



Visión Artificial
Instalación Python 3 + OpenCV
Semestre 2018-03



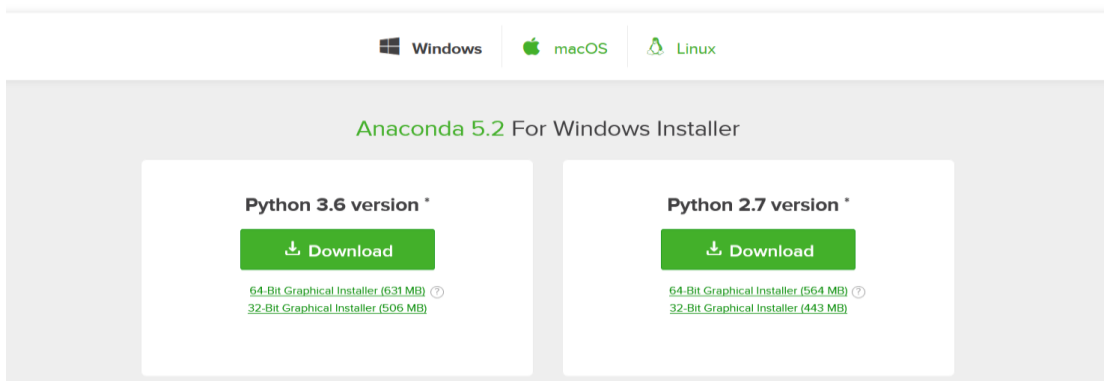
Muchos sistemas operativos incluyen por defecto una distribución de Python. Sin embargo, se recomienda instalar la distribución Anaconda, la cual incluye la mayoría de los paquetes necesarios para el desarrollo de esta clase.

Para el desarrollo de este curso se hará uso de la distribución para Python3 de Anaconda, llamada Anaconda3.

1. INSTALACIÓN EN WINDOWS

Se accede a la distribución de anaconda desde el siguiente enlace:

<https://www.anaconda.com/download/>



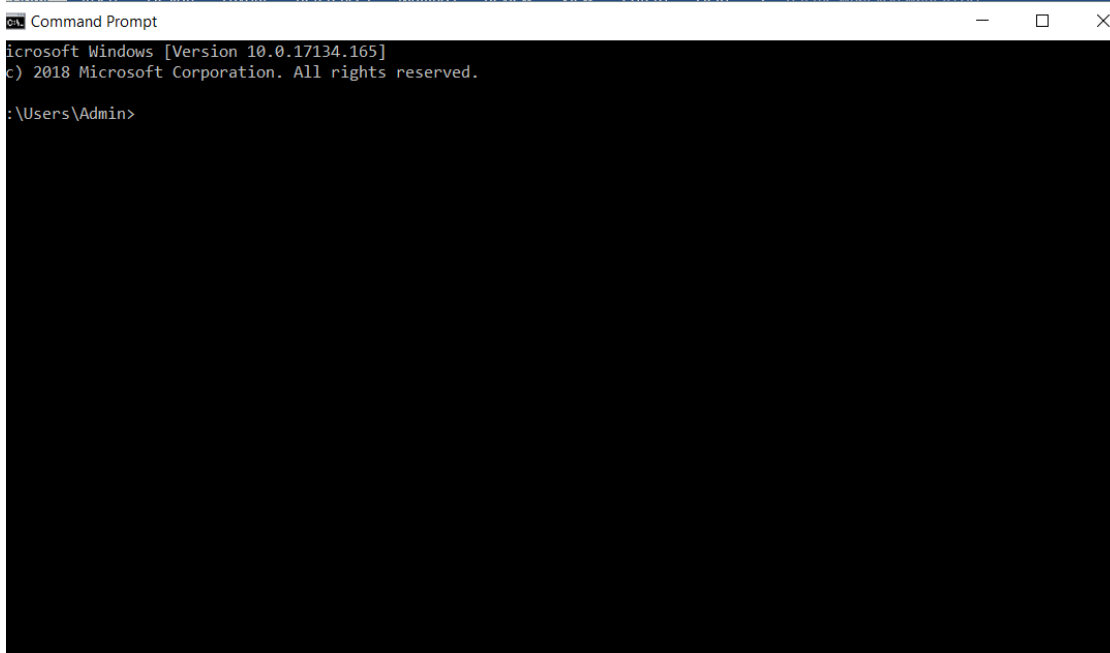
Una vez dentro, hacemos clic en el botón *Download*, debajo de *Python 3.6 version **. Si su ordenador no es de 64 bits, pueden usar el instalador de 32 bits que aparece debajo. Una vez descargado el archivo, procedemos a instalarlo como cualquier *.exe*.

