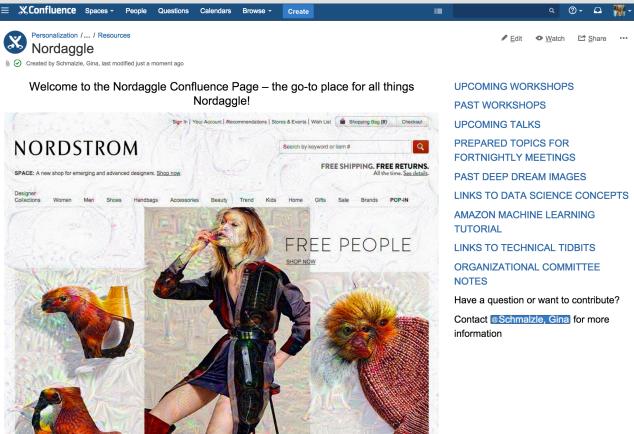
# NORDAGGLE

PRESENTS...

# Introduction to Python Workshop

Caroline Harbitz & Gina Schmalzle

# NORDAGGLE-WHAT IS IT?



**UPCOMING WORKSHOPS** 

PAST DEEP DREAM IMAGES

LINKS TO DATA SCIENCE CONCEPTS

AMAZON MACHINE LEARNING

ORGANIZATIONAL COMMITTEE

Contact @Schmalzle, Gina for more

**Primary purpose** -- increase communication about data topics

**Nordaggle** – spin off from Kaggle

#### We host

- 1. Fortnightly meetings to talk about data topics and problems
- 2. Workshops
- 3. Competitions (not yet offered)

https://confluence.nordstrom.net/display/PER/Nordaggle

# SPECIAL THANKS

Melissa Haklitch − logistics extraordinaire ©

Jason Gowans and the Data Lab – Food and Drink

**Heidi Whiting and James Pestrak** – python/pip installation

Our TA's!

# SPECIAL THANKS

#### You!

- 212 Officially Signed Up
- 105 Beginner, 81 Intermediate and 22 Expert Programmers (4 no response)



Gina Schmalzle & Caroline Harbitz

# Workshop outline

- \* Part 1: Jupyter & Python overview
- \* Part 2: Data types
- \* Part 3: Functions and control structures
- \* Part 4: Reading and writing data

### We have one rule

Ask for help!

# Part 1: Jupyter & Python

#### Outline:

- \* Some history
- \* What is Python?
- \* What's an interpreter? A shell? A script?
- \* What is Jupyter?
- \* Basic commands

# Python history

- \* Created in 1990 by the BDFL
- \* It's not named after snakes
- \* Pretty unpopular until 2.0 was released in 2000
- \* 3.0 was released in 2008
  - \* Backwards incompatibility issues
  - \* Updated scoping rules
  - \* Different string encoding

We're using v2.7 for this workshop

# Programming elements

- \* A **program** or **script** is a sequence of commands
- \* These sequences are evaluated and executed by the Python interpreter in a **shell**
- \* A **command** or **statement** is an instruction for the interpreter

# Jupyter Notebook

Web apps that contain both:

- \* computer code (support for > 40 languages)
- \* rich text elements (images, videos, JavaScript, etc.)

To start a notebook server from your terminal:

\$ jupyter notebook

#### Command types:

\* Terminal: \$

\$ jupyter notebook

\* Python: >>>

# Basic Python Operations

#### Variable assignment:

#### **Tests:**

#### Math:

## Part 2: Data types

- \* Built-in data types
- \* What's a dictionary? A list? A tuple?
- \* Indexing and slicing

#### Write a python script. For example:

- \* Create a file named example.py (note the .py extension)
- \* In example.py write some Python code (e.g.: print("Hello world!")), save, and close.
- \* To run the script, At the command line type:
  - \* python example.py

- \* Use the ipython interactive shell For example:
  - \* At the command line type: ipython
  - \* Start writing python code like you did with notebooks

Try the final projects!

- \* python 5 final\_project.ipynb
- \* python\_6\_final\_project\_with\_pandas.ipynb

- \* Think up your own project.
- \* Tips:
  - \* The Python docs, Google, and Stack Overflow are your friends
  - \* Use print statements to troubleshoot
  - \* Try using a debugger
  - \* <a href="https://wiki.python.org/moin/PythonDebuggingTools">https://wiki.python.org/moin/PythonDebuggingTools</a>