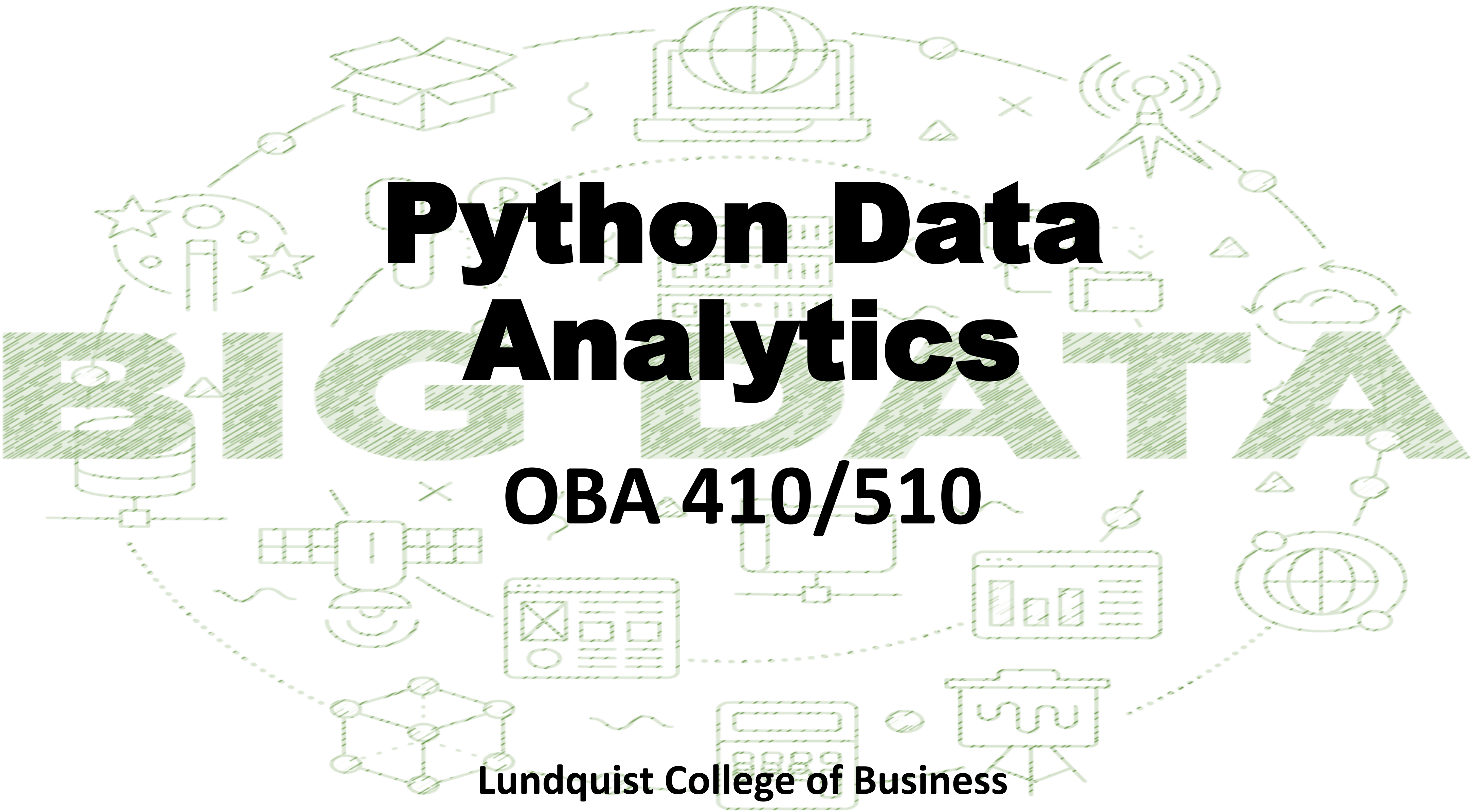


Python Data Analytics

OBA 410/510

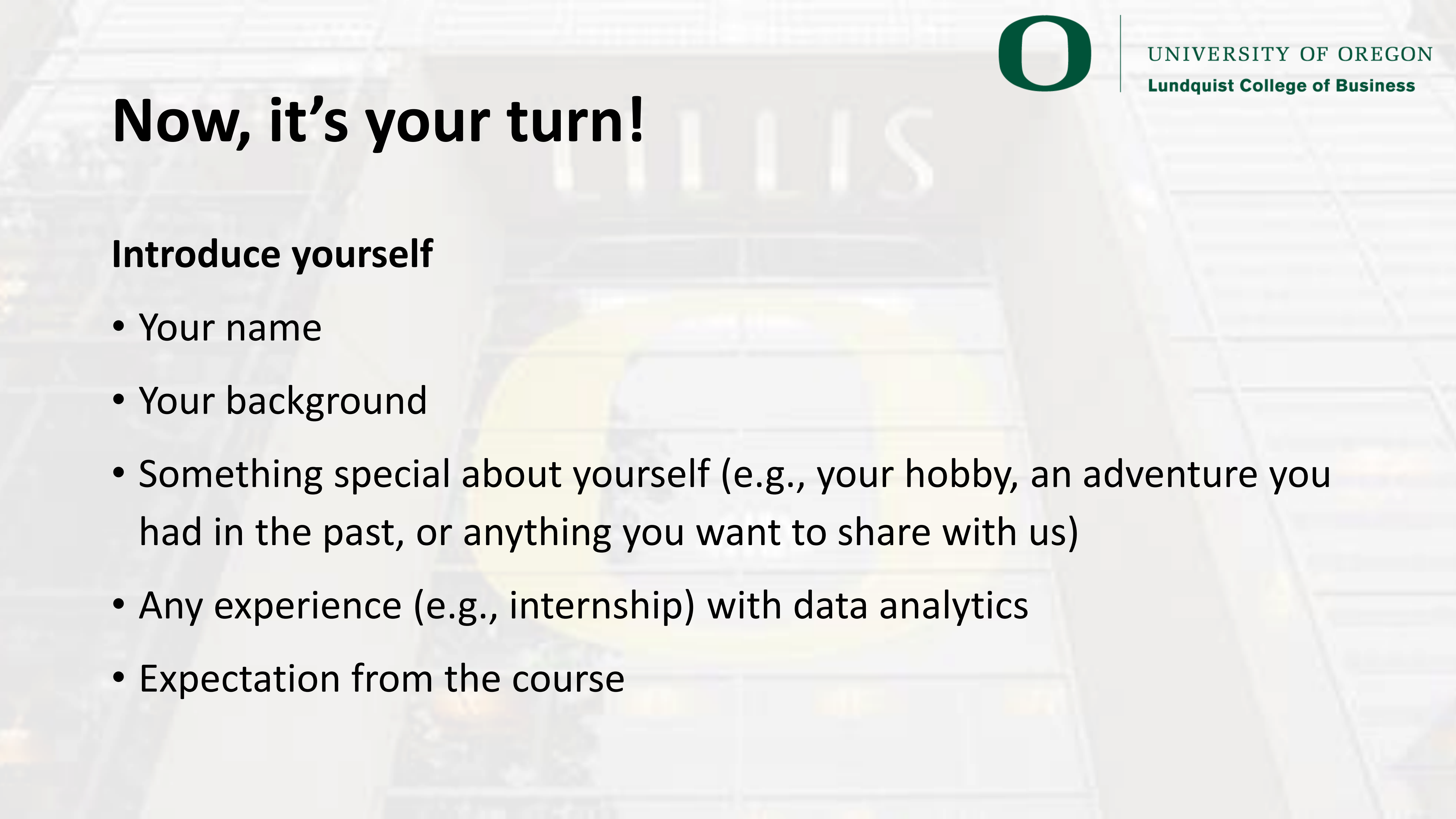
Lundquist College of Business

Background image from www. givweb.com

Some of the slides and contents are adopted from Statistical Learning course by Dr. Trevor Hastie

About me

- Saeed Piri
- My educational degrees are in Industrial Engineering and Management
- Research areas: data analytics, healthcare analytics, and operations analytics
- How can you reach me?
  - E-mail: *spiri@uoregon.edu*
    - Please add **OBA 410/510** and **class time** to the subject of your e-mail.
  - Office hours **(in-person and virtual):**
    - Monday and Wednesday **12:00 PM -1:50 PM** (PST) Or by appointment.
    - Lillis 434
    - You can join via zoom using your **uoregon Zoom account**.
    - Zoom office (meeting ID: 983 1325 6320)
    - Meeting URL: https://uoregon.zoom.us/j/98313256320



Now, it’s your turn!

Introduce yourself

- Your name
- Your background
- Something special about yourself (e.g., your hobby, an adventure you had in the past, or anything you want to share with us)
- Any experience (e.g., internship) with data analytics
- Expectation from the course

Why Data Analytics?

