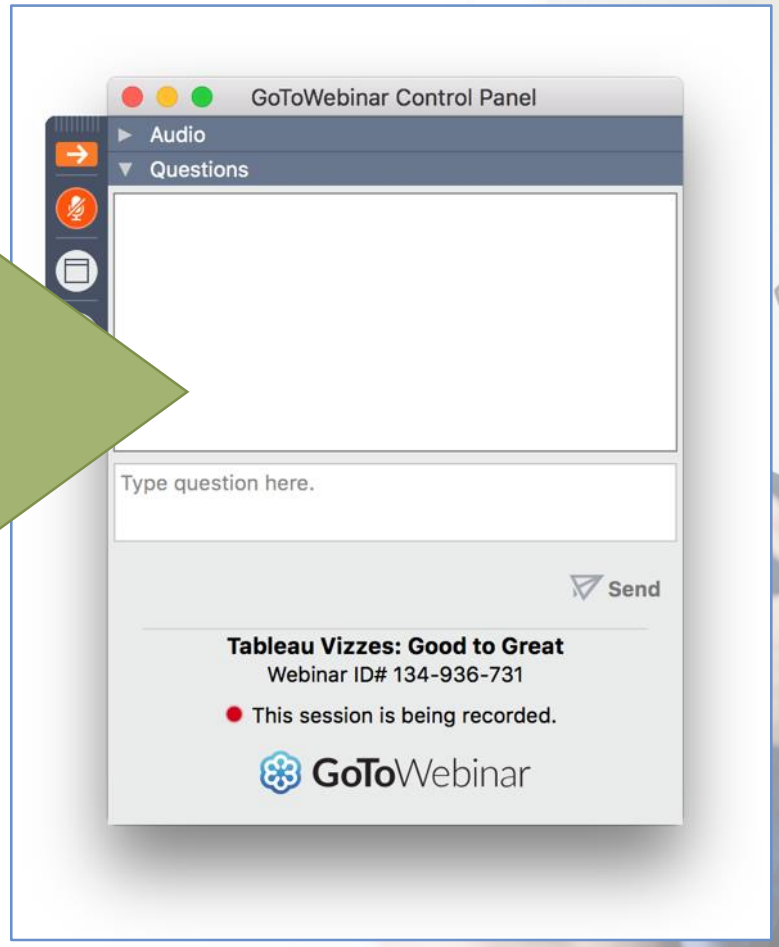
SENTURUS

Using Python with Power BI

GoToWebinar control panel

Click arrow to restore full control panel

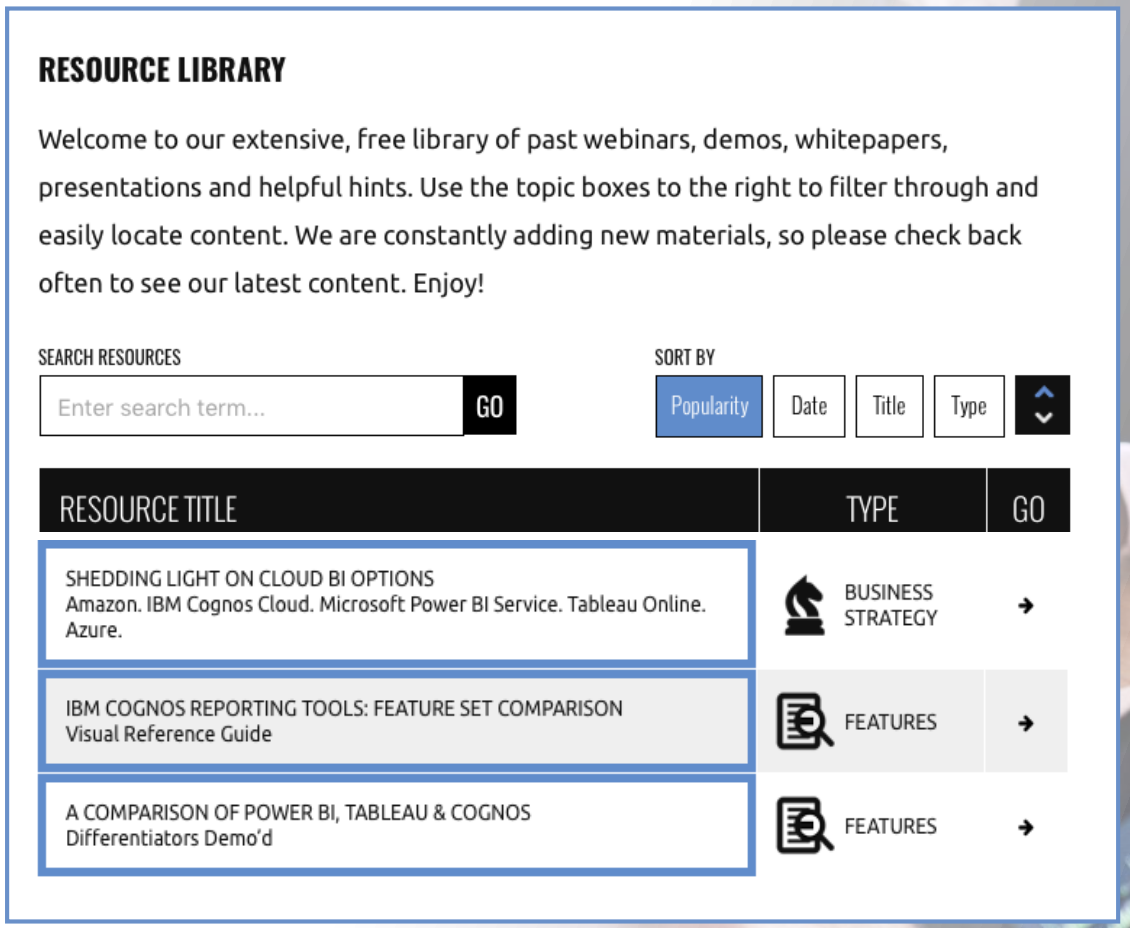
Submit questions in this section



To obtain this presentation

Visit the Resource Library
on the Senturus website
to download this presentation
and explore other assets:

senturus.com/resources



Agenda

- Introductions
- Discussion of Python & Power BI
- Demos
- Additional resources & Senturus Overview
- Q&A

Topics

- Python basics
- Configuring Power BI Desktop
- Using Python for data
- Python visualizations
- Transforming data

Introductions

Patrick Powers

Consultant and Trainer
Senturus, Inc.

Michael Weinhauer

Director
Senturus, Inc.

Poll #1

What is your preferred scripting language?

