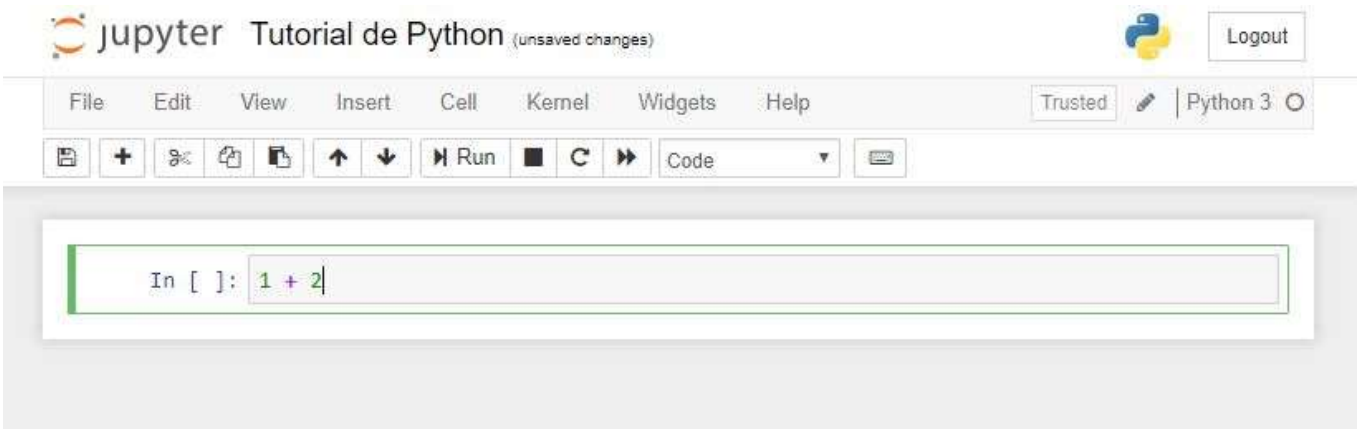
CLUSTER TIC GALICIA

El entorno Jupyter

Ejecución de celdas

Ejecutemos nuestra primera celda. Para ello, hagamos un clic en el interior de la única celda visible y escribamos algo como "1 + 2":



Si te has limitado a escribir el texto comentado verás que la celda muestra una barra verde a la izquierda, indicando que es la celda activa.

Ahora, hagamos clic en el botón "*Run*" que hay en la cinta horizontal de herramientas que se muestra en la parte superior del cuaderno. El resultado es el siguiente:

Entre corchetes (In [1] y Out[1]) se muestra el orden en el que se han ejecutado las celdas. Como solo teníamos una, se marca con el número 1. La etiqueta In indica la celda que se ejecuta y la etiqueta Out indica cuál ha sido la salida o resultado de la ejecución de la celda (si es que se ha producido algún resultado).

En nuestro ejemplo concreto, la instrucción a ejecutar era "1 + 2" y, al ejecutar la celda, Python ha calculado el resultado y lo muestra debajo de la celda: 3.

Vemos también que se ha creado automáticamente una segunda celda, debajo de la anterior. Hagamos un

print("Hello World")

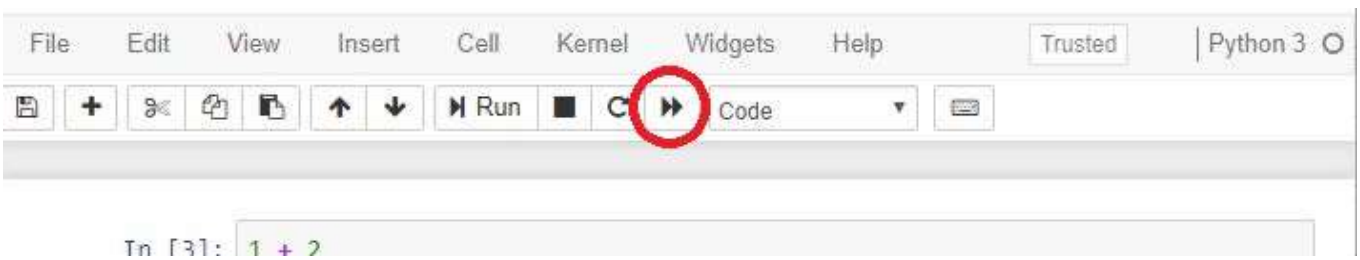
A continuación, volvamos a hacer clic en el botón "*Run*". El resultado es el siguiente:

La celda que acabamos de ejecutar se muestra precedida ahora por la etiqueta In [2], indicando que ha sido la segunda ejecutada, y se ha creado automáticamente otra celda.

Fíjate en que las celdas pueden ser ejecutadas en el orden que deseemos. Si ahora hacemos un clic en la primera celda (la situada en la parte superior) y volvemos a ejecutarla (haciendo clic en "Run"), dicha celda aparecerá etiquetada con el número 3 (pues es la tercera celda que ejecutamos):

En esta ocasión no se ha generado una nueva celda tras la que hemos ejecutado pues ya existía una celda (que se ha seleccionado automáticamente).

Si, en lugar de ejecutar celda por celda, deseas ejecutar el cuaderno completo, puedes hacer un clic en el botón que se muestra con dos triángulos orientados hacia la derecha de la barra de herramientas:



Esto hará que se reinicialice el cuaderno y se ejecuten todas las celdas en orden, una tras otra, comenzando por la superior.

El hecho de que se reinicialice el cuaderno es algo importante a tener en cuenta: cuando ejecutamos una celda lo normal es que se modifique el valor de alguna variable. Y esa variable va a mantener el valor hasta que vuelva a ser modificada (o hasta que se reinicialice el cuaderno). Vamos a verlo en la práctica. Si hemos hecho clic en el botón comentado que reinicializa el cuaderno y ejecuta todas las celdas de nuevo, nuestro cuaderno tendrá el siguiente aspecto:

Ahora escribamos en la tercera celda (la última) el siguiente código:

```
m = 3
```

...y ejecutemos la celda haciendo clic en el botón "Run" (con lo que estamos ejecutando solo esta celda). Lo que hemos hecho con dicha instrucción es crear una variable, llamarla m y asignarle el número entero 3 (la asignación se realiza con un signo "=": damos a lo que esté a la izquierda del signo "=" el valor de lo que esté a la derecha del mismo). Como ya sabemos, tras ejecutar la celda se habrá creado otra debajo. En ésta nueva celda escribamos lo siguiente:

```
m = m + 1
```

Con esta código le estamos diciendo a Python: "Quiero que la variable m (situada a la izquierda del signo =) tome el valor de lo que valga -hasta ese momento- la variable m, más uno (lo que hay a la derecha del signo =). Es decir, estaríamos incrementando el valor de m en uno, de forma que ahora tomaría el valor 4). Si ejecutamos esta celda (con el botón "Run"), nuestro cuaderno tendrá el siguiente aspecto:

Una vez más, tras ejecutar esta última celda, se ha creado otra debajo automáticamente.

Aunque no lo veamos, en alguna parte de la memoria de nuestro ordenador está almacenado el hecho de que la variable m ahora vale 4.

Y hagamos clic ahora en la celda etiquetada como In [4] (la que acabamos de ejecutar) y ejecutémosla de nuevo (con el botón "Run"):