HBR.ORG

Harvard Business Review

OCTOBER 2012
REPRINT R1210D

SPOTLIGHT ON BIG DATA

Data Scientist: The Sexiest Job Of the 21st Century

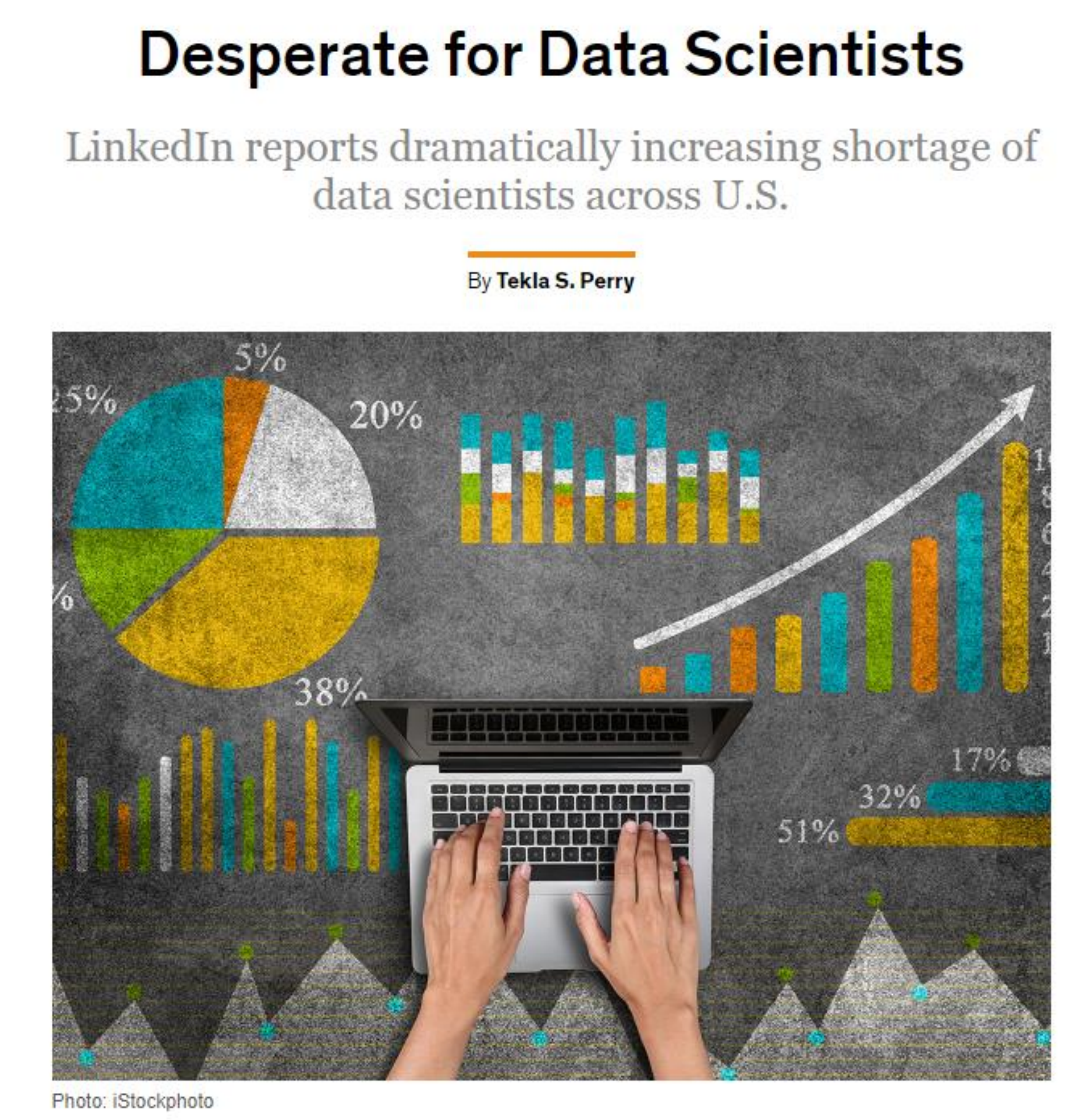
Meet the people who can coax treasure out of messy, unstructured data.
by Thomas H. Davenport and D.J. Patil

McKinsey Global Institute

May 2011

Big data: The next frontier for innovation, competition, and productivity

“The United States alone faces a shortage of 140,000 to 190,000 people with analytical expertise and 1.5 million managers and analysts with the skills to understand and make decisions based on the analysis of big data”



Desperate for Data Scientists

LinkedIn reports dramatically increasing shortage of data scientists across U.S.

By **Tekla S. Perry**

Photo: iStockphoto

"**LinkedIn** says that in August 2018 employers were seeking 151,717 more data scientists than exist in the US".

