



Argentina
programa
4.0



Universidad
Nacional
de San Martín

Google Colab

Instrucciones

¿Qué es Google Colab?

Google Colab, también conocido como Collaboratory, es un entorno gratuito basado en Jupyter notebook que se ejecuta en los servidores de la nube de Google.

- No requiere instalación ni configuración de Python.
- Se tiene acceso gratuito a hardware CPU, GPU y TPU.
- Se puede compartir el código de manera similar que con Google Drive.

Requisitos:

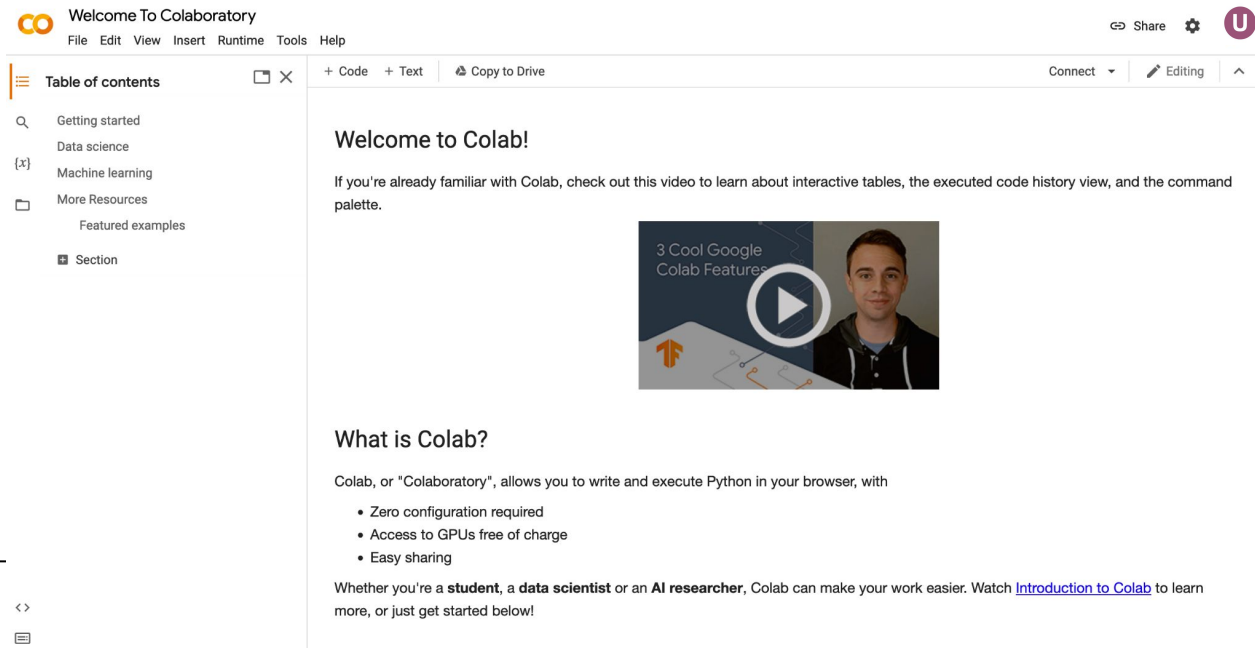
Tener una cuenta Gmail.

- Si la tenés, basta con estar conectado a la cuenta antes de abrir una nueva ventana en tu navegador para ingresar a Colab.
- Si no tenés cuenta en Gmail, lee las instrucciones de cómo abrir una cuenta Gmail en este [link](#).

¿Cómo ingresar a Google Colab?

1. En tu navegador abrí una ventana e ingresá a tu cuenta Gmail haciendo click en <https://www.gmail.com/>
2. Luego, en otra ventana de tu navegador ir a la siguiente dirección: <https://colab.research.google.com/>

De esta forma vas a poder entrar a la página de inicio de Collaboratory y en ella encontrarás un tutorial para iniciar.



co Welcome To Colaboratory

File Edit View Insert Runtime Tools Help

Share Settings U

Table of contents


- Getting started
- Data science
- {x} Machine learning
- More Resources
- Featured examples
- Section

+ Code + Text Copy to Drive

Connect Editing ^

Welcome to Colab!

If you're already familiar with Colab, check out this video to learn about interactive tables, the executed code history view, and the command palette.



What is Colab?

Colab, or "Colaboratory", allows you to write and execute Python in your browser, with

- Zero configuration required
- Access to GPUs free of charge
- Easy sharing

Whether you're a **student**, a **data scientist** or an **AI researcher**, Colab can make your work easier. Watch [Introduction to Colab](#) to learn more, or just get started below!



