

## Empezando con Python en IBM System i

Octubre 2020

Clay Lanzino

Por un poco más de 30 años, he venido desarrollando y manteniendo sistemas de información para empresas de distintos sectores comerciales y de gobierno, específicamente en ambientes IBM; S/3x, AS/400 y System i, utilizando como lenguaje de programación el robusto RPG en sus diferentes versiones (II, III e ILE), ésta ha sido mi zona de confort durante muchos años donde me he sentido muy seguro.

Desde hace un par de años atrás, he venido escuchando de un lenguaje de programación orientado a objetos, con el cual según se podía hacer infinidades de proyectos tecnológicos con un mínimo de líneas de códigos; al decir verdad, al principio dude un poco, sin embargo, la curiosidad me llevó a investigar en internet, acerca de este popular lenguaje de programación conocido como Python.

A principios del 2019, decidí descargar desde YouTube mi primer curso de programación básica de Python, apenas antes de haber completado la mitad del curso ya estaba convencido de que este lenguaje de programación había conquistado mi corazón.

Por años había intentado aprender otros lenguajes de programación, con la idea de salir de mi zona de confort y adentrarme en otras plataformas tecnológicas; sin embargo, muchos de ellos me parecieron aburridos o tediosos en su estilo de codificación, no sin mencionar la cantidad de líneas de códigos que se deben utilizar para ejecutar una regla de negocio o función.

Sin duda alguna, el lenguaje de programación Python, no va a reemplazar mi lenguaje de programación predilecto que es el RPGILE de IBM, no obstante, ya lo estoy utilizando como mi lenguaje de programación alternativo.

Python no solo sintetiza en una o pocas líneas de código de programa, lo que en otros lenguajes de programación te llevaría muchas líneas, sino que, es un lenguaje multiplataforma, su sintaxis es muy divertida y sencilla por estar más cercana al lenguaje natural de los humanos y como valor agregado, resulta mucho más comprensible para aprendices por su simplicidad, versatilidad y popularidad.

Haciendo un poco de historia, Guido van Rossum ideó el lenguaje Python a finales de los 80 y comenzó a implementarlo en diciembre de 1989. En febrero de 1991 publicó la primera versión pública, la versión 0.9.0. La versión 1.0 se publicó en enero de 1994, la versión 2.0 se publicó en octubre de 2000 y la versión 3.0 se publicó en diciembre de 2008.

Hasta Julio del 2018, el desarrollo de Python estuvo dirigido personalmente por Guido van Rossum y bajo el paraguas de la Python Software Foundation. A partir del 2019 el desarrollo de Python está presidido por un consejo de dirección de cinco miembros elegidos entre los desarrolladores de Python, el cual se renueva anualmente.

Para descargar la última versión de Python, por favor visite su sitio oficial de descargas:

<https://www.python.org/downloads/>

Ahora, ¿cómo adentrarse en el mundo de IBM System i (AS/400) utilizando el potente lenguaje de programación Python, recurriendo a las mejores prácticas y no morir en el intento?

Hay muy poca documentación en internet acerca de esto y la que encuentras no dicen casi nada. Obviamente, conectarse a un Servidor IBM System i usando Python como lenguaje de programación dependerá de lo que quieras hacer.

A continuación se presenta la forma de conectarse a un servidor IBM System i, utilizando los métodos `ftp.connect` y `ftp.login` usando un script Python. En el siguiente enlace podrás observar un ejemplo sencillo de cómo utilizar dichos métodos:

[https://github.com/ClayLanzino/IBM-System-i\\_AS400/blob/master/Python\\_Scripts/Ftp\\_IBM\\_System\\_i.py](https://github.com/ClayLanzino/IBM-System-i_AS400/blob/master/Python_Scripts/Ftp_IBM_System_i.py)

Otra manera pudiera ser, escribiendo un Script que corra en el cliente y se conecte a una base de datos IBM db2 que resida en un servidor IBM System i, y ejecutar ciertas sentencias SQL.

Hay dos vías para ello, pudiera ser utilizando una conexión iSeries Access ODBC Driver e importando la librería pyodbc (`import pyodbc`). En lo personal muy pocas veces utilizo ODBC Driver para conectarme a Bases de Datos IBM System i, por la vulnerabilidad que representa y por la naturaleza propia de la conexión y que no profundizaré en este momento.

La otra manera sería utilizando las funciones `ibm_db.connect` e `ibm_db.pconnect` respectivamente. Para obtener información de cómo conectarse a una base de datos IBM DB2 utilizando un Script Python por favor visite el siguiente enlace:

[https://www.ibm.com/support/knowledgecenter/es/SSEPGG\\_10.5.0/com.ibm.swg.im.dbclient.python.doc/doc/t0054368.html](https://www.ibm.com/support/knowledgecenter/es/SSEPGG_10.5.0/com.ibm.swg.im.dbclient.python.doc/doc/t0054368.html)

Finalmente, la otra forma de conectarse podría ser, escribiendo el código o Script Python dentro del propio servidor IBM System i; considerando que IBM incorporó el lenguaje de programación Python dentro de su gama de lenguajes de programación a partir de la versión 7R1 o superior del Sistema Operativo del iSeries.

Los prerequisites para instalar Python en un IBM System i son:

1. Python requiere que se instale Portable Application Solutions Environment (PASE) en el servidor IBM i, que está disponible a través de los números de producto 5770SS1 opción 33 y 5733SC1.

IBM recomienda que obtenga los arreglos temporales del programa (PTF) más recientes para el número de producto 5733SC1. Para obtener más información, vaya a la página web IBM i OpenSSH & OpenSSL Community y busque los PTF 5733-SC1.

2. Verifique que los últimos PTF acumulativos estén instalados en el sistema IBM i.
3. Instale Python 3.6 en el sistema IBM i utilizando el nuevo método RPM proporcionado por IBM porque el producto de código abierto 5733OPS ya no está disponible para descargar. Para obtener más información sobre el método RPM, vaya a la página web de IBM i Open Source.

A continuación, se muestran algunos de los comandos de instalación:

- Todo el software proporcionado por los RPM se instala en el directorio con prefijo / QOpenSys / pkgs. Para usar el software, puede calificar completamente la ruta al

programa, o puede agregar / QOpenSys / pkgs / bin a su PATH usando el siguiente comando:

```
PATH=/QOpenSys/pkgs/bin:$PATH  
export PATH
```

- Instale Python 3 y otros paquetes útiles de Python.

```
yum install python3-pip python3-ibm_db python3-itoolkkit
```

- Instale los paquetes de aprendizaje automático de Python 3.

```
yum install python3-numpy python3-pandas python3-scikit-learn python3-scipy
```

- Instalar node.js

```
yum install nodejs10
```

- Instalar GCC y herramientas de desarrollo

```
yum group install "Development tools"
```

- Ejecute el siguiente comando para probar la instalación de Python:

```
python3 --version
```

- Instale el paquete dependiente pycryptodomex usando el siguiente comando:

```
python3 -m pip instalar pycryptodomex
```

Luego tendrá como resultado que el software Python se instala en la computadora IBM i junto con los paquetes dependientes.

Para conseguir ejemplos de Python en IBM i visita el siguiente enlace: <http://ibm.biz/pythonexamplesonibmi>