LinkedIn lists data science in its Emerging Jobs lists

Glassdoor reports:

For the year 2020, Glassdoor named Data Scientist as

- the third most desired job in the United States with
- more than 6500 openings and
- a median base data scientist salary of $107,801

Growing Volume of Data

- Massive data collected every day and is available to take advantage of
  - Every click in the internet is tracked
  - Every patient’s health records are stored
  - Every shopping transaction is stored
  - Smart phones collect so much of information from our behaviors
    - Locations
    - Videos we watch on YouTube
    - Social Media activities, etc.
  - Census collects data about people and economy

Every minute…

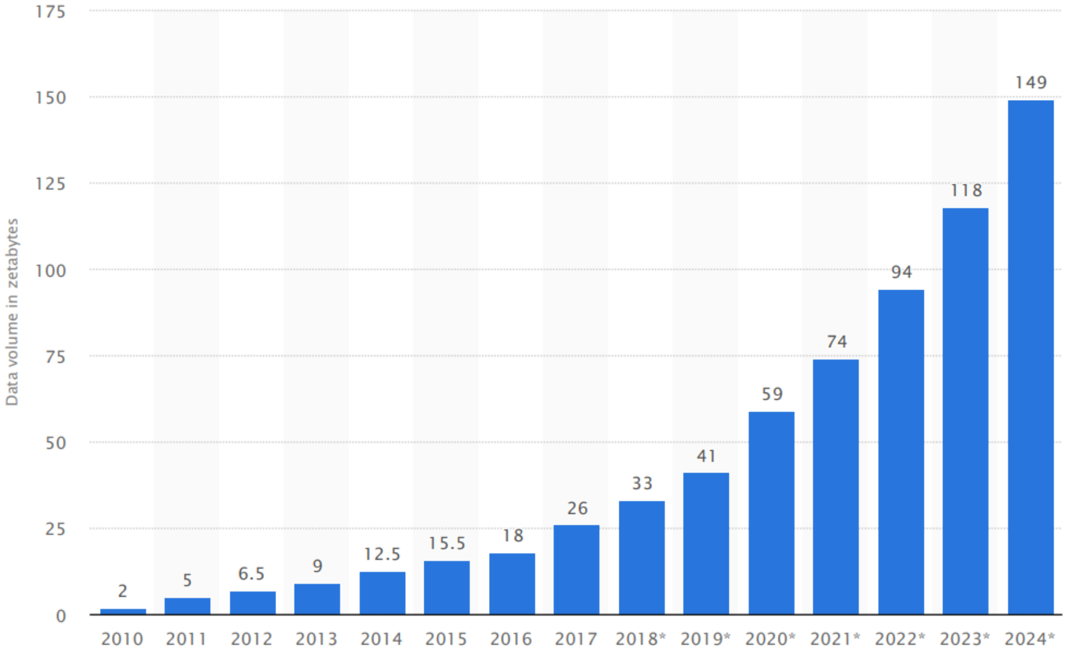
204 Million emails

200,000 photos and 1.8 Million likes on Facebook

1.3 Million video views on YouTube

Volume of data/information created, captured, copied, and consumed worldwide from 2010 to 2024

(in zettabytes)



