

# Instrucciones para instalar Rasa

## Requisitos

Verificar que Python y Pip están instalados en nuestro sistema ejecutando

```
python3 --version  
pip3 --version
```

En caso de estar instalados deberíamos ver los números de versión.  
En caso contrario, es necesario instalarlos antes de poder instalar Rasa.

---

## Instalar un ambiente Python

### Procedimiento en Linux

Para instalarlos basta con utilizar nuestro gestor de paquetes. En este caso utilizamos *'apt'*, pero el proceso es idéntico para otros.

```
sudo apt update  
sudo apt install python3-dev python3-pip
```

### Procedimiento de instalación en macOS

Primero instalar el gestor de paquetes [Homebrew](#) ejecutando

```
/bin/bash -c "$(curl -fsSL  
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

Una vez finalizada la instalación podremos ejecutar

```
brew update  
brew install python
```

### Procedimiento de instalación en Windows

Hay que asegurarse de tener instalado el compilador Microsoft VC++, el cual se puede descargar desde [Visual Studio](#).

Ejecutar el instalador y seleccionar VC++ Build tools de la lista.

Para instalar Python, entrar en [este link](#) y descargar el último instalador de windows. (Al momento de confeccionar estas instrucciones la última versión soportada por Rasa según la documentación oficial es la 3.8 pero estas instrucciones funcionan con la última hasta el momento, la 3.9.4)

Luego de pasar por el proceso de instalación abrir una consola y ejecutar

```
pip3 install -U pip
```

---

## Crear un ambiente virtual (Opcional pero recomendable)

Es necesario instalar una herramienta como [virtualenv](#) o [virtualenvwrapper](#). Ambas pueden obtenerse mediante *'pip'*.

```
pip3 install virtualenv
o
pip3 install virtualenvwrapper
```

## Procedimiento para Linux

Para crear un ambiente virtual basta con ejecutar en la carpeta de nuestro proyecto

```
python3 -m venv ./venv
```

Este comando crea un ambiente virtual llamado 'venv' bajo del directorio './venv' usando la misma versión de Python del sistema (La que instalamos en el paso anterior).

Una vez creado, es necesario activarlo antes de usar/desarrollar la aplicación que queramos. Para esto ejecutamos

```
. ./venv/bin/activate
```

Es posible que al intentar crear un ambiente virtual se nos presente un mensaje de error en donde se nos pide instalar el paquete mediante nuestro gestor. En el caso de Ubuntu, nuevamente, lo hacemos mediante 'apt' pero el proceso sería similar con otros gestores.

```
sudo apt-get install python3-venv
```

## Procedimiento para MacOS

El procedimiento es el mismo que para Linux

## Procedimiento para Windows

Para crear un ambiente virtual basta con ejecutar en la carpeta de nuestro proyecto

```
python3 -m venv .\venv
```

Este comando crea un ambiente virtual llamado 'venv' bajo del directorio './venv' usando la misma versión de Python del sistema.

Una vez creado, es necesario activarlo antes de usar/desarrollar la aplicación que queramos. Para esto ejecutamos

```
.\venv\Scripts\activate
```

---

## Instalar Rasa Open Source

El proceso es el mismo para las 3 variantes de sistema operativo.

```
pip3 install -U pip // asegura que tengamos la última versión de pip
pip3 install rasa
```

---

## Crear un asistente básico

En la carpeta del proyecto ejecutamos

```
// En caso de haber creado un ambiente virtual, activarlo
. ./venv/bin/activate //.\venv\Scripts\activate para Windows
```

```
rasa init
```

Este comando crea la siguiente estructura de archivos

```
.
├── actions
│   ├── __init__.py
│   └── actions.py
├── config.yml
├── credentials.yml
├── data
│   ├── nlu.yml
│   └── stories.yml
├── domain.yml
├── endpoints.yml
├── models
│   └── <timestamp>.tar.gz
├── tests
│   └── test_stories.yml
```

Modificando estos archivos le daremos forma a nuestro asistente.

Para poder probar el asistente basta con ejecutar

```
rasa run
```