- Python
- R
- Other
- Don’t use a scripting language

Poll #2

Do you create your own advanced queries with M?

- Yes
- No

What is Python?

- An interpreted, interactive, object-oriented programming language
- Combines power with a straight-forward, easy to understand syntax
- Is portable to Windows, Linux, and MacOS
- Is fully extensible with C or C++
- Named after Monty Python's Flying Circus

Why Python with Power BI?

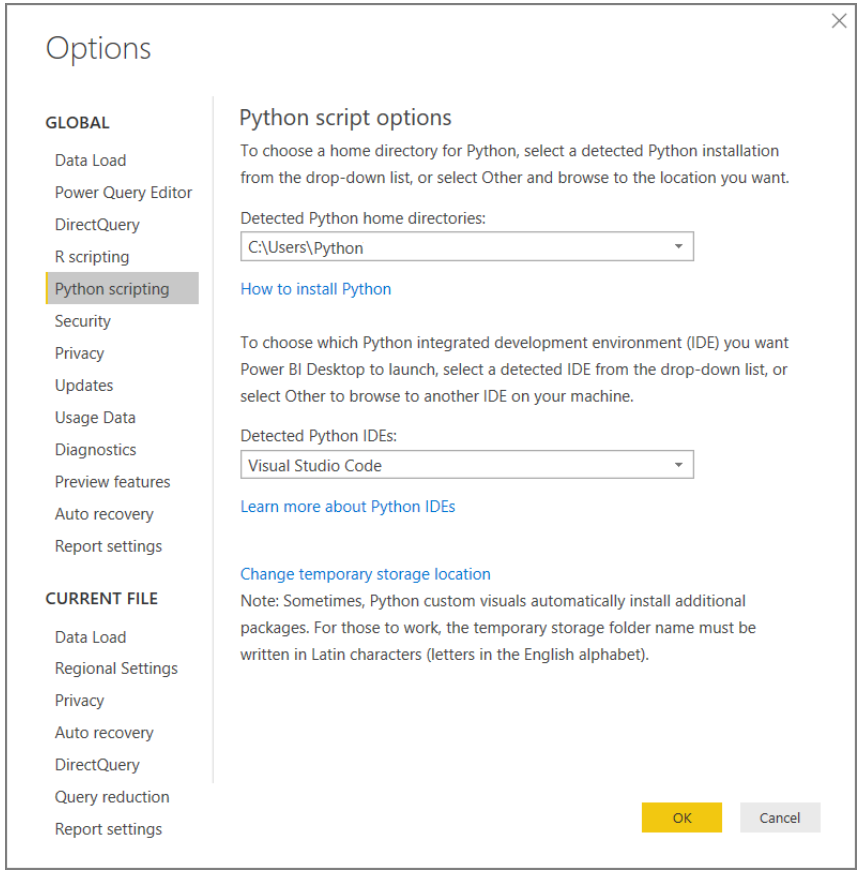
- Can help with automation, creating visuals, and even creating machine learning modules
- Using Python allows access to additional visualizations libraries
- When combined with Power Query, Python can be used for data cleansing, advanced data shaping, and even predictions and clustering