Argentina
programa
4.0

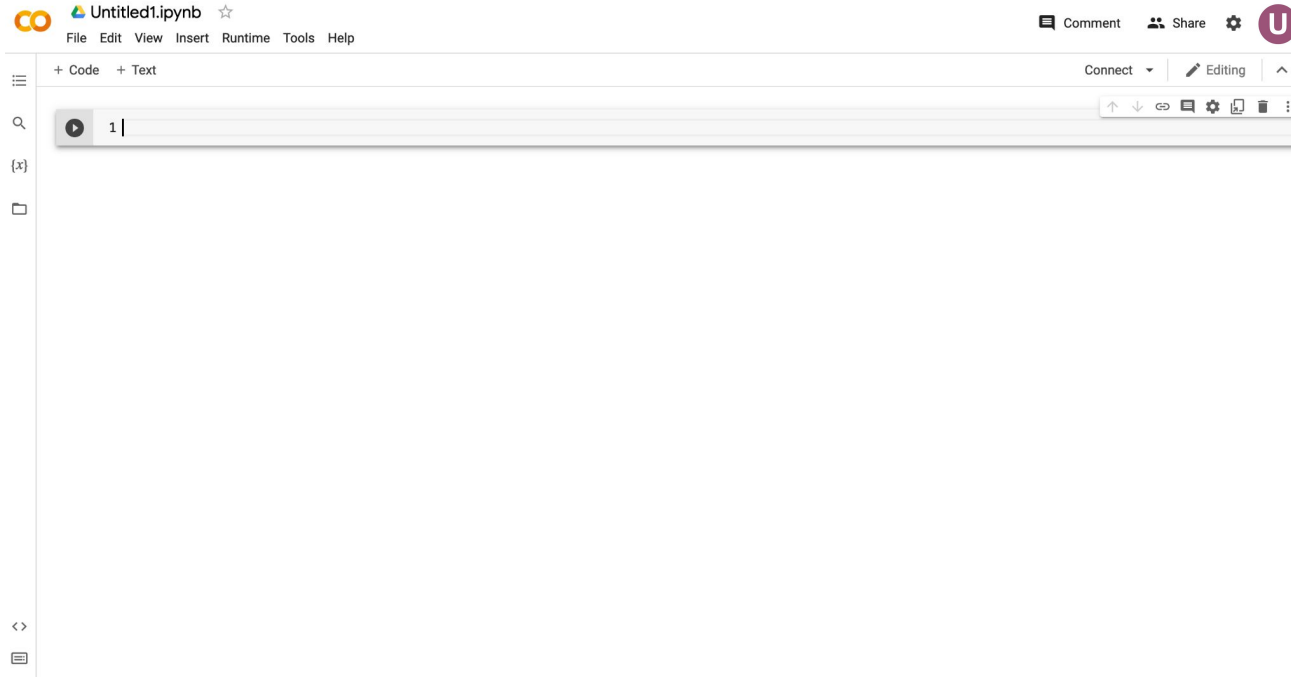


Universidad
Nacional
de San Martín



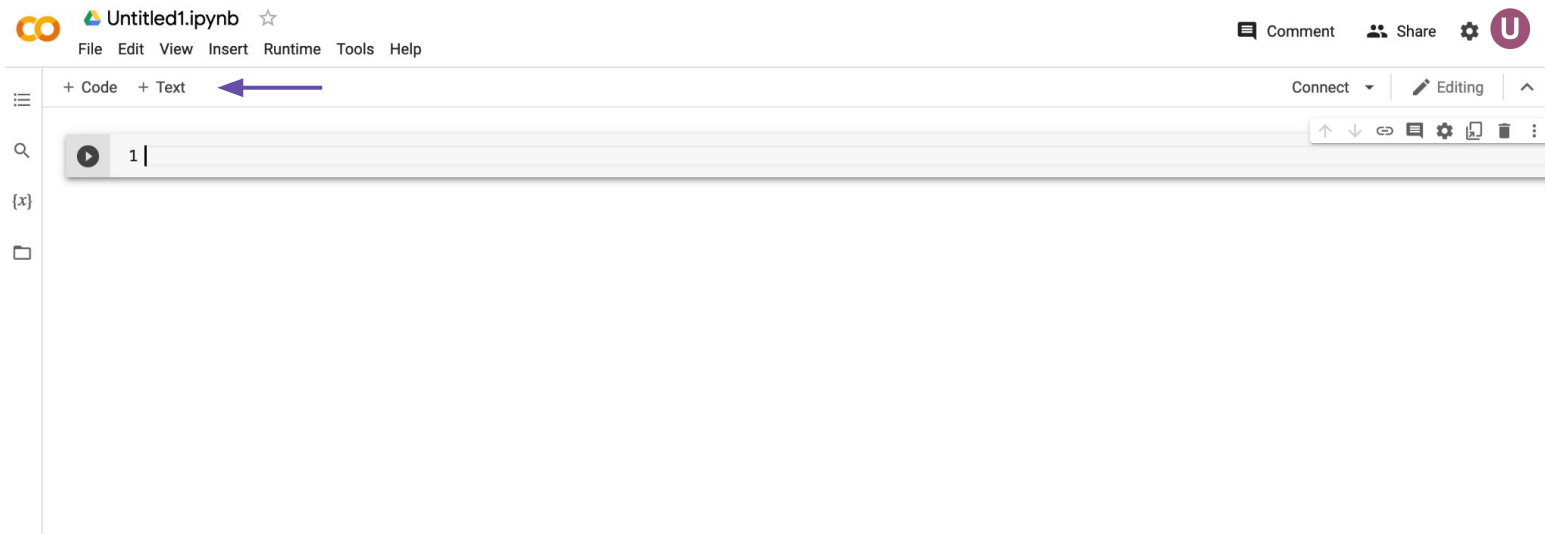
Aspectos básicos

1. Cuando se inicia un notebook obtenemos una vista como la siguiente



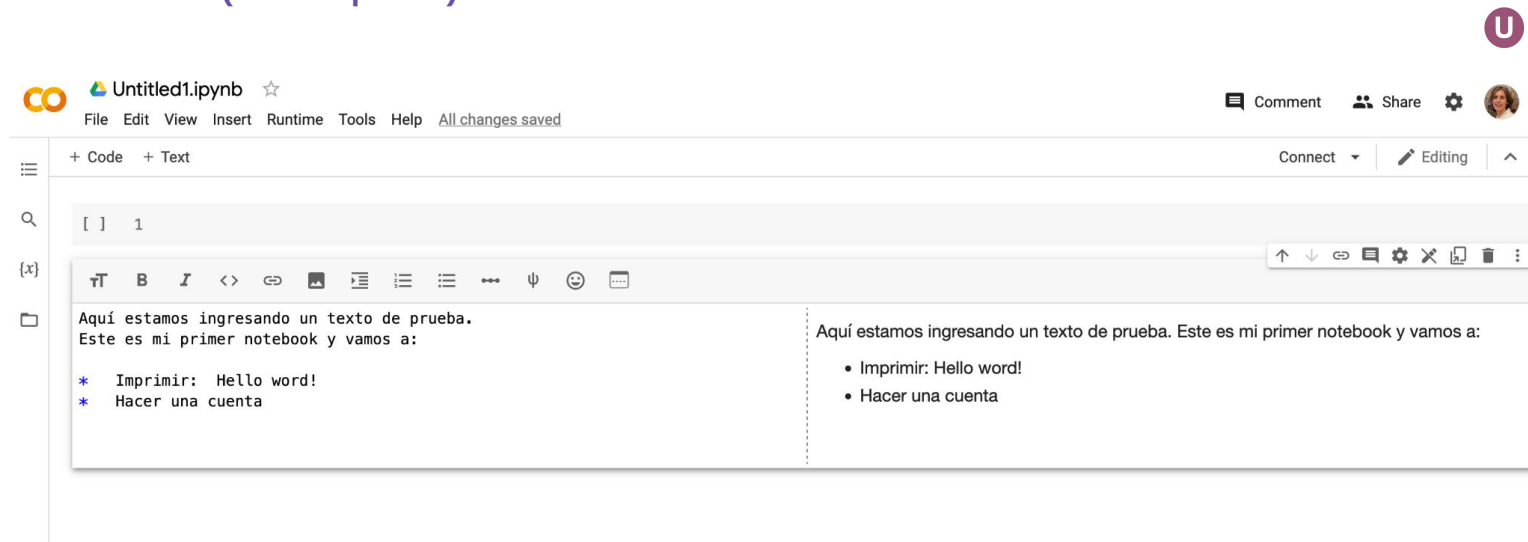
Aspectos básicos

2. En Google Colab (así como en Jupyter Notebook) existen dos tipos de celdas:
 - a. Celdas de texto: donde se puede redactar cualquier tipo de texto
 - b. Celdas de código: para la ejecución del código en el lenguaje de programación Python
3. Para agregar una celda de texto será necesario hacer click en el botón +Text (flecha roja).



