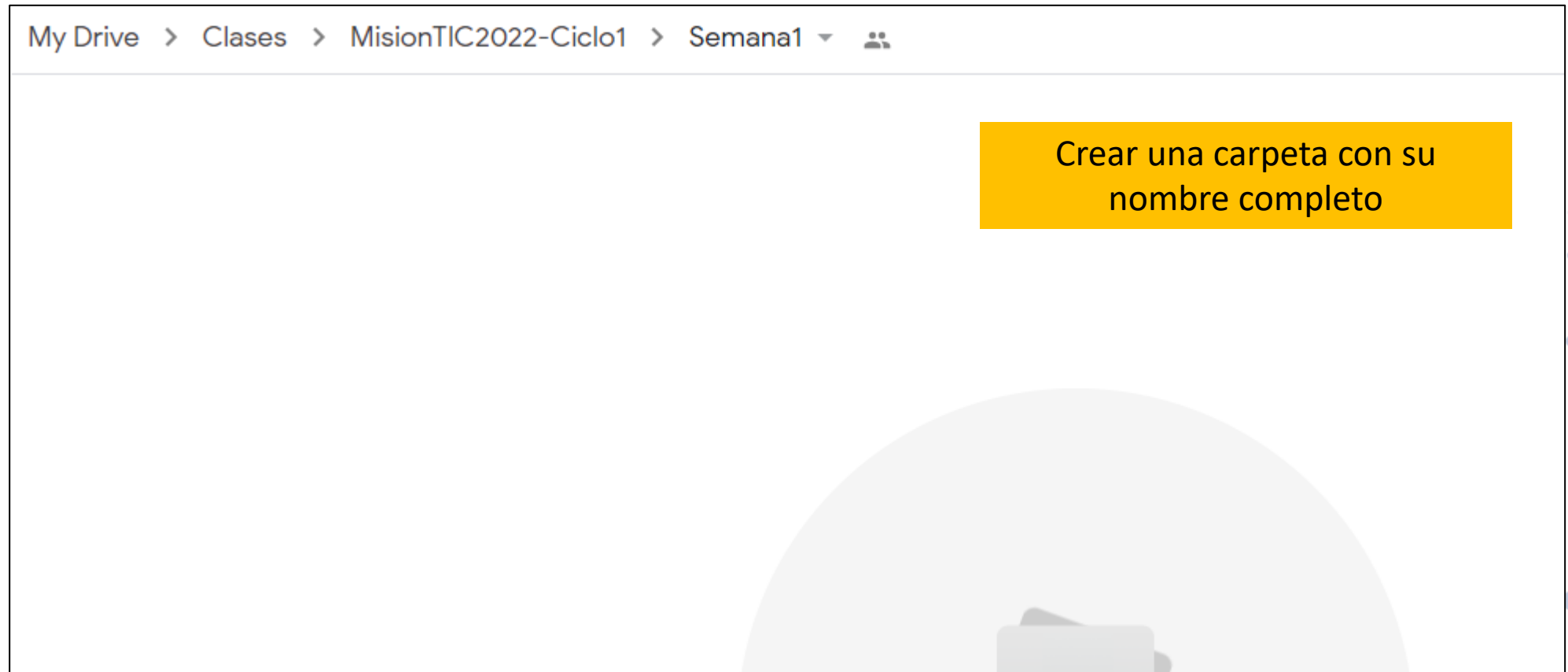
MinTIC



UNIVERSIDAD  
DE ANTIOQUIA  
Facultad de Ingeniería