Installing Python

- A base installation of Python is required
- Requires two Python packages before scripts can be used:
  - **Pandas**
    - A software library for data manipulation and analysis. It offers data structures and operations for manipulating numerical tables and time series
  - **Matplotlib**
    - A plotting library for Python and its numerical mathematics extension NumPy
    - Provide an object-oriented API for embedding plots into applications using general-purpose GUI toolkits – e.g. Tkinter, wxPython, Qt, or GTK+

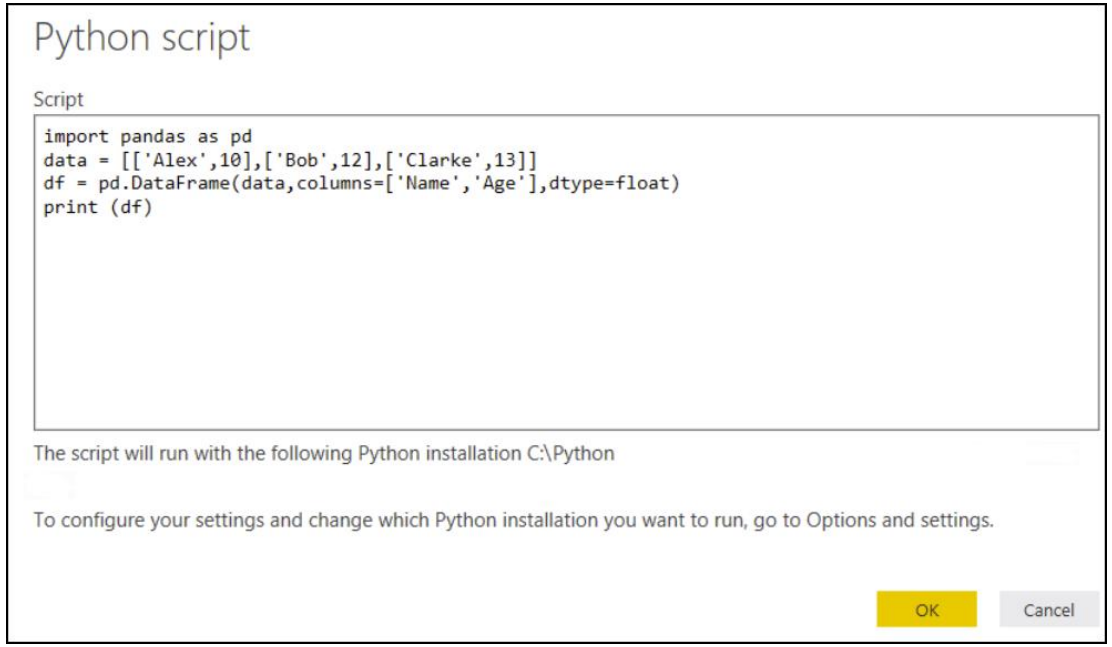
Configuring Python for Power BI

