

Spyder: Un IDE diseñado para científicos

Carlos Córdoba

`ccordoba12@gmail.com`

Continuum Analytics

SciPy Latin America
Posadas, Argentina
22 de Mayo de 2015

Contenidos

- 1 Introducción
- 2 Componentes Básicos
- 3 Componentes Adicionales
- 4 Lo que viene en Spyder 3.0

Contenidos

- 1 **Introducción**
- 2 Componentes Básicos
- 3 Componentes Adicionales
- 4 Lo que viene en Spyder 3.0

¿Quién soy yo?

@ccordoba12

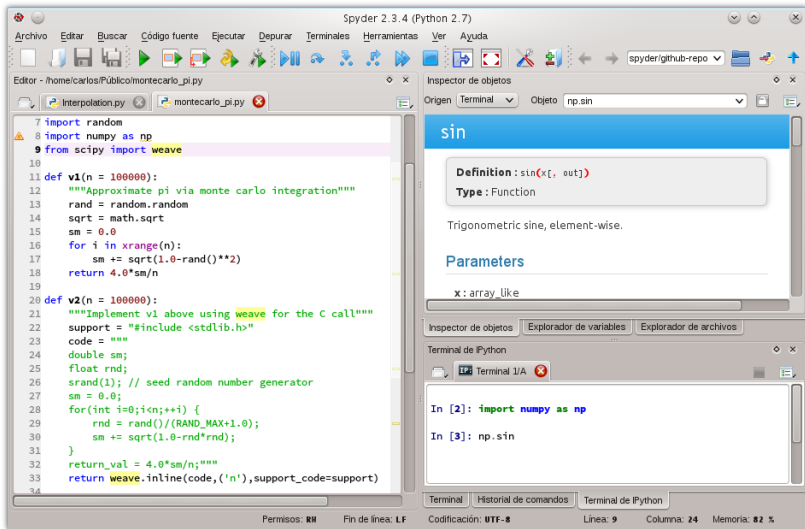
- Trabajo para **Continuum Analytics**
- **Encargado actual** de Spyder
- Estudiante de **PhD en Ing. Industrial** (U. de los Andes, Aplazado)
- **Máster en Física** (U. Nacional de Colombia, 2008)
- Usuario y programador de **Mathematica** por 6 años
- Programador de **C++** por 3 años
- Programador de **Python** desde 2006

¿Qué es Spyder?



- **Spyder** = The **Scientific PY**thon **Development EnviRonment**
- Creado por **Pierre Raybaut** en **2009**
- **Licencia:** MIT
- Un programa para desarrollar **software científico**
- Multiplataforma
- Soporte para **Python 2 y 3**
- **30.000** líneas de código

Cómo luce Spyder



Más sobre Spyder

- Estamos en **Github**:
<https://github.com/spyder-ide/spyder>
- **Fácil** de instalar
 - **Windows**: Anaconda, WinPython o PythonXY
 - **MacOS X**: Anaconda o nuestro instalador DMG
 - **Linux**: Ubuntu, Arch, Fedora, Gentoo
- 5 desarrolladores principales y 25 ocasionales
- 300.000 descargas al año

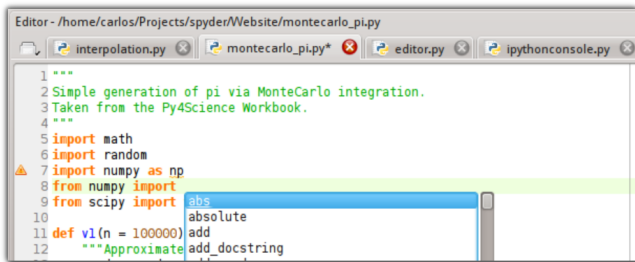
Contenidos

- 1 Introducción
- 2 Componentes Básicos
- 3 Componentes Adicionales
- 4 Lo que viene en Spyder 3.0

Las bases de Spyder

- **Editor:** Escribir código
- **Terminal:** Para correrlo
- **Explorador de Variables:** Visualizar variables generadas en la ejecución
- **Inspector de Objetos:** Visualizar documentación

