INTRODUCCION A LA PROGRAMACIÓN EN PYTHON

Facultad de Estudios Ambientales y Rurales
Departamento de Ecología y Territorio
john.chavarro@javeriana.edu.co

12 de Agosto de 2014

CONTENIDO - PANDAS

1. Definición
2. Instalación
3. Generalidades
4. Estructuras de Datos
5. Creación de Objetos
6. Viendo Datos
7. Rebanado de DataFrame
8. Lectura/Escritura de archivos

Definición

Pandas es una librería de código abierto (al igual que todo en pyhton), de alto rendimiento para el fácil manejo de datos estructurados y análisis de datos en Python.

El mejor tutorial en http://pandas.pydata.org/

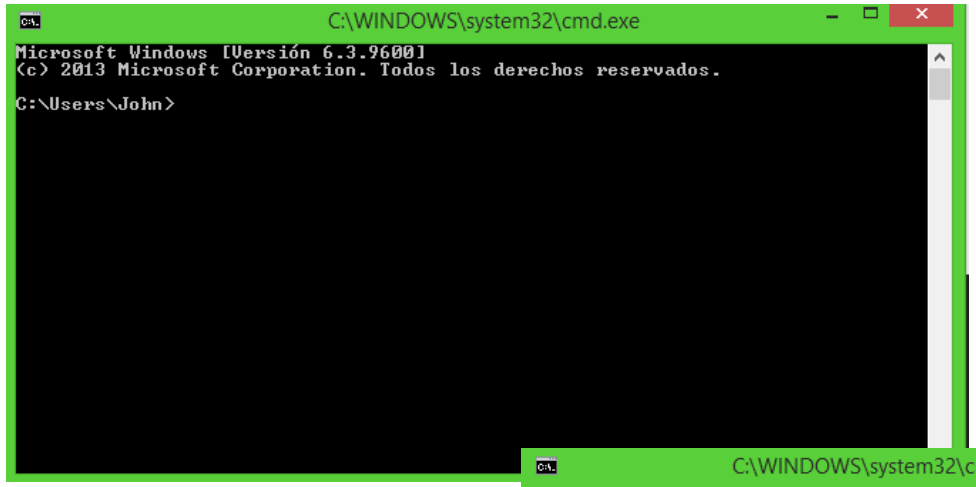
Instalación

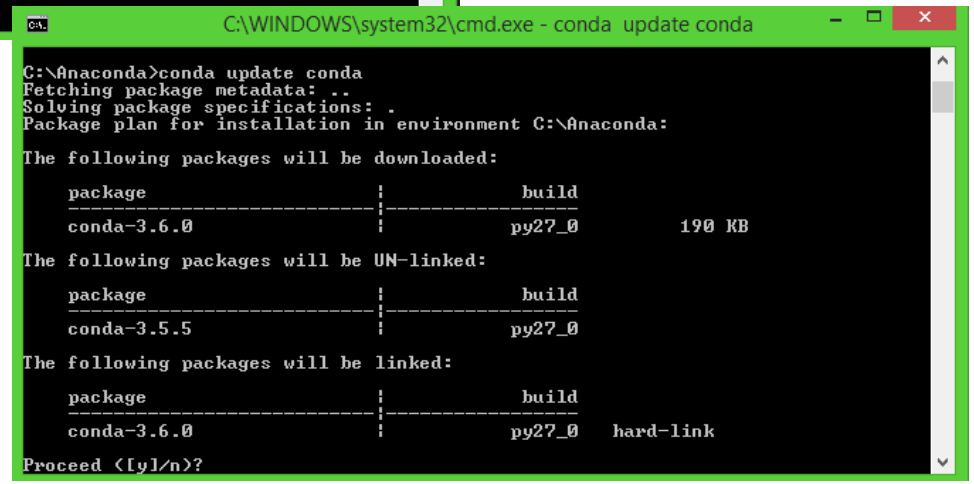
Instalación:

Para **MS-Windows** ir al command

Tecla + R

cmd + ejecutar





Buscar directorio de instalación de Python

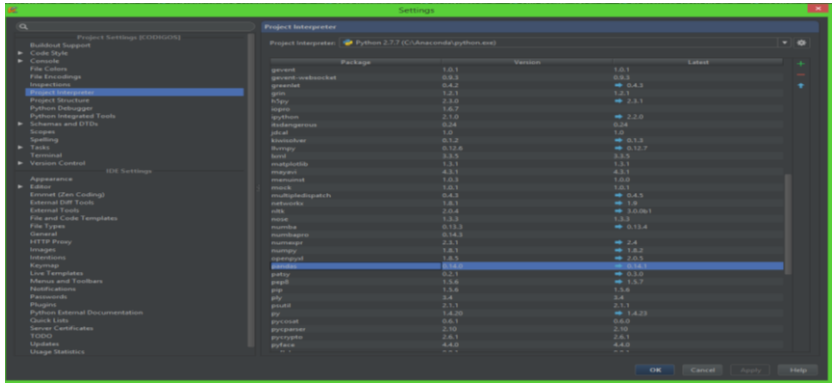
- **Anaconda**
  - Ir a la raíz, normalmente queda instalado en **C:\Anaconda**
  - **conda install pandas**