La única consideración que hay que tener en cuenta a la hora de instalar, por motivos de comodidad, es clicar la casilla que pregunta si queremos añadir Anaconda al PATH. Esto es muy conveniente ya que nos permite ejecutar comandos de la distribución de Python que estemos usando desde cualquier ventana de línea de comandos.

2. CREANDO UN AMBIENTE E INSTALANDO OPENCV.

Instalar OpenCV es muy simple, pero es recomendable hacerlo en un ambiente aislado (*environment*) a fin de no crear conflictos en nuestra distribución de Python base. Para ello seguimos los siguientes pasos:

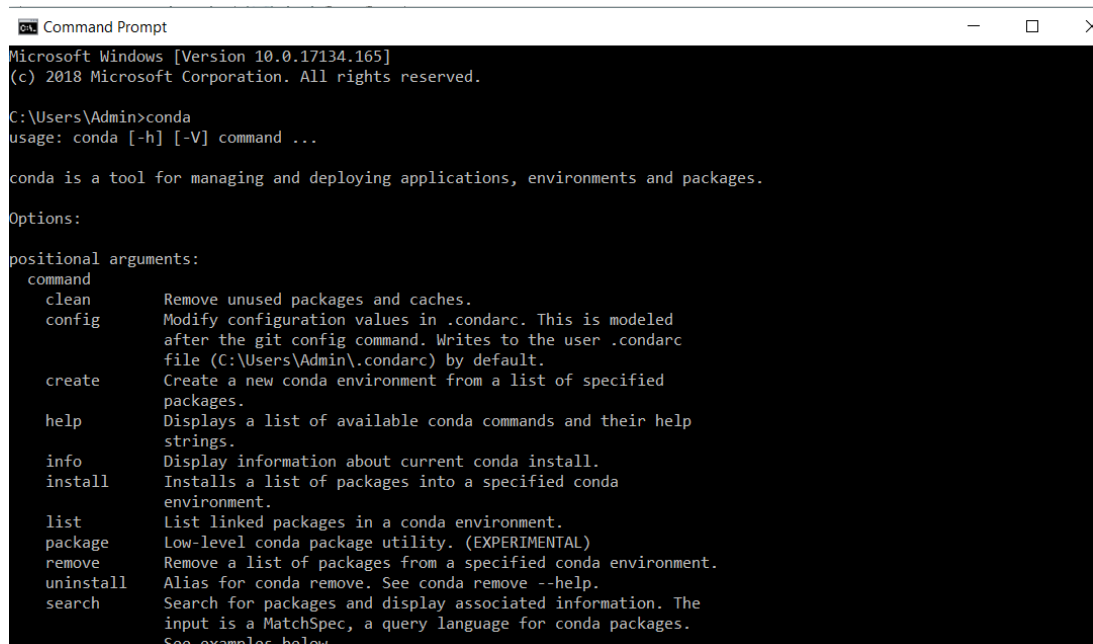
- Abrimos una ventana de línea de comandos (cmd).



```
Command Prompt
Microsoft Windows [Version 10.0.17134.165]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Admin>
```