El Editor: Programar con rapidez



```
Editor - /home/carlos/Projects/spyder/Website/montecarlo_pi.py
interpolation.py x montecarlo_pi.py* x editor.py x ipythonconsole.py x

1 """
2 Simple generation of pi via MonteCarlo integration.
3 Taken from the Py4Science Workbook.
4 """
5 import math
6 import random
7 import numpy as np
8 from numpy import
9 from scipy import abs
10
11 def v1(n = 100000)
12     """Approximate
    absolute
    add
    add_docstring
```

- **Completado** de código \Rightarrow `Tab` ó `Ctrl` + `Espacio`
- Resaltado de **Errores** (rojo) y **Advertencias** (naranja)
- Acceso a la **documentación** \Rightarrow `Ctrl` + `I`
- Ir a la **definición** \Rightarrow `Ctrl` + `G` + nombre

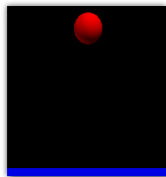
Demostración: Caída de una bola en 1D

- Simulemos la caída de una bola, integrando las ecuaciones de movimiento de Newton:

$$a = \frac{F}{m}$$

$$v(t) = \int a \, dt + v_0$$

$$x(t) = \int v \, dt + x_0$$

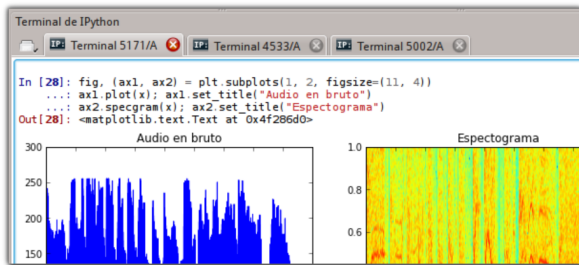


- Asumiendo que $a = -9,8 \text{ m/s}^2$ es constante, y utilizando el método de discretización de Euler, llegamos a que:

$$v_{i+1} = a \Delta t + v_i$$

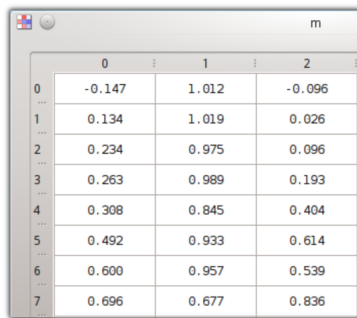
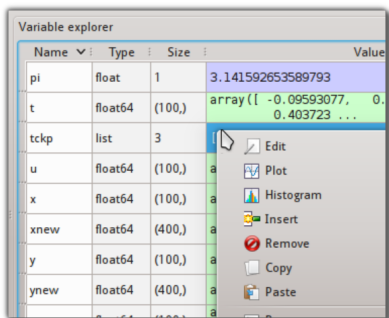
$$x_{i+1} = v_i \Delta t + x_i$$

La Terminal: Evaluar al instante



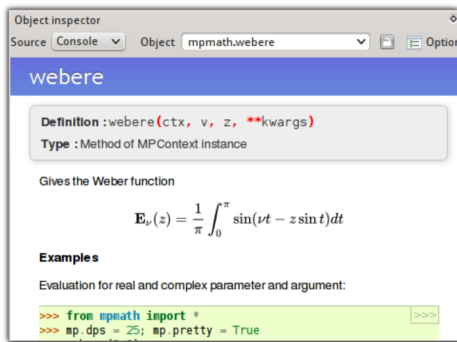
- Evaluar **todo** el archivo \Rightarrow **F5**
- Evaluar una **sección** (o **celda**) \Rightarrow **Ctrl** + **Enter** ó **Shift** + **Enter**
- Evaluar una **selección** o **línea** \Rightarrow **F9**
- Acceso a la **documentación** \Rightarrow **Ctrl** + **I**

El Explorador de Variables: Visualizar variables



- **Inspeccionar** las variables definidas en la terminal
- **Modificar** sus contenidos **gráficamente**
- **Copiar, graficar y remover** variables.