- Once Python and associated libraries are installed, configure Power BI



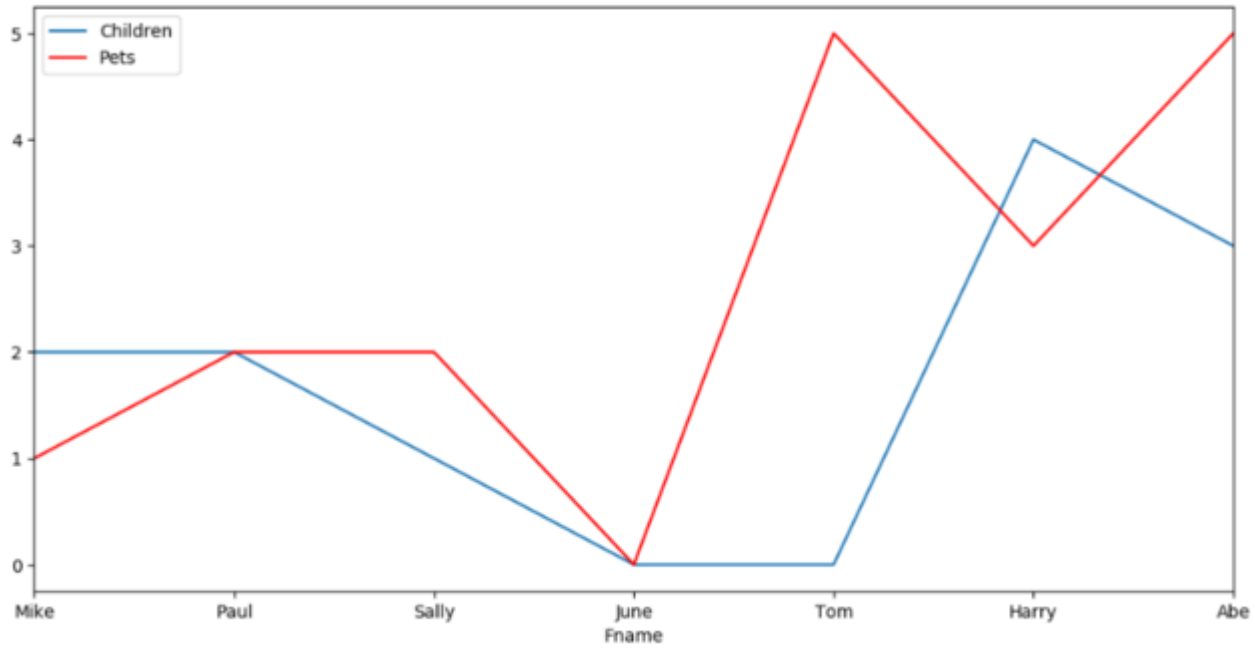
Data Creation

- Python scripts can be used to either hard code data frames or a connection script can be created to use existing data sources



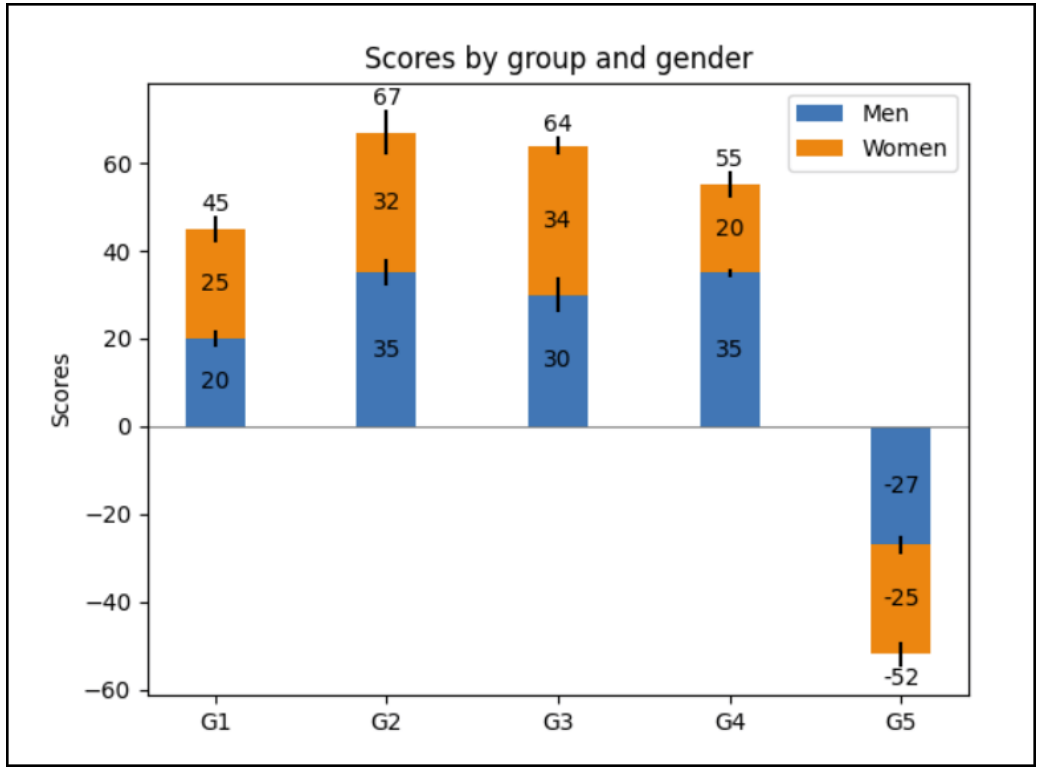
Python Visualizations

- To create visuals with Python, we leverage pandas and matplotlib



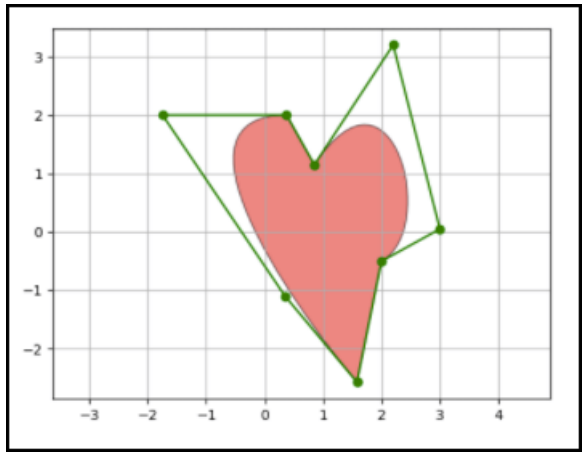
MATPLOTLIB

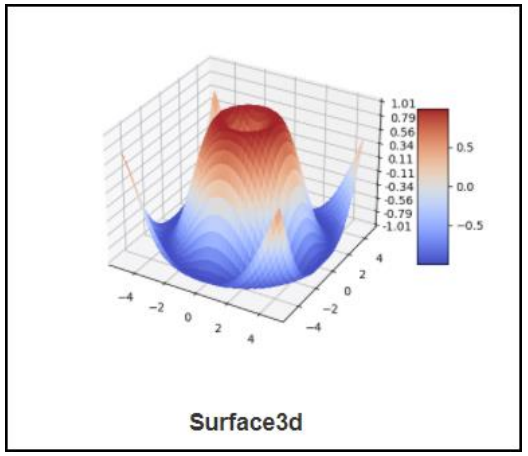
An open source library that allows users to create static, animated, and even interactive visualizations with Python