Sources
IDC; Seagate; Statista estimates

Survey by
IDC; Statista estimates

Published by
IDC; Statista

Details: Worldwide; 2010 to 2020

© Statista 2021

Exabyte=1,000,000,000 GB
Zettabytes= 1,000 Exabyte

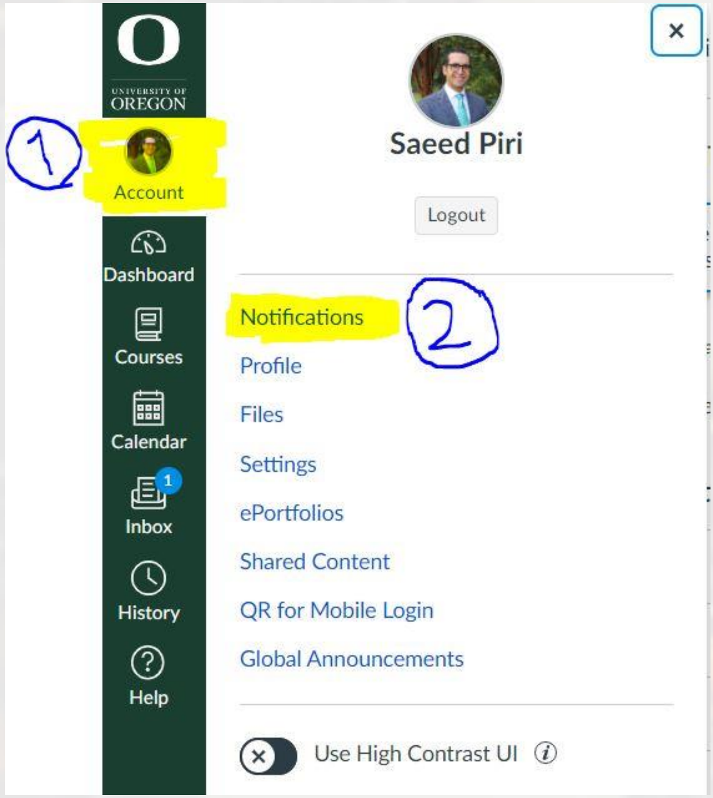
Data Analytics Applications

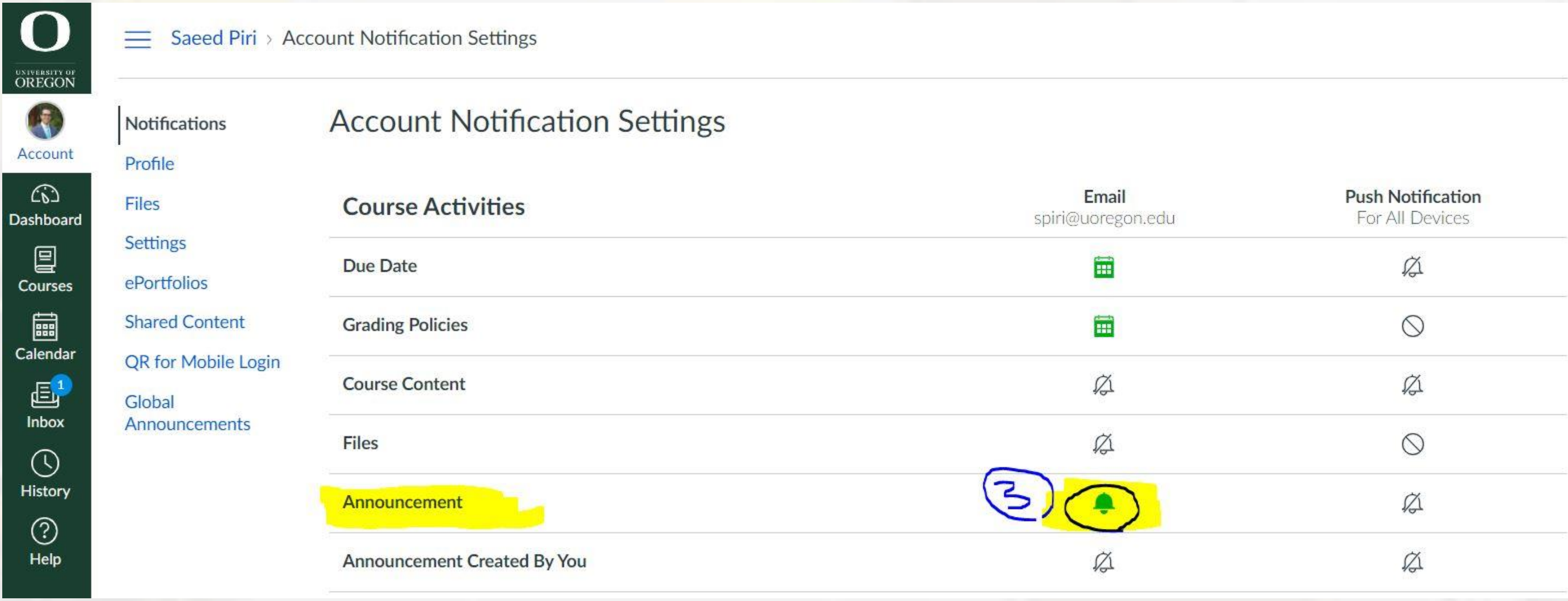
- Nowadays, data analytics plays an important role in various areas of industry and science
- It enables us to make better decisions facing different problems
  - Predict whether a person will have a heart attack
  - Predict whether a patient will survive a surgery
  - Predicting the success rate of a marketing campaign
  - Detecting credit card fraud transactions
  - Predicting the shopping behaviors of our customers
  - Identifying the success factors for a football team
  - Recommending which movies to watch
  - Customize an email spam detection system.
  - And many more applications in government, insurance, marketing, operations, sports, healthcare, e-commerce, media/entertainment and finance fields

About this class…

How I communicate with you

- All of my communications about the course with you will be **through announcements.**
- Make sure to check them regularly
- Activate notifications for announcements (see below for instructions):





Account Notification Settings

Course Activities	Email spiri@uoregon.edu	Push Notification For All Devices
Due Date	📅	🔕
Grading Policies	📅	⊘
Course Content	🔕	🔕
Files	🔕	⊘
Announcement	🔔	🔕
Announcement Created By You	🔕	🔕