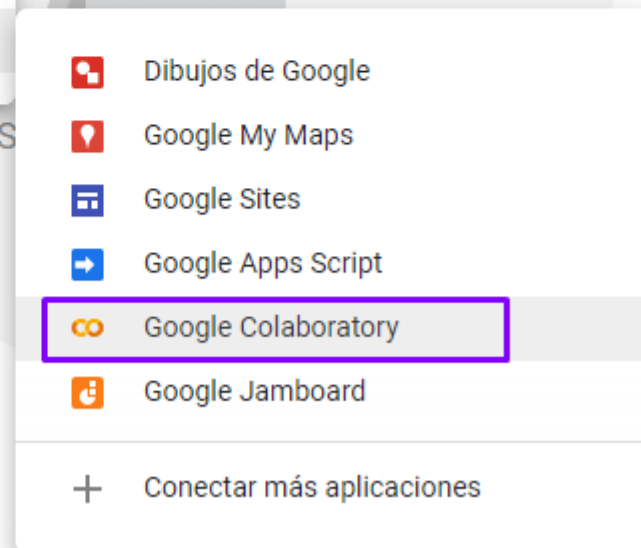
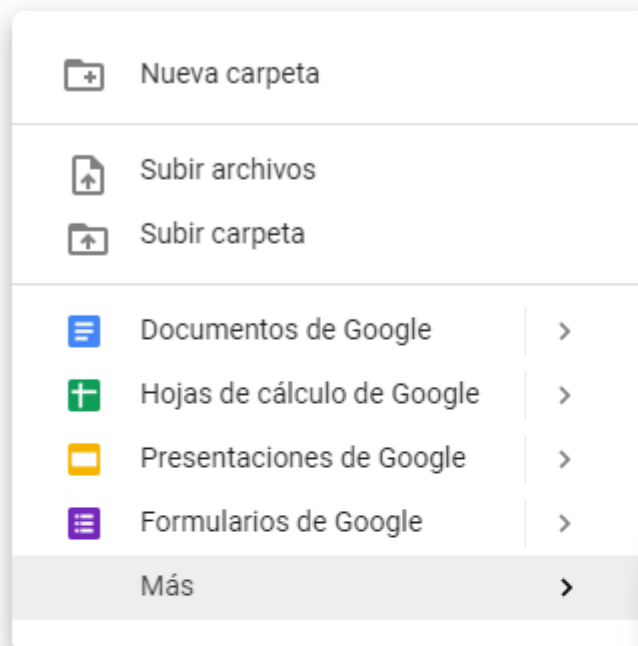