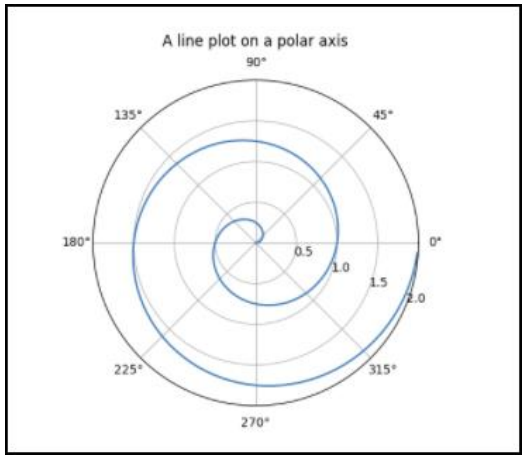


Additional Visualizations

- With MATPLOTLIB, access visualizations not currently available in Power BI







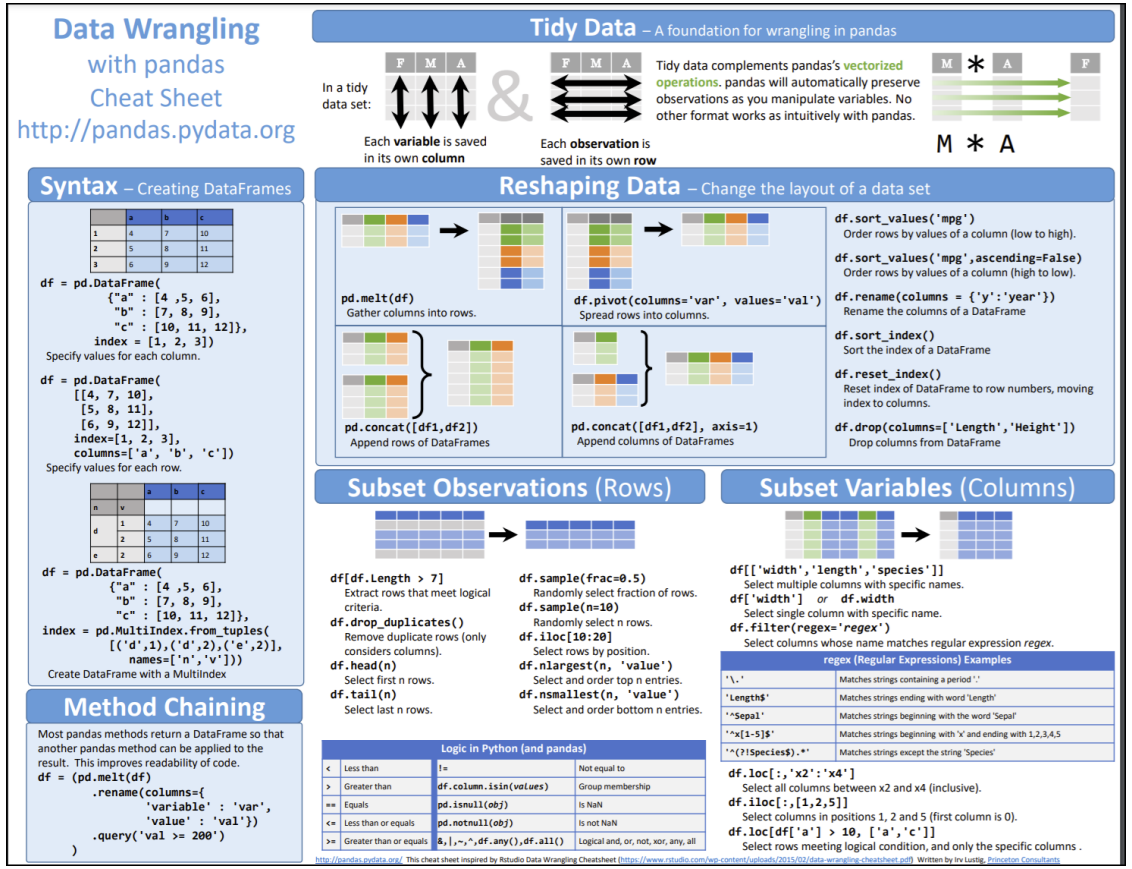
Data Manipulation

- Scripts can be used in Power Query Editor
- Data can be shaped in multiple ways:
  - Completion of missing data
  - Pivoting data
  - Column and row removal
  - Predictions
  - Clustering

PANDAS quick reference

- Print this *Data Wrangling with pandas Cheat Sheet*

  https://pandas.pydata.org/Pandas_Cheat_Sheet.pdf



Data Wrangling with pandas Cheat Sheet http://pandas.pydata.org

Tidy Data – A foundation for wrangling in pandas

In a tidy data set: Each **variable** is saved in its own **column**. Each **observation** is saved in its own **row**.