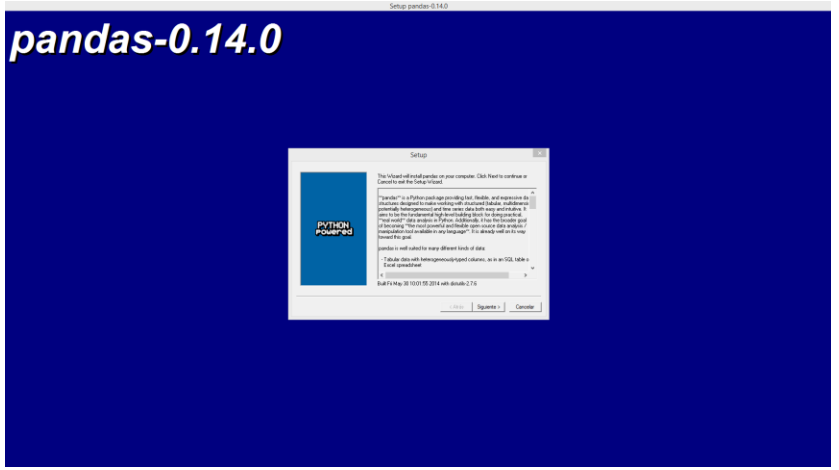
Python Data Analysis Library

Instalación

Overview

Platform	Distribution	Status	Download / Repository Link	Install method
Windows	all	stable	*All platforms*	`pip install pandas`
Mac	all	stable	*All platforms*	`pip install pandas`
Linux	Debian	stable	official Debian repository	`sudo apt-get install python-pandas`
Linux	Debian & Ubuntu	unstable (latest packages)	NeuroDebian	`sudo apt-get install python-pandas`
Linux	Ubuntu	stable	official Ubuntu repository	`sudo apt-get install python-pandas`
Linux	Ubuntu	unstable (daily builds)	PythonXY PPA; activate by: `sudo add-apt-repository ppa:pythonxy/pythonxy-devel && sudo apt-get update`	`sudo apt-get install python-pandas`
Linux	OpenSuse & Fedora	stable	OpenSuse Repository	`zypper in  python-pandas`





Tomado de http://pandas.pydata.org/pandas-docs/stable/install.html#all-platforms

Generalidades

- **PANDAS** consiste en una matriz de datos estructurados y etiquetados, principalmente series de tiempo y DataFrame.
- Al igual que Numpy, los objetos son indexados esta vez no por su posición (aunque también se puede) sino por etiqueta(s) INDICE(S) [**MULTI-INDEXADO**].
- Generación de rangos de fechas (***date_range***)
- Herramientas de **lectura /escritura**: desde y hacia archivos planos tabulados (**CSV, delimitados, Excel**) y objetos de PANDAS [**Pytables/HDF5**].
- Manejo eficiente de datos
- Estadísticas dinámicas

Estructuras de Datos

Dimensión	Tipo	Descripción
1	Series	Etiquetado 1D – Tipo Array
1	Serie de Tiempo	Serie con indexado en el tiempo
2	DataFrame	Etiquetado 2D, multi-indexado, estructura tabular con tipo de columnas heterogéneas
3	Panel	Etiquetado 3D con matrices de diferentes tamaños.

Introducción – Creación de Objetos

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

#  Creacion de objetos [Series]

s = pd.Series([1,3,5,np.nan,6,8])

#  Creacion de objetos [DataFrame]

dates = pd.date_range('20130101',periods=6)
print dates
df = pd.DataFrame(np.random.randn(6, 4), index=dates, columns=list('ABCD'))
print df
```

Introducción – Viendo los datos

```
#  Creando un DataFrame desde un diccionario

df2 = pd.DataFrame({     'A':1.,
                         'B': pd.Timestamp('20130102'),
                         'C': pd.Series(1, index=list(range(4)), dtype='float32'),
                         'D': np.array([3] * 4, dtype='int32')
                         'E':'foo'})
print df2
df2.dtypes

#  Viendo datos

df.head()
df.tail(2)
df.index()
df.columns()
df.values()
df.describe()
```

Introducción – Rebanado de datos

```
#  Seleccionando una columna del DataFrame, lo cual produce una Serie

df[‘A’]
type(df)
type(df[‘A’])

#  Al igual que en Numpy

df[0:3]
df[‘20130102’:’20130104’]

#  Por su etiqueta, funciones loc/iloc

df.loc[dates[0]]
df.loc[:, [’A’,’B’]]
df.loc['20130102':'20130104', ['A','B']]
df.loc[dates[0],'A']
```

Introducción – Lectura/Escritura

```
#  Guardando el DataFrame en un archivo CSV

df.to_csv('foo.csv')

#  leyendo el DataFrame desde un archivo CSV

pd.read_csv('foo.csv')

#  Guardando el DataFrame en un archivo Excel

df.to_excel('foo.xlsx', sheet_name='datos')

#  leyendo el DataFrame desde un archivo Excel
#  cambiemos los valores de algunas celdas del archivo guardado por -99999

pd.read_excel('foo.xlsx', 'datos', index_col=None, na_values=['-99999'])
```