Classroom Technology Use

Only on programming time

Textbook

- The following textbooks are **recommended (not required)**:
- **Introduction to Machine Learning with Python,** by Sarah Guido, Andreas Müller
  - http://shop.oreilly.com/ (Links to an external site.)
  - E-version is available on UO Library for **free**:
  - E-book- Introduction to Machine Learning with Python (Links to an external site.)
- **The Elements of Statistical Learning, Data Mining, Inference, and Prediction,** 2nd Edition by Hastie, T., Tibshirani, R., Friedman, J.
  - Available online for **free** at:
  - https://web.stanford.edu/~hastie/ElemStatLearn/ (Links to an external site.)
- These books are not required, and they are both available online for free. While I will provide my own lecture notes throughout the term, these books serve as great compliments of the lecture notes and provide a lot more details.

Assessment

- **In-class Activities - 5%**
  - You participate in in-class activities.
  - For most activities, you will be asked to upload your in-class work (**Jupyter Notebook**) to Canvas **at the end of the class** to receive points based on effort.
  - If you miss a class session, you can still work on the code, and submit your Jupyter notebook **within 24 hours** of the class time. You may do this **only up to 3 times**. Any other late submissions won't be credited.

Assessment

- **Quizzes – 24%**
  - There will be **eight quizzes** throughout the term.
  - You will take the quizzes **during our Class time** on Canvas.
  - Questions will be short-answer, multiple-choice, or true/false, **both coding and conceptual**.
  - Quizzes are open book. However, they **are timed**. Therefore, you must be prepared; otherwise, if you want to go back and forth to your notes, you won't be able to finish on time!
- **Midterm Exam - 33% (First class of Week 8, regular class time)**

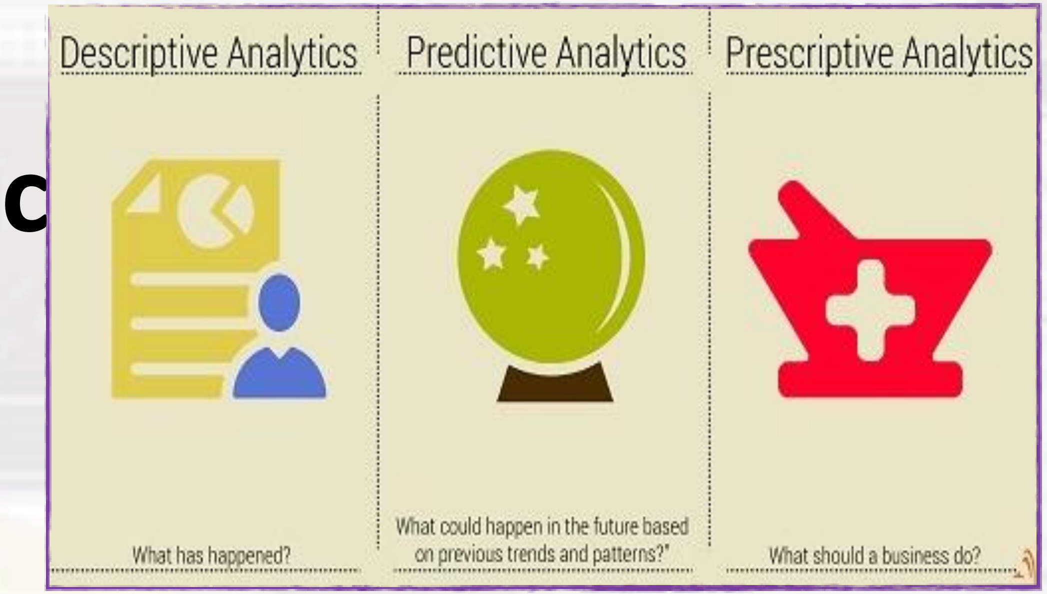
Assessment

- **Term Project-38%**
- Term project will pose a relevant business question, gather and clean data, perform data analysis, and report your results in a detailed write-up
- Group of **three** students
- Deliverables:
  - Proposal presentation (in about 5 minutes) – 10% (during **week 6**)
  - Final Presentation – 10% (during **week 10**)
  - Final Report – 18% (**by 8:00 am, Monday of final week**)- Absolutely no late submission will be accepted

Term Project

- **Peer-Assessment-** Team Project Grading Adjustments:
  - Should you not contribute to a team assignment, your team is instructed not to include your name on the assignment. If this occurs, you will receive a 0 for that assignment.
  - Furthermore, your overall grade on the team project components of the class (Proposal, Final Presentation & Final Report) is **subject to adjustments either up or down based on peer evaluations** from your teammates as reported in and calculated from the Peer Evaluations form

3 Types of Business Analytic


Descriptive Analytics
What has happened?
Predictive Analytics
What could happen in the future based on previous trends and patterns?"
Prescriptive Analytics
What should a business do?