Aspectos básicos

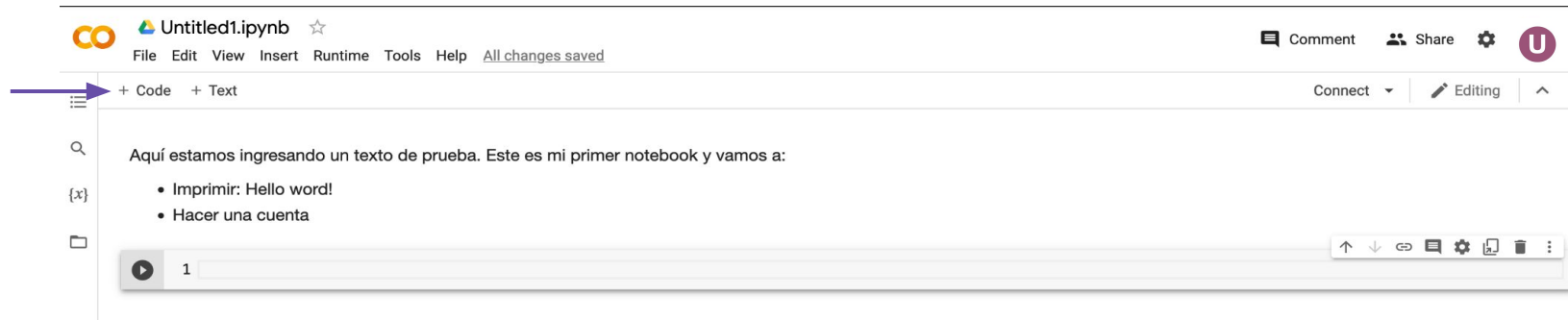
4. Entonces, basta con escribir el texto que quieras agregar. Este tipo de celdas tienen una barra de herramientas básica para modificar el texto (subrayar, colocar en negritas, enumerar, colocar viñetas, etc). A medida que edites esta celda vas a poder visualizar en el lado derecho como va a quedar el texto (la vista previa).



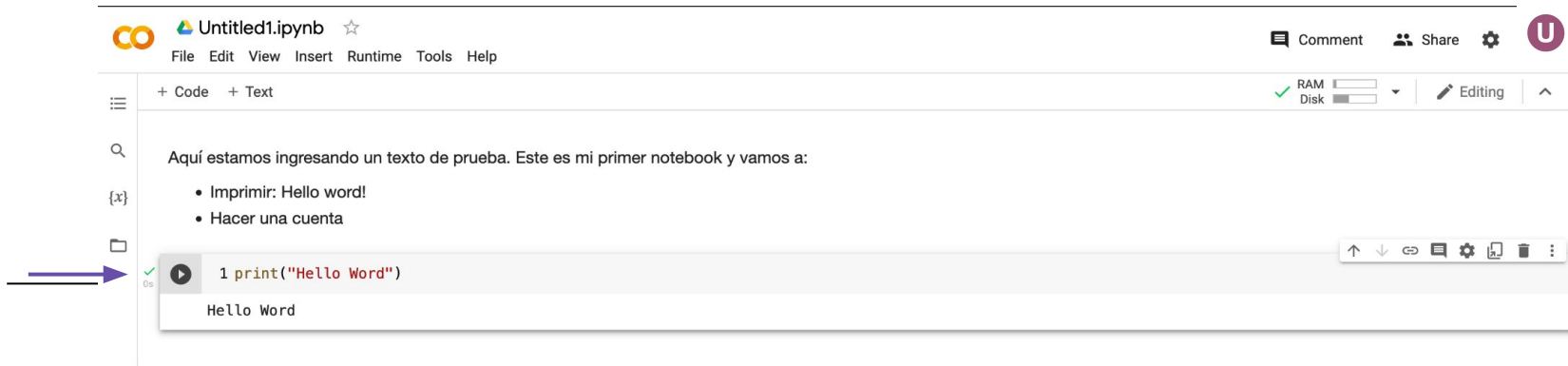
The screenshot displays the Jupyter Notebook interface. At the top, the title bar shows 'Untitled1.ipynb' with a star icon. Below it, a menu bar includes 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', 'Help', and 'All changes saved'. On the right side of the title bar, there are icons for 'Comment', 'Share', a settings gear, and a user profile icon labeled 'U'. The main workspace is divided into two sections: a left sidebar with icons for file management and search, and a central area for the notebook cells. The top of the central area shows '+ Code' and '+ Text' tabs, with 'Text' being the active tab. Below this, a toolbar contains various text formatting icons like bold (B), italic (I), underline (U), and bulleted list (•). The text cell is in edit mode, showing a preview of the text on the right. The text in the cell is: 'Aquí estamos ingresando un texto de prueba. Este es mi primer notebook y vamos a:'. Below this, there are two bulleted items: '• Imprimir: Hello word!' and '• Hacer una cuenta'.