- Escribimos el comando `conda` para verificar que nuestra distribución de Anaconda funcione correctamente.



```
Command Prompt
Microsoft Windows [Version 10.0.17134.165]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Admin>conda
usage: conda [-h] [-V] command ...

conda is a tool for managing and deploying applications, environments and packages.

Options:
  -h, --help            show this help message and exit
  -V, --version          display the version number and exit

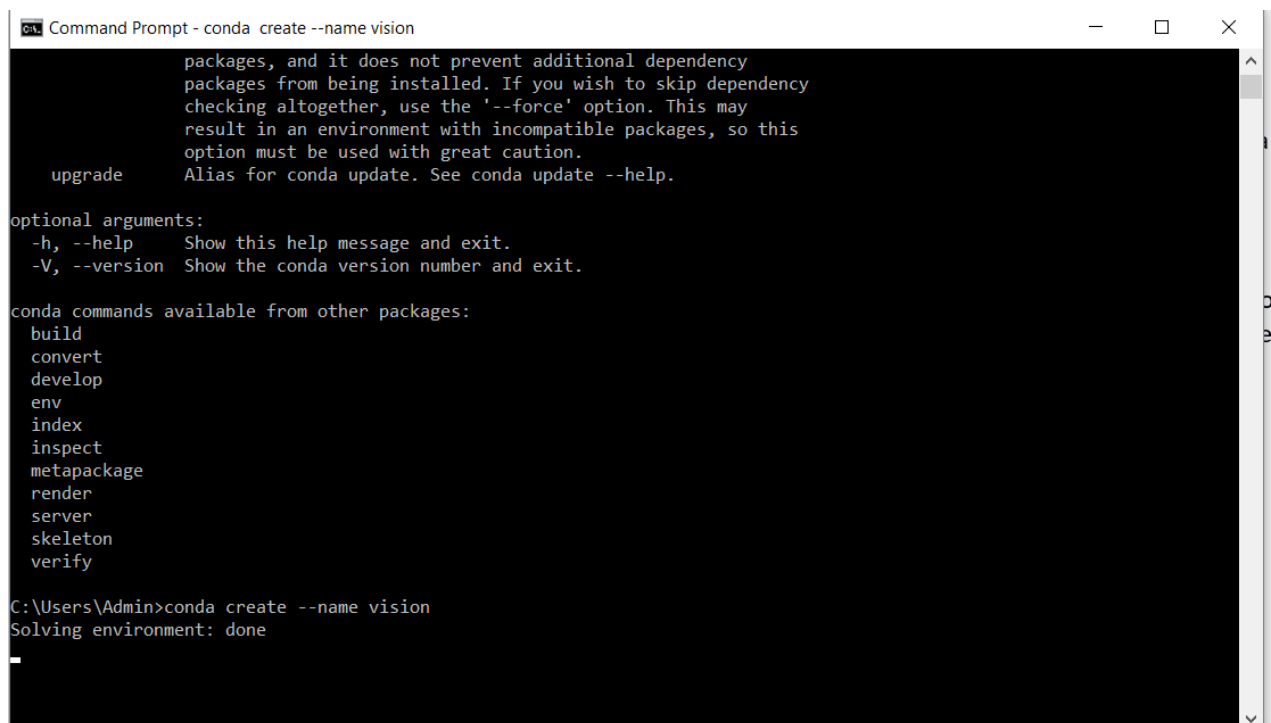
positional arguments:
  command
    clean                Remove unused packages and caches.
    config               Modify configuration values in .condarc. This is modeled
                        after the git config command. Writes to the user .condarc
                        file (C:\Users\Admin\.condarc) by default.
    create               Create a new conda environment from a list of specified
                        packages.
    help                 Displays a list of available conda commands and their help
                        strings.
    info                 Display information about current conda install.
    install              Installs a list of packages into a specified conda
                        environment.
    list                 List linked packages in a conda environment.
    package              Low-level conda package utility. (EXPERIMENTAL)
    remove               Remove a list of packages from a specified conda environment.
    uninstall            Alias for conda remove. See conda remove --help.
    search               Search for packages and display associated information. The
                        input is a MatchSpec, a query language for conda packages.
                        See examples below.
```

- Creamos un nuevo ambiente utilizando la siguiente instrucción desde la línea de comando:

```
conda create --name [vision]
```

ATENCIÓN: Los corchetes son para indicar que en esa posición se coloca el nombre que deseamos para nuestro ambiente de Anaconda. Deben omitirse al momento de ejecutar el comando. Asimismo, es recomendable evitar el uso de tildes y otros caracteres especiales, una excepción común cuando se tienen nombres compuestos por varias palabras es el carácter *underscore* (`_`). Ej: vision_artificial.

NOTA: Existen otras opciones que podemos utilizar al momento de crear un ambiente, para saber cuales son pueden acceder a la documentación en el siguiente enlace: <https://conda.io/docs/user-guide/tasks/manage-environments.html>



```
Command Prompt - conda create --name vision

packages, and it does not prevent additional dependency
packages from being installed. If you wish to skip dependency
checking altogether, use the '--force' option. This may
result in an environment with incompatible packages, so this
option must be used with great caution.
upgrade      Alias for conda update. See conda update --help.

optional arguments:
-h, --help      Show this help message and exit.
-V, --version   Show the conda version number and exit.

conda commands available from other packages:
build
convert
develop
env
index
inspect
metapackage
render
server
skeleton
verify

C:\Users\Admin>conda create --name vision
Solving environment: done
```