Abrir el enlace dado: <https://bit.ly/3htvj5J>



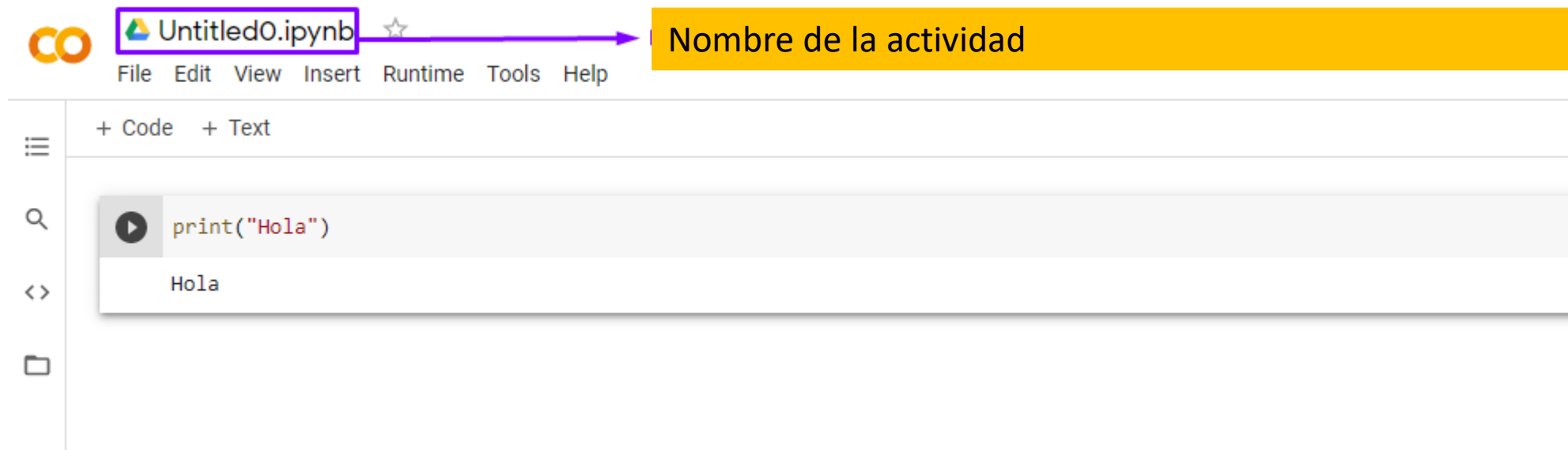


Dentro de su carpeta, crear un  
archivo colaboratory:





Cambiar el nombre del archivo:







El futuro digital  
es de todos

MinTIC



UNIVERSIDAD  
DE ANTIOQUIA

Facultad de Ingeniería

‘Mision  
<TIC2022>’



GustavoVelez.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

Locate in Drive

Open in playground mode

New notebook

Open notebook Ctrl+O

Upload notebook

Rename

Move to trash

Save a copy in Drive

Save a copy as a GitHub Gist

Save a copy in GitHub

Save Ctrl+S

Save and pin revision Ctrl+M S



El futuro digital  
es de todos

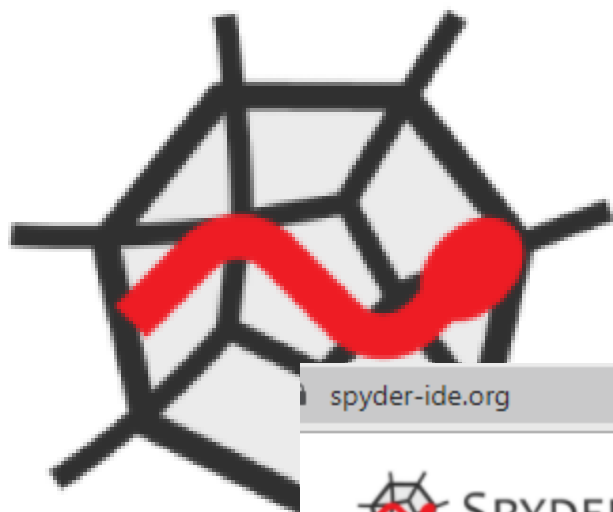
MinTIC



UNIVERSIDAD  
DE ANTIOQUIA

Facultad de Ingeniería

‘Mision  
<TIC2022>’



spyder-ide.org



SPYDER

HOME

OVERVIEW

COMPONENTS

PLUGINS

DOWNLOAD

DO

another. Happy Spydering!