Aspectos básicos

5. Para agregar una celda de código hacé click en el botón +Code

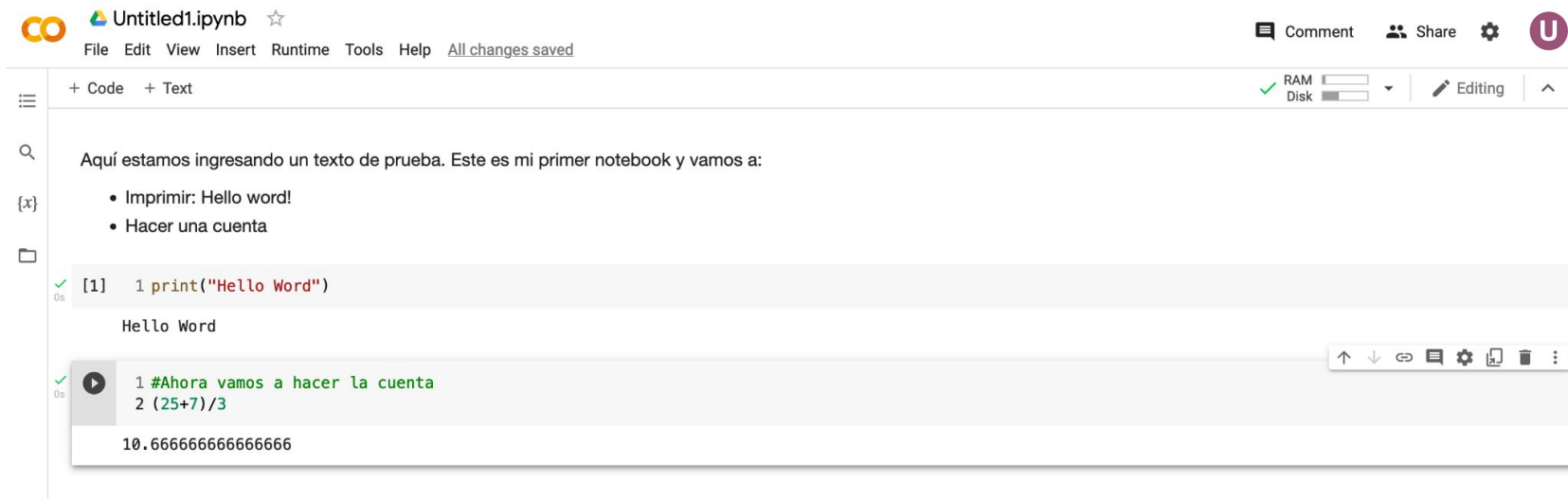


6. En la nueva celda vas a poder escribir código y ejecutarlo con el botón de “Play” (Ejecutar)



Aspectos básicos

7. Al escribir o editar código puede ser que quieras agregar comentarios o notas, pero no como una celda de texto, sino en la misma celda del código. Para esto, hay que agregar al comienzo el símbolo #



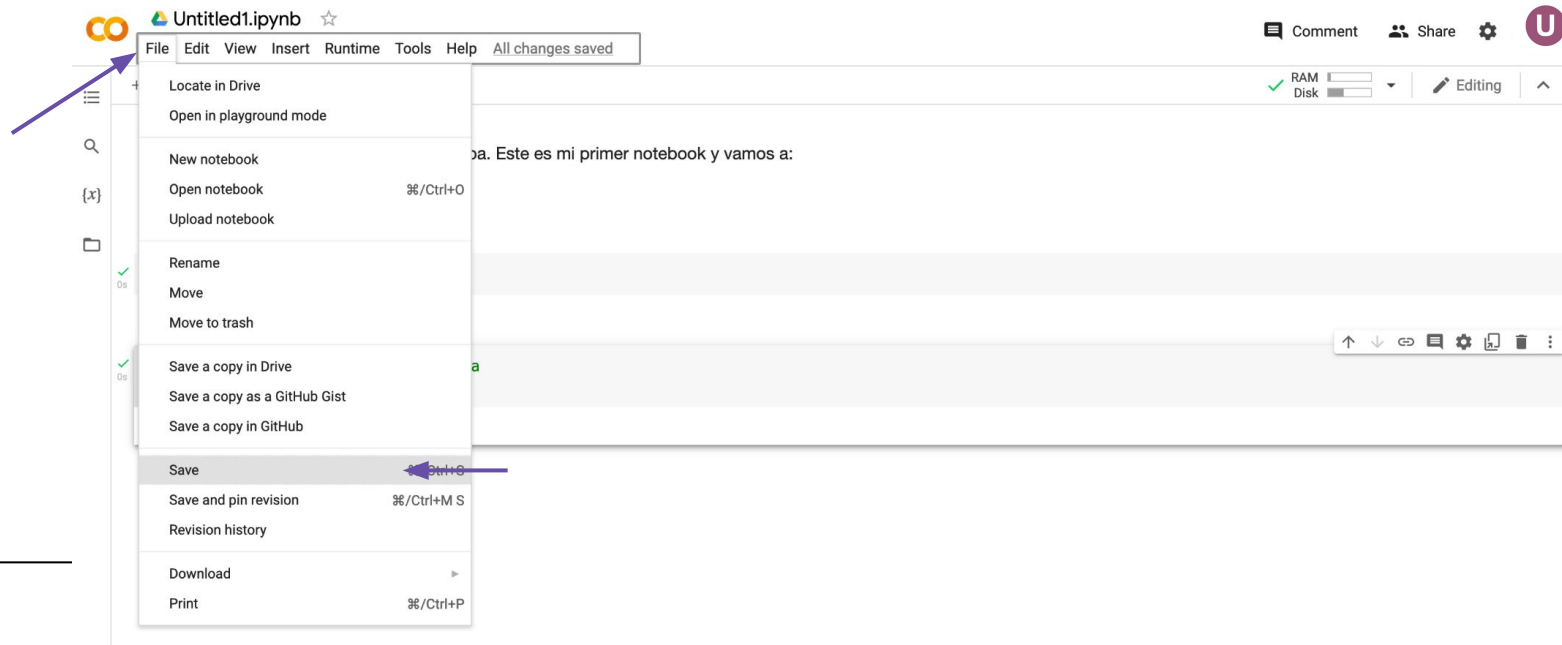
The screenshot shows a Jupyter Notebook titled 'Untitled1.ipynb'. The interface includes a top bar with 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help' menus, along with a 'Comment' button, a 'Share' button, and a user profile icon. Below the top bar, there are tabs for '+ Code' and '+ Text'. The notebook contains two code cells. The first cell has the code `1 print("Hello Word")` and outputs `Hello Word`. The second cell has the code `1 #Ahora vamos a hacer la cuenta` (commented) and `2 (25+7)/3` (code), and outputs `10.666666666666666`. The commented line is highlighted in green.

Las líneas comentadas siempre van a aparecer en verde, de esta forma serán fácilmente identificables y sabemos que no son parte del código que vamos a ejecutar.

¿Cómo guardar un notebook?