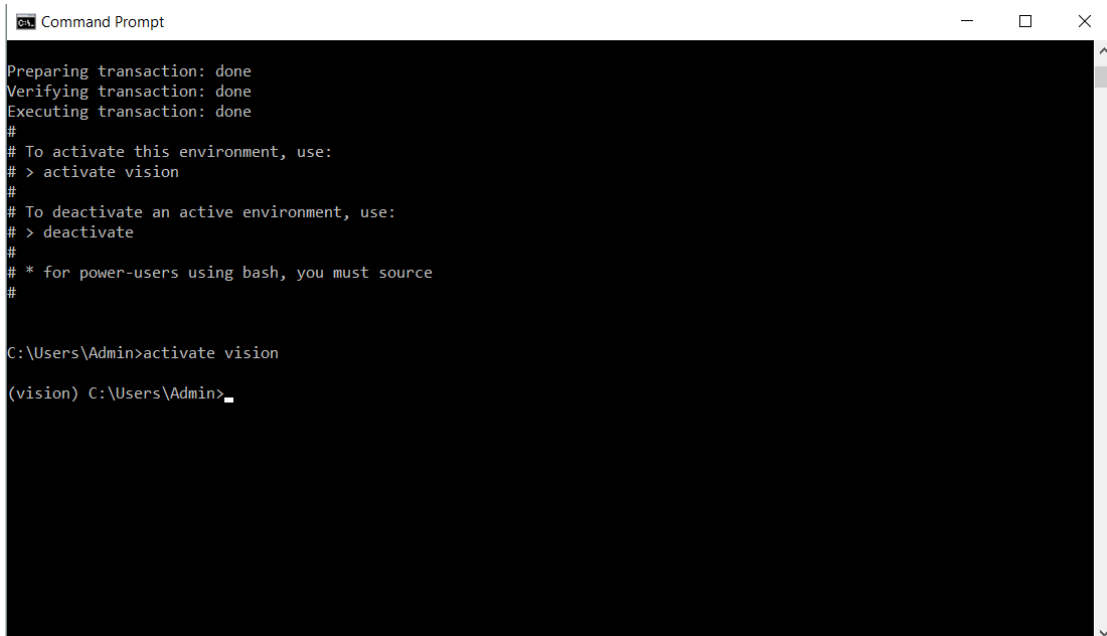
- Una vez creado el ambiente, podemos activarlo usando siguiente comando:

```
activate [vision]
```

- Y podemos desactivarlo usando siguiente comando:

```
deactivate
```

- En cualquier caso, volvamos a activar nuestro ambiente de Anaconda con el comando mencionado anteriormente. Nos podemos dar cuenta de que nuestro ambiente está activo porque al *prompt* de la línea de comando tiene como prefijo el nombre de nuestro ambiente entre paréntesis.



```
Command Prompt
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use:
# > activate vision
#
# To deactivate an active environment, use:
# > deactivate
#
# * for power-users using bash, you must source
#

C:\Users\Admin>activate vision
(vision) C:\Users\Admin>
```

- A continuación, instalaremos la librería Open CV. Esta es una de las librerías más completas de visión artificial para Python, C++ y Java. Para ello, estando dentro de nuestro ambiente, escribimos el comando siguiente:

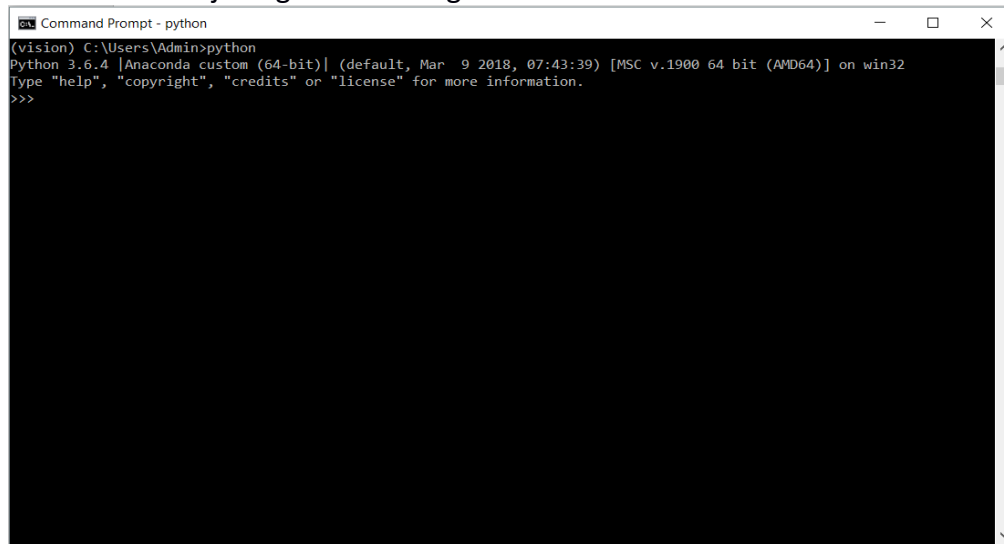
```
pip install opencv-contrib-python
```