El Inspector de Objetos: Documentación inmediata

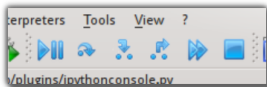


- Docstrings en **texto enriquecido** (con la ayuda de **Sphinx**)
- **Copiar ejemplos** al Editor o la Terminal
- Renderiza ecuaciones escritas en **Latex**

Contenidos

- 1 Introducción
- 2 Componentes Básicos
- 3 Componentes Adicionales**
- 4 Lo que viene en Spyder 3.0

Depurador



Breakpoints			
File	Line	Condition	
interpolation.py	19		
montecarlo_pi.py	30		
montecarlo_pi.py	44		

- Establecer *breakpoints* en el **Editor**
- Barra de herramientas con:
 - Ejecutar la **línea actual** (*Step over*)
 - **Ingresar** y **salir** de la función o método actual (*Step into* / *Step out*)
 - **Continuar** hasta el siguiente *breakpoint* (*Continue*)
- Visualizar todos los *breakpoints* en un componente especial

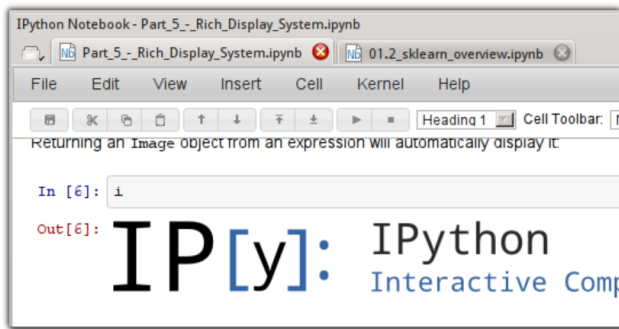
Muchas más facilidades

- Profiler (Perfilador)
- Proyectos
- Buscar en archivos
- Historial de comandos
- Explorador de archivos

Contenidos

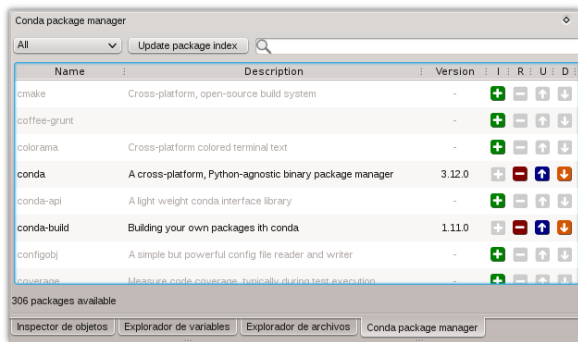
- 1 Introducción
- 2 Componentes Básicos
- 3 Componentes Adicionales
- 4 Lo que viene en Spyder 3.0

Notebooks de IPython



- Cuadernos de trabajo o **Notebooks**, gracias a **IPython 3.0**
- **Integración actual:**
 - Convertir notebooks a archivos de Python
 - Eliminar las salidas (outputs) de los notebooks

Instalador de paquetes de Conda



- Instalar paquetes desde Spyder
- Integrar entornos de conda (*conda environments*) con proyectos de Spyder

Ventajas de Spyder

- Interfaz en **Español** y **Portugués**
- Encontrar **todo** en un **solo** lugar
 - Pero en un ambiente **sencillo** y **fácil de usar**
 - Creado **por** y **para científicos**
- Un **buena puerta** de entrada para aprender **Python**
 - Para estudiantes de **pregrado**
 - Para los **colegas** del trabajo

Ventajas de Spyder

- Interfaz en **Español** y **Portugués**
- Encontrar **todo** en un **solo** lugar
 - Pero en un ambiente **sencillo** y **fácil de usar**
 - Creado **por** y **para científicos**
- Un **buena puerta** de entrada para aprender **Python**
 - Para estudiantes de **pregrado**
 - Para los **colegas** del trabajo

Ventajas de Spyder

- Interfaz en **Español** y **Portugués**
- Encontrar **todo** en un **solo lugar**
 - Pero en un ambiente **sencillo** y **fácil de usar**
 - Creado **por** y **para científicos**
- Un **buena puerta** de entrada para aprender **Python**
 - Para estudiantes de **pregrado**
 - Para los **colegas** del trabajo

Preguntas

¿Preguntas?

Gracias!

Correo: **ccordoba12@gmail.com**

Github: **@ccordoba12**

Twitter: **@ccordoba12**

`https://github.com/spyder-ide/spyder`