Tidy data complements pandas's vectorized operations. pandas will automatically preserve observations as you manipulate variables. No other format works as intuitively with pandas.

M * A

Syntax – Creating DataFrames

```
df = pd.DataFrame(
        {"a" : [4 ,5, 6],
         "b" : [7, 8, 9],
         "c" : [10, 11, 12]},
       index = [1, 2, 3])
```
Specify values for each column.

```
df = pd.DataFrame(
    [[4, 7, 10],
     [5, 8, 11],
     [6, 9, 12]],
    index=[1, 2, 3],
    columns=['a', 'b', 'c'])
```
Specify values for each row.

```
df = pd.DataFrame(
        {"a" : [4 ,5, 6],
         "b" : [7, 8, 9],
         "c" : [10, 11, 12]},
index = pd.MultiIndex.from_tuples(
        [('d',1),('d',2),('e',2)],
         names=['n','v']))
```
Create DataFrame with a MultiIndex

Method Chaining

Most pandas methods return a DataFrame so that another pandas method can be applied to the result. This improves readability of code.

```
df = (pd.melt(df)
        .rename(columns={
                'variable' : 'var',
                'value' : 'val'})
        .query('val >= 200')
    )
```

Reshaping Data – Change the layout of a data set

`pd.melt(df)` Gather columns into rows.

`df.pivot(columns='var', values='val')` Spread rows into columns.

`pd.concat([df1,df2])` Append rows of DataFrames

`pd.concat([df1,df2], axis=1)` Append columns of DataFrames

`df.sort_values('mpg')` Order rows by values of a column (low to high).

`df.sort_values('mpg',ascending=False)` Order rows by values of a column (high to low).

`df.rename(columns = {'y':'year'})` Rename the columns of a DataFrame

`df.sort_index()` Sort the index of a DataFrame

`df.reset_index()` Reset index of DataFrame to row numbers, moving index to columns.

`df.drop(columns=['Length','Height'])` Drop columns from DataFrame

Subset Observations (Rows)

`df[df.Length > 7]` Extract rows that meet logical criteria.

`df.drop_duplicates()` Remove duplicate rows (only considers columns).

`df.head(n)` Select first n rows.

`df.tail(n)` Select last n rows.

`df.sample(frac=0.5)` Randomly select fraction of rows.

`df.sample(n=10)` Randomly select n rows.

`df.iloc[10:20]` Select rows by position.

`df.nlargest(n, 'value')` Select and order top n entries.

`df.nsmallest(n, 'value')` Select and order bottom n entries.

Logic in Python (and pandas)

<	Less than	!=	Not equal to
>	Greater than	`df.column.isin(values)`	Group membership
==	Equals	`pd.isnull(obj)`	Is NaN
<=	Less than or equals	`pd.notnull(obj)`	Is not NaN
>=	Greater than or equals	&,|,~,^,df.any(),df.all()	Logical and, or, not, xor, any, all

Subset Variables (Columns)

`df[['width','length','species']]` Select multiple columns with specific names.

`df['width']` or `df.width` Select single column with specific name.

`df.filter(regex='regex')` Select columns whose name matches regular expression *regex*.

regex (Regular Expressions) Examples

`'\.'`	Matches strings containing a period '.'
`'Length$'`	Matches strings ending with word 'Length'
`'^Sepal'`	Matches strings beginning with the word 'Sepal'
`'^x[1-5]$'`	Matches strings beginning with 'x' and ending with 1,2,3,4,5
`'^(?!Species$).*'`	Matches strings except the string 'Species'

`df.loc[:,'x2':'x4']` Select all columns between x2 and x4 (inclusive).

`df.iloc[:,[1,2,5]]` Select columns in positions 1, 2 and 5 (first column is 0).

`df.loc[df['a'] > 10, ['a','c']]` Select rows meeting logical condition, and only the specific columns .

http://pandas.pydata.org/ This cheat sheet inspired by Rstudio Data Wrangling Cheatsheet (https://www.rstudio.com/wp-content/uploads/2015/02/data-wrangling-cheatsheet.pdf) Written by Irv Lustig, Princeton Consultants

Power BI demos

SENTURUS

Demos

Creating an enterprise workbook

- Configuring Power BI and libraries
- Creating data
- Building a visualization
- Transforming data

Python & Power BI Benefits

- Power BI & Python together gives you:
  - Access to re-using existing code
  - More visualizations than currently available in Power BI
  - Greater flexibility in retrieving data from tricky sources
  - A potential replacement for cumbersome M scripts
  - Access to pre-existing scripts to leverage in your workbooks