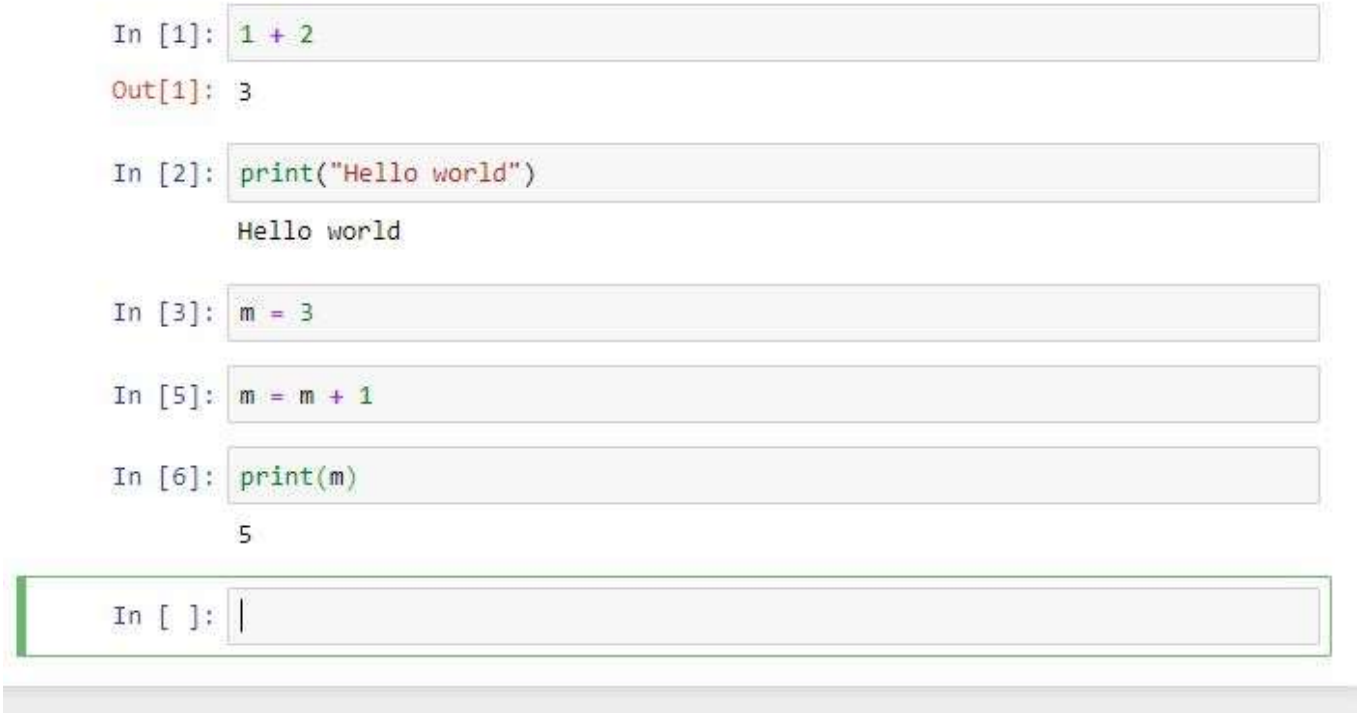
Tras ejecutarla, la celda se muestra precedida por la etiqueta In [5], pues es la quinta celda que ejecutamos.

La pregunta es ¿cuánto vale la variable m ahora? Antes valía 4, y en la última celda ejecutada le hemos dicho a Python que vuelva a aumentar el valor de m en uno. Es decir, ahora m vale 5.

Podemos probarlo si en la última celda (la celda vacía que se ha creado) escribimos lo siguiente:

print(m)

...y ejecutamos la celda haciendo clic en el botón "Run". Veremos lo siguiente:



La cuestión es que cualquier operación que realicemos con m considerará su valor actual (5). Y si ejecutamos la celda anterior 7 veces más, m acabará valiendo 12. Y esto no es obvio viendo el código: En nuestro programa no hay ninguna instrucción que asigne a m el valor 12. m vale 12 porque hemos ejecutado las celdas en cierto orden, simplemente. Esto muy potente pues nos permite ejecutar nuestro programa por bloques y realizar las asignaciones que en cada momento nos interese, pero también es peligroso, pues si olvidamos que hemos ejecutado una cierta celda 7 veces (por ejemplo) estaremos olvidando también que el valor de una variable se ha incrementado un número semejante de veces.

Al reinicializar el cuaderno y ejecutándolo todo (haciendo clic en el icono que contiene dos triángulos apuntando hacia la derecha) estamos eliminando todas las variables que existan y comenzando la ejecución del código desde cero, lo que nos asegura que no vamos a encontrarnos ninguna sorpresa.

Inicio