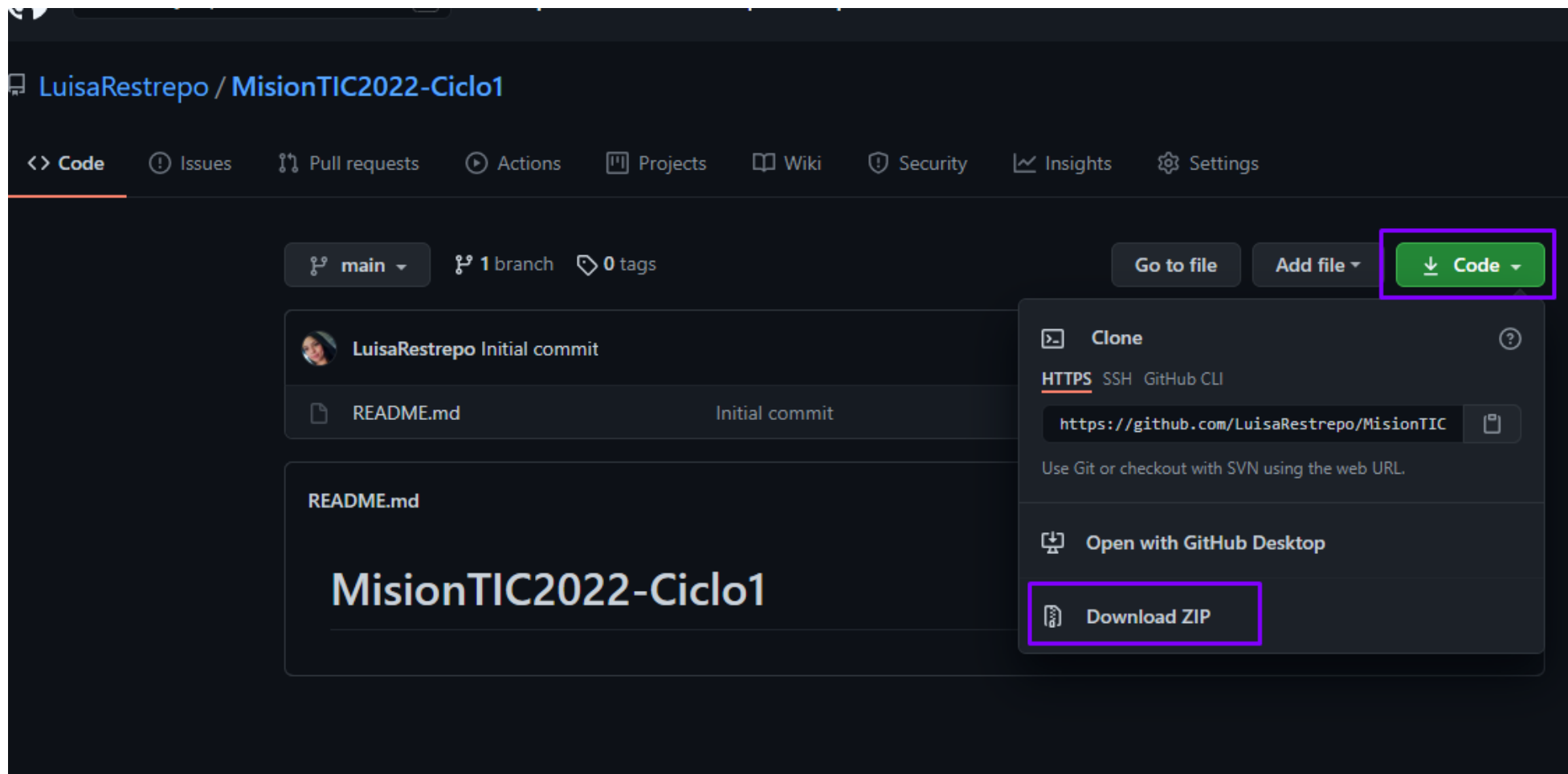
**macOS Big Sur users:** Full support for macOS 11 Big Sur will be included in Spyder 4.2.1, scheduled for release on December 18, 2020. However, see [our FAQ question on Big Sur](#) for how to get it working right now.



Download for Windows

Descargue el contenido de clase:

<https://github.com/LuisaRestrepo/MisionTIC2022-Ciclo1>



LuisaRestrepo / MisionTIC2022-Ciclo1

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags

Go to file Add file

↓ Code

LuisaRestrepo Initial commit

README.md Initial commit

README.md

MisionTIC2022-Ciclo1

Clone

HTTPS SSH GitHub CLI

<https://github.com/LuisaRestrepo/MisionTIC>

Use Git or checkout with SVN using the web URL.

Open with GitHub Desktop

Download ZIP

# Referencias

<https://nbviewer.jupyter.org/>

<https://rise.readthedocs.io/en/stable/installation.html>

<https://jupyter.org/try>

<https://github.com/DiegOliveros/MISION-TIC-2020>

<https://desktop.github.com/>

<https://www.jetbrains.com/es-es/pycharm/download/#section=windows>

<https://notepad-plus-plus.org/downloads/>