New self-paced class

Power BI & Python

https://senturus.com/product/power-bi-python/

Complete BI training

Power BI

tableau

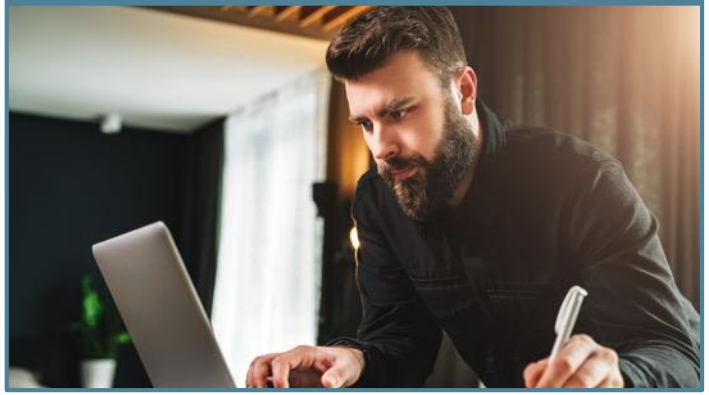
IBM Cognos Analytics

Tailored group sessions

Mentoring

Instructor-led online courses

Self-paced learning



SENTURUS

Additional resources

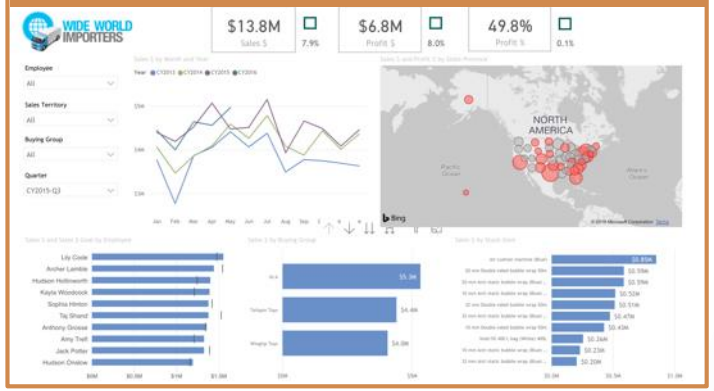
Unbiased product reviews

Technical tips

Insider viewpoints

More on this subject

Product demos



Upcoming events

SENTURUS

Upcoming event

What's New in Cognos 11.2.1

With IBM's Rachel Su

Thursday, Oct 28, 11am PT/2pm ET

Register: https://senturus.com/events/whats-new-in-cognos-11-2-1/

Easily Connect Power BI & Tableau to Cognos Data

Senturus Analytics Connector demo

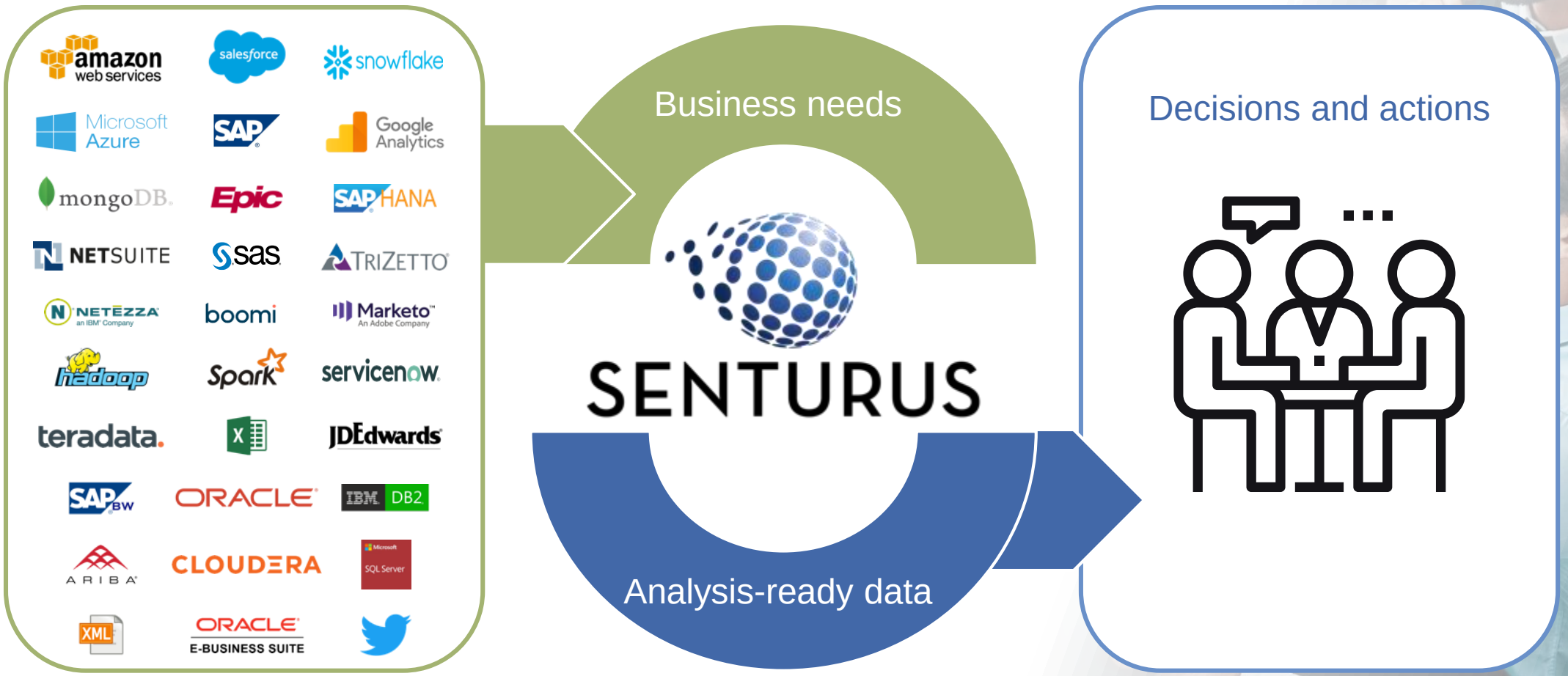
Thursday, Nov 14, 2021, 11am PT/2pm ET

Your path to modern BI

Accelerating self-service analytics for the enterprise

SENTURUS

Bridging the gap



Full spectrum of BI services