- ***Descriptive Analytics:***
  - Understanding past and current business performance and make informed decisions
- ***Predictive Analytics:***
  - Predicting the future by examining historical data, detecting patterns or relationships
- ***Prescriptive:***
  - Identifying the best alternatives.

This Course…

- This course is about extracting knowledge from data
- This course is about learning how to develop predicting models
- What are the tools we can use in data mining?
  - Python
  - R
  - SAS
  - SPSS, etc.
- Which one we will learn and use in this course?

Data Analytics

- Two examples:
  - Predicting whether a hospitalized patient will survive based on the demographic and clinical information (Age, gender, blood pressure, heart rate, …)
  - Predicting a house/property price based on its characteristics such as lot size, size, number of rooms, locations, etc.
- Inputs/predictors/independent variables/features
- Output/response variable/dependent variable/target variable
- predict the output for any given input set

Two types of learning

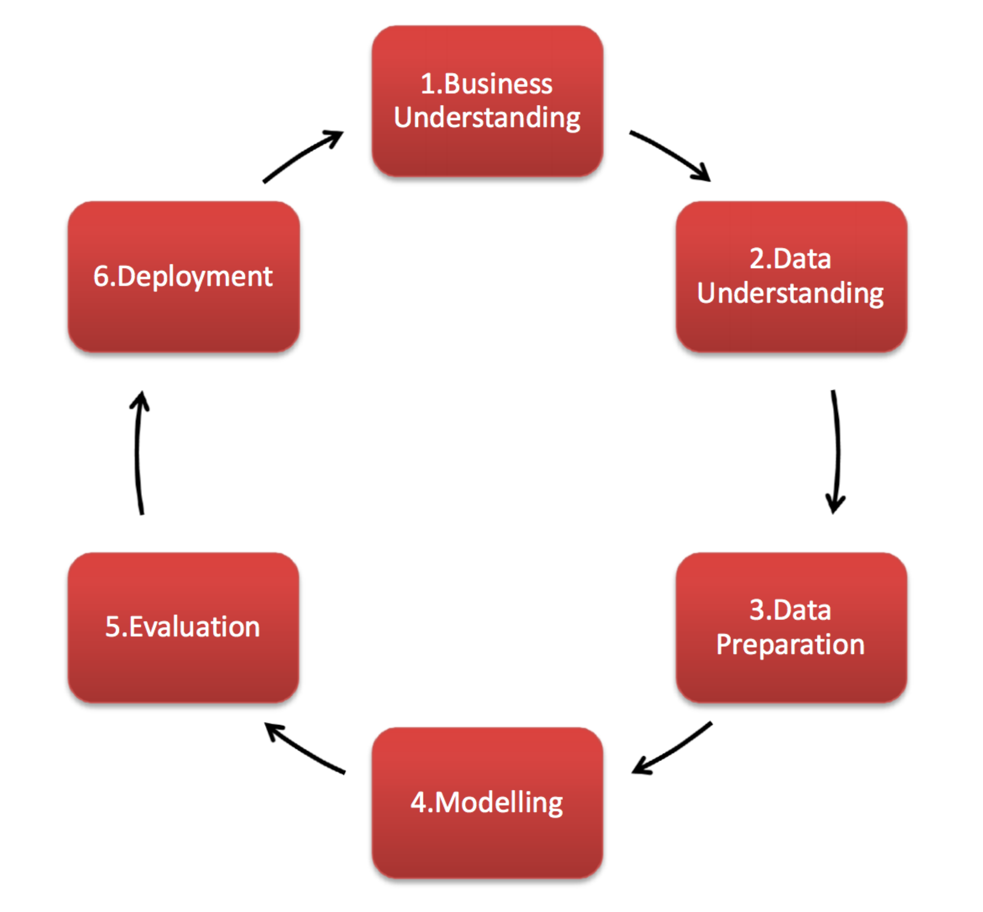
- **Supervised machine learning (predictive analytics)**
  - When the output is present in the learning process (the algorithm is provided with pairs of inputs and output)
- Unsupervised learning
  - When the output is not present in the learning process
  - Examples: customers segmentation, identifying topics in blog posts

Data Analytics Objectives

- On the basis of the training data we would like to:
  - Accurately predict unseen test cases.
  - Understand which inputs affect the outcome, and how.
  - Assess the quality of our predictions and inferences.