NOTA: el comando `pip install` (perteneciente a la librería `pip`) puede ser utilizado para instalar otras librerías como `skimage`, `sklearn`, `matplotlib`, `scipy`, entre otras, aunque la mayoría de estas vienen por defecto con la distribución de Anaconda.

- La línea de comando puede ser utilizada también para programar, aunque en nuestro caso la utilizaremos para comprobar que la instalación de Open CV se haya completado exitosamente. Para ello, ejecutamos el siguiente comando dentro de nuestro ambiente:

```
python
```

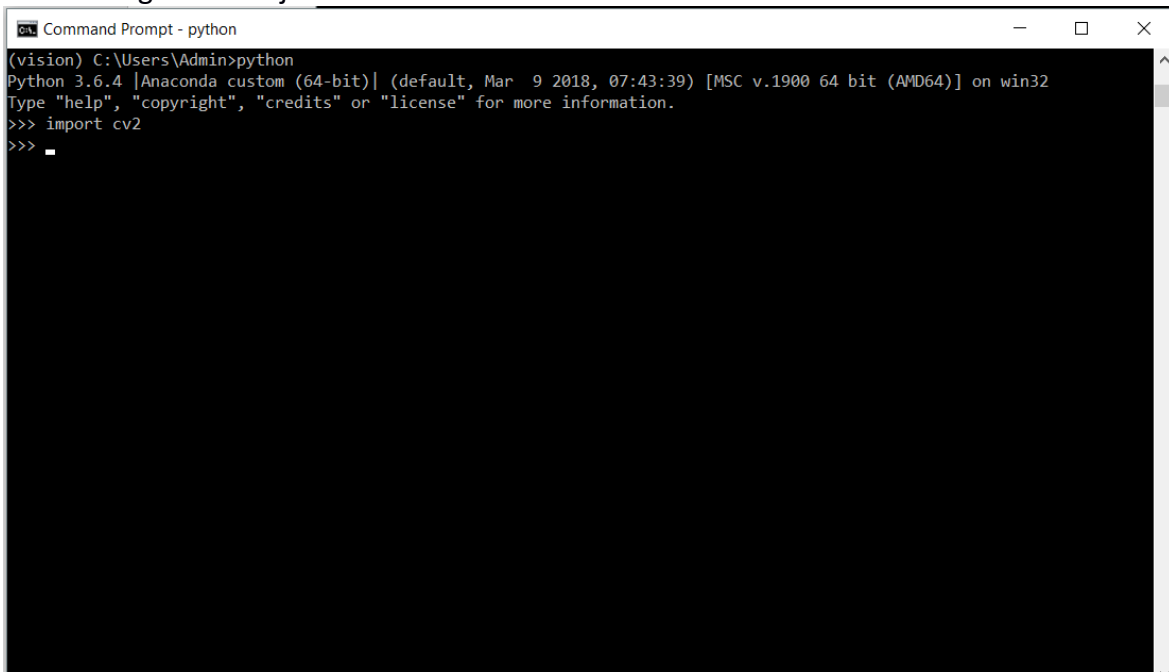
- Ejecutar el comando anterior activa el modo de Python en línea de comando, que nos debería dejar algo como lo siguiente:



- En este modo es posible ejecutar líneas de código, pero nuestra mayor preocupación es saber si Open CV se instaló correctamente, por lo que simplemente utilizaremos el siguiente comando:

```
import cv2
```

- En caso de que nuestra instalación se haya realizado correctamente, no aparecerá ningún mensaje:



- Para salir del modo de ejecución de código de Python, basta con escribir el comando `exit()`
- Esto concluye el tutorial de instalación de Python y OpenCV. Para los que están acostumbrados a la línea de comando hay buenas noticias: Anaconda incluye una aplicación llamada Anaconda Navigator que permite realizar la mayoría de las tareas explicadas anteriormente: <https://www.youtube.com/watch?v=cX6l3lzWewc>
- Es imprescindible leer a continuación los tutoriales de Jupyter Notebook y Spyder3 incluidos en el resto de la documentación.