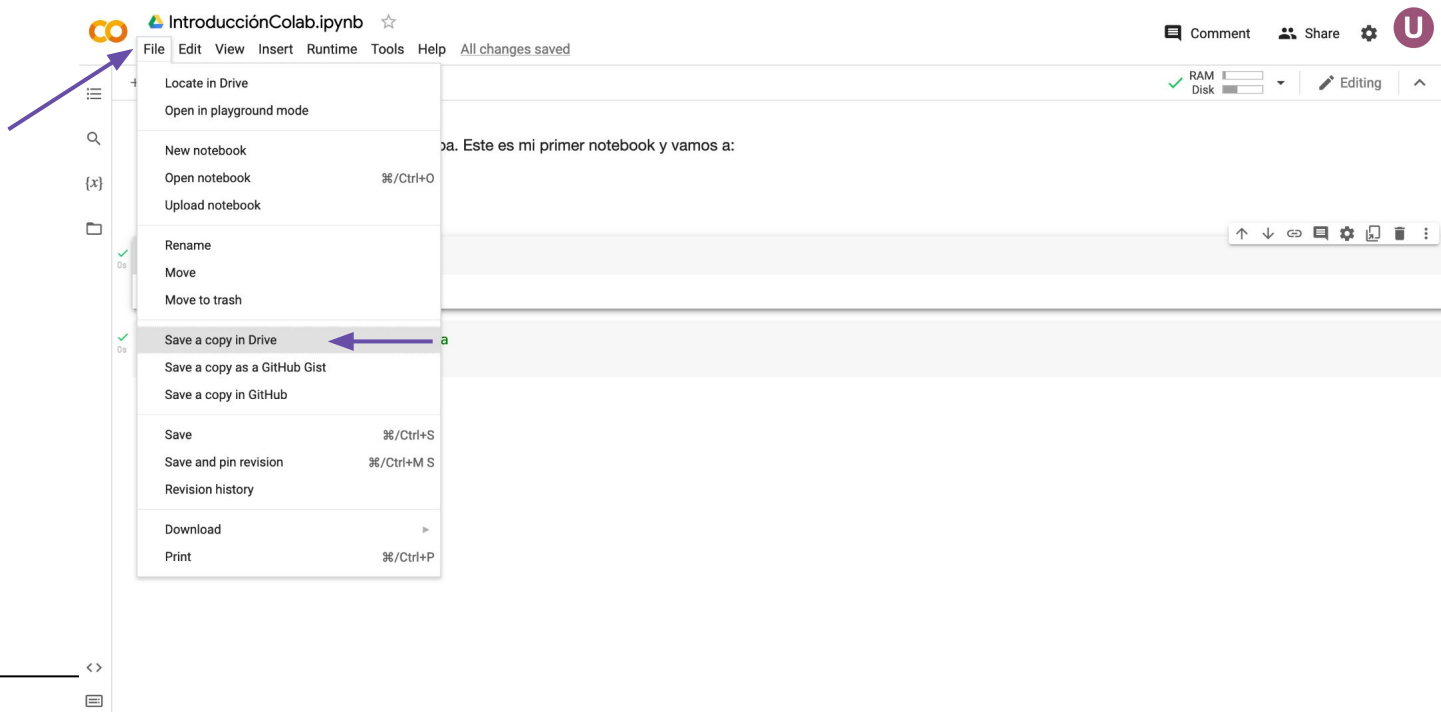
El primer paso es dar un nombre al notebook, para eso hacer doble click en el nombre del archivo arriba de todo (marcado con un rectángulo).

Google Colab cuenta con guardado automático que algunos editores tienen implementados. Pero si querés asegurarte, usá la combinación de las teclas Ctrl+S o ir al menú File y elegir la opción Save.



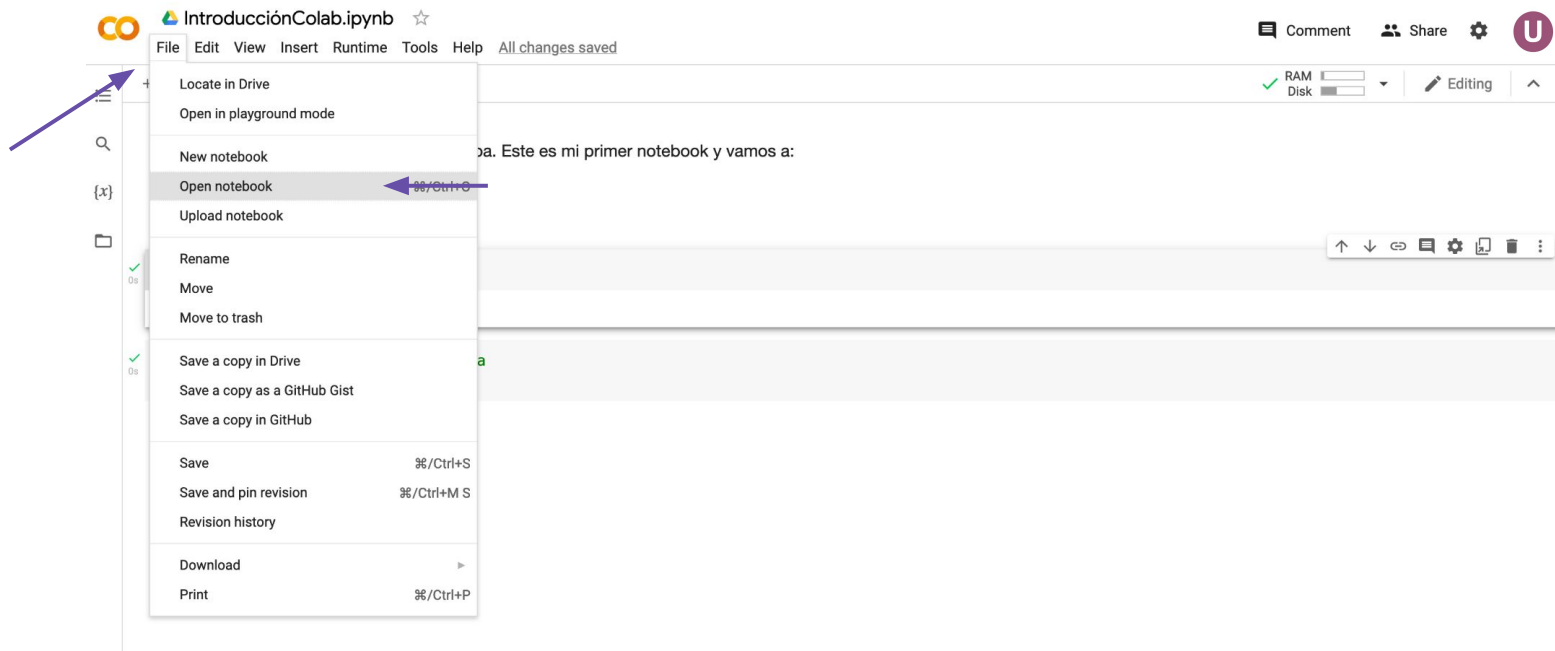
¿Cómo guardar un notebook?

También es posible guardar una copia en tu Drive haciendo click en File y eligiendo la opción Save a copy in Drive.



¿Cómo abrir un notebook?