- Data preparation and modern data warehousing
- Dashboards, reporting and visualizations in Power BI, Cognos and Tableau
- Hybrid BI environments (migrations, security, etc.)
- Software to accelerate bimodal BI and migrations
- BI services and support retainer (expertise on demand)
- Training and mentoring

A long, strong history of success

20+ years

1350+ clients

3000+ projects

Join the Senturus team

We're hiring talented and experienced professionals

- Senior Microsoft BI Architect
- Senior Azure Architect
- Modern Analytics Solution Architect
- ETL Developer
- And more

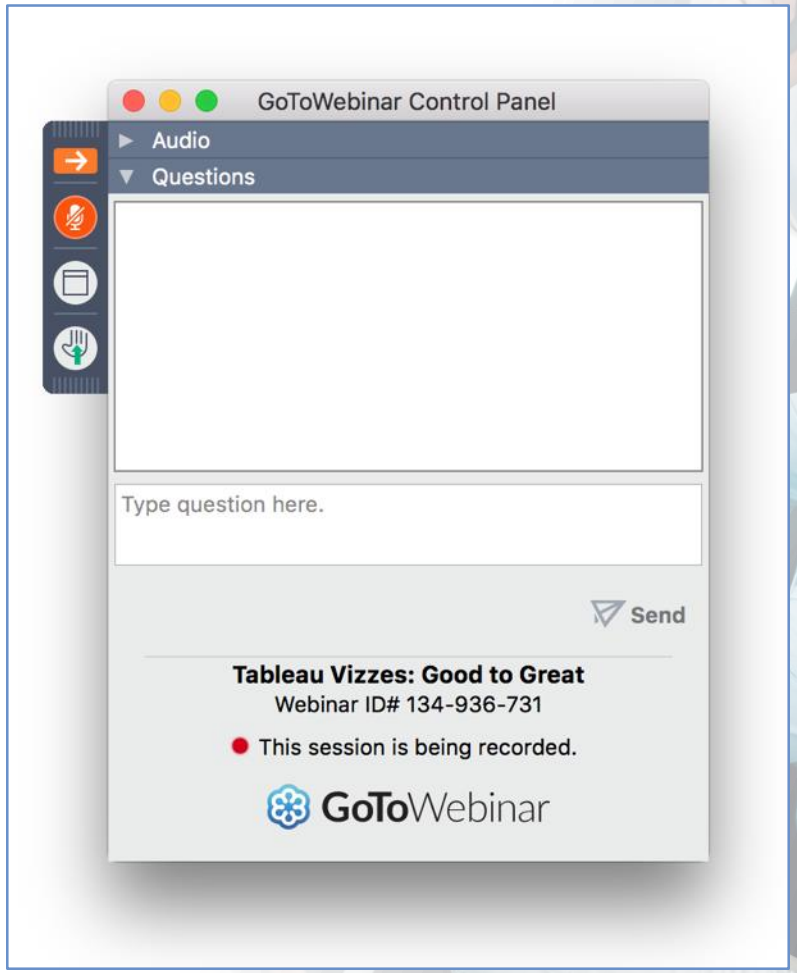
See job descriptions: https://senturus.com/why-senturus/careers/
Send your resume to jobs@senturus.com

Q & A

If your question or issue is broader than what we are able to answer today, contact us at:

info@senturus.com

and we will set up a free consultation.



Thank You

www.senturus.com 888 601 6010 info@senturus.com