Para abrir un notebook tenes que ir al menú principal, hacer click en File y seleccionar Open notebook, o presionar simultáneamente las teclas Ctrl-O.






¿Cómo abrir un notebook?

Una ventana con varias opciones va a permitirte que abras el notebook deseado dependiendo de donde esté el archivo: en tu drive, tu computadora, repositorio, etc, como en estos ejemplos:

ExamplesRecentGoogle DriveGitHubUpload

Filter notebooks

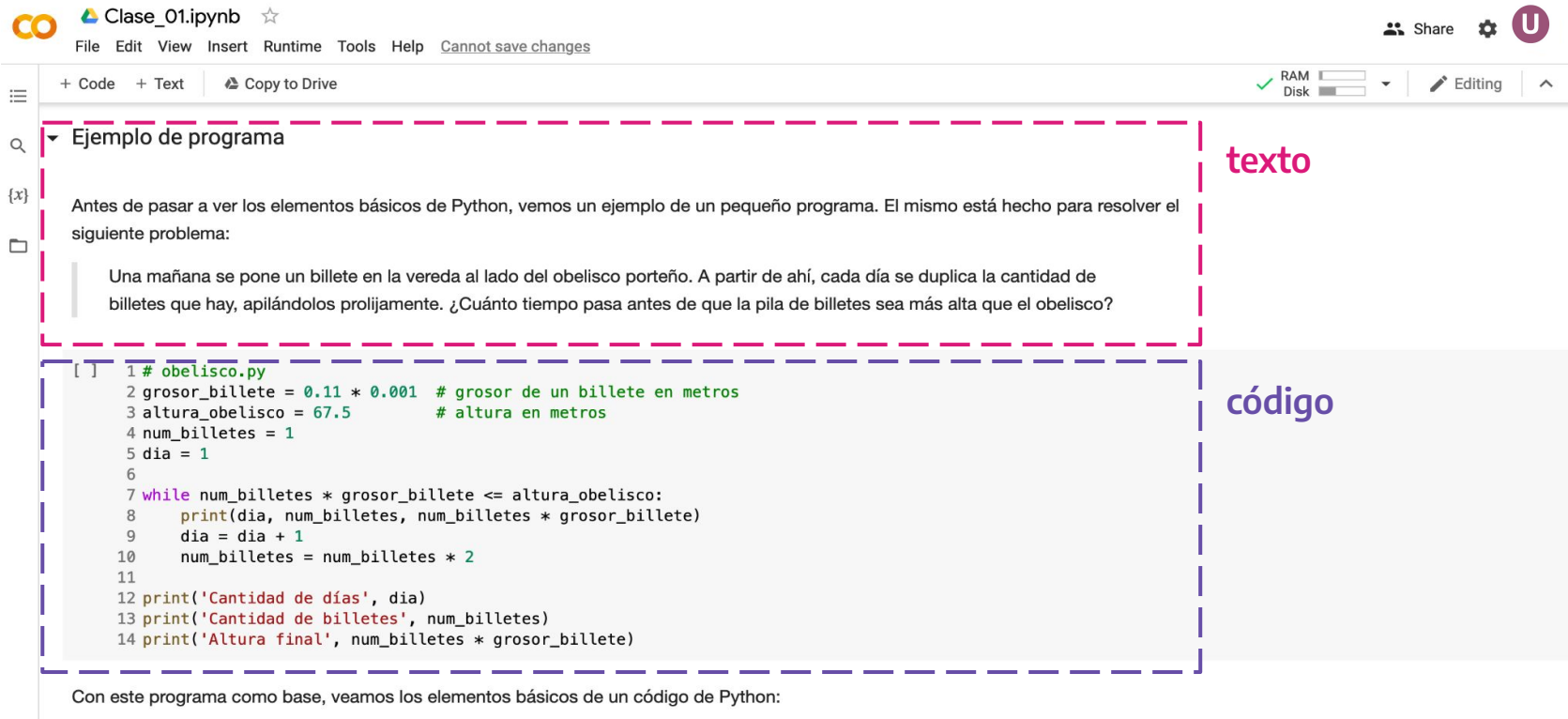
Title	Owner	Last opened ▲	Last modified ▼
 IntroducciónColab.ipynb	Carla Bonifazi	5:38 PM	5:38 PM  

ExamplesRecentGoogle DriveGitHubUpload

Browse... No file selected.

Para los notebooks que vamos a ir dando en cada clase vas a tener que descargarlos y guardarlos en tu computadora o en tu drive. Otra posibilidad para abrirlo, una vez guardado, es hacer click en el nombre del archivo y se abrirán automáticamente (si es que ya estas conectado en tu cuenta de Gmail).

Ejemplo del notebook



The screenshot shows a Jupyter Notebook titled "Clase_01.ipynb". The interface includes a top menu bar with options like File, Edit, View, Insert, Runtime, Tools, and Help. Below the menu is a toolbar with icons for adding code or text cells, copying to drive, and status indicators for RAM and disk usage. The notebook content is divided into two main sections, each highlighted with a dashed border. The first section, labeled "texto" in pink, contains a paragraph about a problem involving a queue of tickets and an obelisk. The second section, labeled "código" in purple, contains a Python script that calculates the time and number of tickets needed for the queue to exceed the height of the obelisk. The script uses variables for ticket thickness, obelisk height, and the number of tickets, and includes a while loop and print statements.

Clase_01.ipynb ☆

File Edit View Insert Runtime Tools Help [Cannot save changes](#)

+ Code + Text Copy to Drive

RAM Disk Editing

▼ Ejemplo de programa

Antes de pasar a ver los elementos básicos de Python, vemos un ejemplo de un pequeño programa. El mismo está hecho para resolver el siguiente problema:

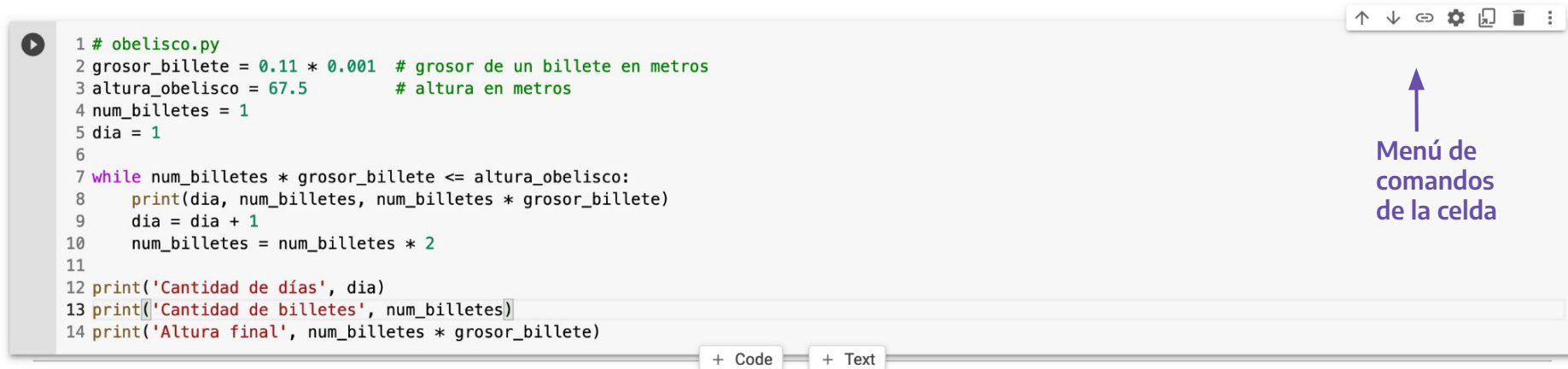
Una mañana se pone un billete en la vereda al lado del obelisco porteño. A partir de ahí, cada día se duplica la cantidad de billetes que hay, apilándolos prolijamente. ¿Cuánto tiempo pasa antes de que la pila de billetes sea más alta que el obelisco?

```
[ ] 1 # obelisco.py
2 grosor_billete = 0.11 * 0.001 # grosor de un billete en metros
3 altura_obelisco = 67.5      # altura en metros
4 num_billetes = 1
5 dia = 1
6
7 while num_billetes * grosor_billete <= altura_obelisco:
8     print(dia, num_billetes, num_billetes * grosor_billete)
9     dia = dia + 1
10    num_billetes = num_billetes * 2
11
12 print('Cantidad de días', dia)
13 print('Cantidad de billetes', num_billetes)
14 print('Altura final', num_billetes * grosor_billete)
```

Con este programa como base, veamos los elementos básicos de un código de Python:

Ejemplo del notebook

1. Elegí cada celda de código que quieras ejecutar parándote sobre ella. Vas a ver en la parte superior izquierda el símbolo 



The screenshot shows a Jupyter Notebook interface. On the left, a code cell contains the following Python code:

```
1 # obelisco.py
2 grosor_billete = 0.11 * 0.001 # grosor de un billete en metros
3 altura_obelisco = 67.5        # altura en metros
4 num_billetes = 1
5 dia = 1
6
7 while num_billetes * grosor_billete <= altura_obelisco:
8     print(dia, num_billetes, num_billetes * grosor_billete)
9     dia = dia + 1
10    num_billetes = num_billetes * 2
11
12 print('Cantidad de días', dia)
13 print('Cantidad de billetes', num_billetes)
14 print('Altura final', num_billetes * grosor_billete)
```

At the bottom of the code cell are two buttons: "+ Code" and "+ Text". To the right of the code cell, there is a vertical arrow pointing upwards to a toolbar. The toolbar contains icons for: up arrow, down arrow, link, settings, insert, delete, and a three-dot menu. Below the arrow, the text "Menú de comandos de la celda" is written.

2. Para ejecutar la celda hacé click en el botón de “Play” (ejecutar) 

Ejemplo del notebook

3. El botón de ejecución de la celda tendrá líneas punteadas en su perímetro. Esto nos indica que la celda está siendo ejecutada y debemos esperar.



The screenshot shows a Jupyter Notebook interface. At the top, the title bar says "Clase_01.ipynb" with a star icon. Below it are menu items: File, Edit, View, Insert, Runtime, Tools, Help, and a status message "Cannot save changes". On the right, there are icons for "Share", "Settings", and a user profile "U". Below the menu bar, there are tabs for "+ Code", "+ Text", and "Copy to Drive". On the right side of this bar, there are indicators for "RAM" and "Disk" usage, and a status "Editing".

The main content area is titled "Ejemplo de programa". It contains a paragraph of text: "Antes de pasar a ver los elementos básicos de Python, vemos un ejemplo de un pequeño programa. El mismo está hecho para resolver el siguiente problema: Una mañana se pone un billete en la vereda al lado del obelisco porteño. A partir de ahí, cada día se duplica la cantidad de billetes que hay, apilándolos prolijamente. ¿Cuánto tiempo pasa antes de que la pila de billetes sea más alta que el obelisco?".

Below the text is a code cell. A blue arrow points to the execution button (a circular icon with a play symbol) on the left side of the code cell. The code cell contains the following Python code:

```
1 # obelisco.py
2 grosor_billete = 0.11 * 0.001 # grosor de un billete en metros
3 altura_obelisco = 67.5      # altura en metros
4 num_billetes = 1
5 dia = 1
6
7 while num_billetes * grosor_billete <= altura_obelisco:
8     print(dia, num_billetes, num_billetes * grosor_billete)
9     dia = dia + 1
10    num_billetes = num_billetes * 2
11
12 print('Cantidad de días', dia)
13 print('Cantidad de billetes', num_billetes)
14 print('Altura final', num_billetes * grosor_billete)
```

At the bottom of the code cell, there are three dots "...".

Ejemplo del notebook

4. Cuando la ejecución termina correctamente, aparece un tilde verde ✓. Esto nos indica que la celda terminó de ejecutarse correctamente. Además aparece el resultado debajo.



The screenshot shows a Jupyter Notebook interface. At the top, the title bar reads "Clase_01.ipynb" with a star icon. Below it is a menu bar with "File", "Edit", "View", "Insert", "Runtime", "Tools", "Help", and "Cannot save changes". On the right, there are icons for "Share", "Settings", and a user profile "U". Below the menu bar is a toolbar with "+ Code", "+ Text", and "Copy to Drive". The main area is titled "Ejemplo de programa" and contains a text block: "Antes de pasar a ver los elementos básicos de Python, vemos un ejemplo de un pequeño programa. El mismo está hecho para resolver el siguiente problema: Una mañana se pone un billete en la vereda al lado del obelisco porteño. A partir de ahí, cada día se duplica la cantidad de billetes que hay, apilándolos prolijamente. ¿Cuánto tiempo pasa antes de que la pila de billetes sea más alta que el obelisco?". Below the text is a code cell with a green checkmark icon and a play button icon. The code is as follows:

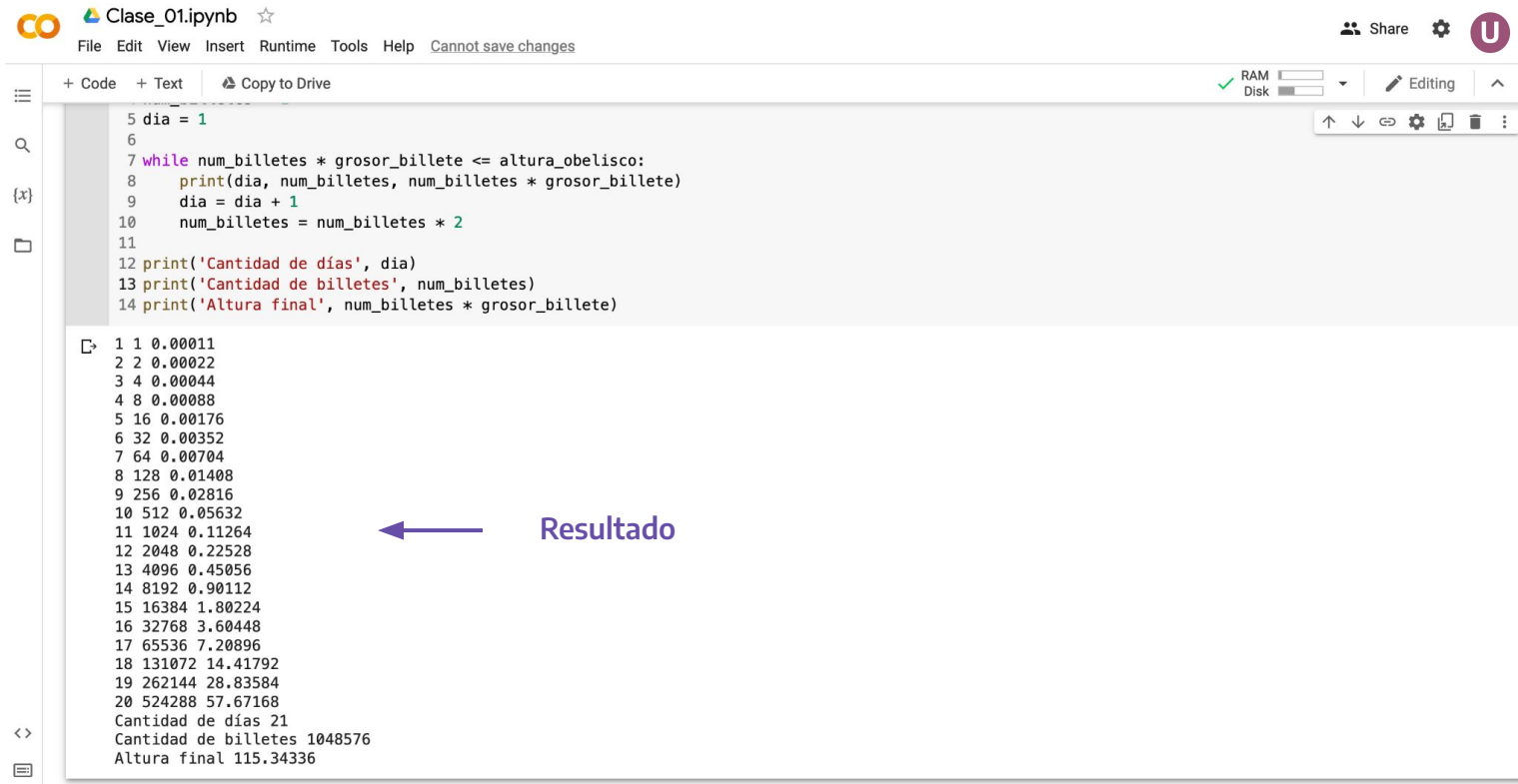
```
1 # obelisco.py
2 grosor_billete = 0.11 * 0.001 # grosor de un billete en metros
3 altura_obelisco = 67.5        # altura en metros
4 num_billetes = 1
5 dia = 1
6
7 while num_billetes * grosor_billete <= altura_obelisco:
8     print(dia, num_billetes, num_billetes * grosor_billete)
9     dia = dia + 1
10    num_billetes = num_billetes * 2
11
12 print('Cantidad de días', dia)
13 print('Cantidad de billetes', num_billetes)
14 print('Altura final', num_billetes * grosor_billete)
```

Below the code cell is the output, which is a list of numbers:

```
1 1 0.00011
2 2 0.00022
3 4 0.00044
4 8 0.00088
5 16 0.00176
6 32 0.00352
7 64 0.00704
```

A blue arrow points to the green checkmark icon, and another blue arrow points to the output text, with the word "Resultado" written next to it.

Ejemplo del notebook



The screenshot shows a Jupyter Notebook titled "Clase_01.ipynb". The interface includes a top menu bar with options like File, Edit, View, Insert, Runtime, Tools, and Help. A status bar at the top right shows "Share", "Settings", and a user profile icon. The notebook is in "Code" mode, and the left sidebar shows icons for file explorer, search, and other functions. The code cell contains a Python script that calculates the number of days, tickets, and final height based on a loop. The output cell shows the results of the script, including a list of values and summary statistics. A blue arrow points from the word "Resultado" to the output cell.

```
5 dia = 1
6
7 while num_billetes * grosor_billete <= altura_obelisco:
8     print(dia, num_billetes, num_billetes * grosor_billete)
9     dia = dia + 1
10    num_billetes = num_billetes * 2
11
12 print('Cantidad de días', dia)
13 print('Cantidad de billetes', num_billetes)
14 print('Altura final', num_billetes * grosor_billete)
```

```
1 1 0.00011
2 2 0.00022
3 4 0.00044
4 8 0.00088
5 16 0.00176
6 32 0.00352
7 64 0.00704
8 128 0.01408
9 256 0.02816
10 512 0.05632
11 1024 0.11264
12 2048 0.22528
13 4096 0.45056
14 8192 0.90112
15 16384 1.80224
16 32768 3.60448
17 65536 7.20896
18 131072 14.41792
19 262144 28.83584
20 524288 57.67168
Cantidad de días 21
Cantidad de billetes 1048576
Altura final 115.34336
```

Resultado

Ejemplo del notebook

5. Para modificar el código de las notebooks basta con seleccionar la celda, realizar cambios y ejecutarla nuevamente.

Atención: Existen celdas de códigos que son dependientes de otras celdas anteriores por lo que se recomienda ejecutar las celdas de forma consecutiva desde el comienzo. Dicho de otra forma, un programa entero puede abarcar varias celdas!

Todos los comandos para correr (y resetear) tu código los encontrás en el menú haciendo click en Runtime y eligiendo el comando que deseases.

Buen trabajo !