Data Mining process

- CRISP-DM methodology
- CRISP-DM:
  - cross-industry process for data mining



Knowing the Data and Task

- The most important part of a data analytics project is knowing the data and business problem
- In any *comprehensive data analytics* study, more than 80% of the total project time is spent for data cleaning and preprocessing.
- This could include:
  - access
  - filter
  - transform
  - manage
  - impute
  - store and retrieve

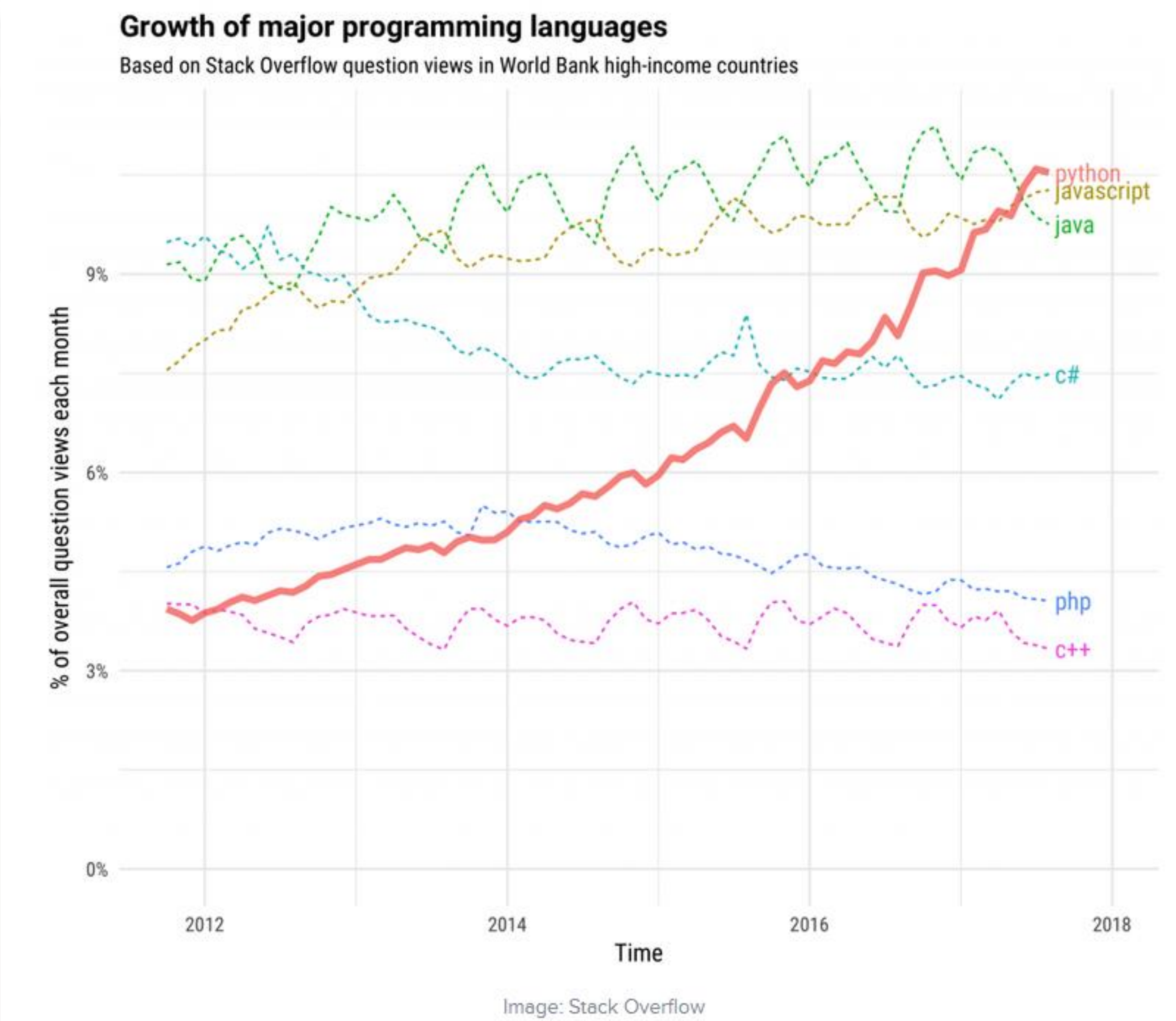
python
TM

Why Python

- Python is the fastest-growing programming language, according to the Stack Overflow community.
- It is likely Python will significantly outstrip other coding languages in terms of popularity.
- Its flexibility and relative ease of use are some of the top reasons that make Python stand out from the crowd.

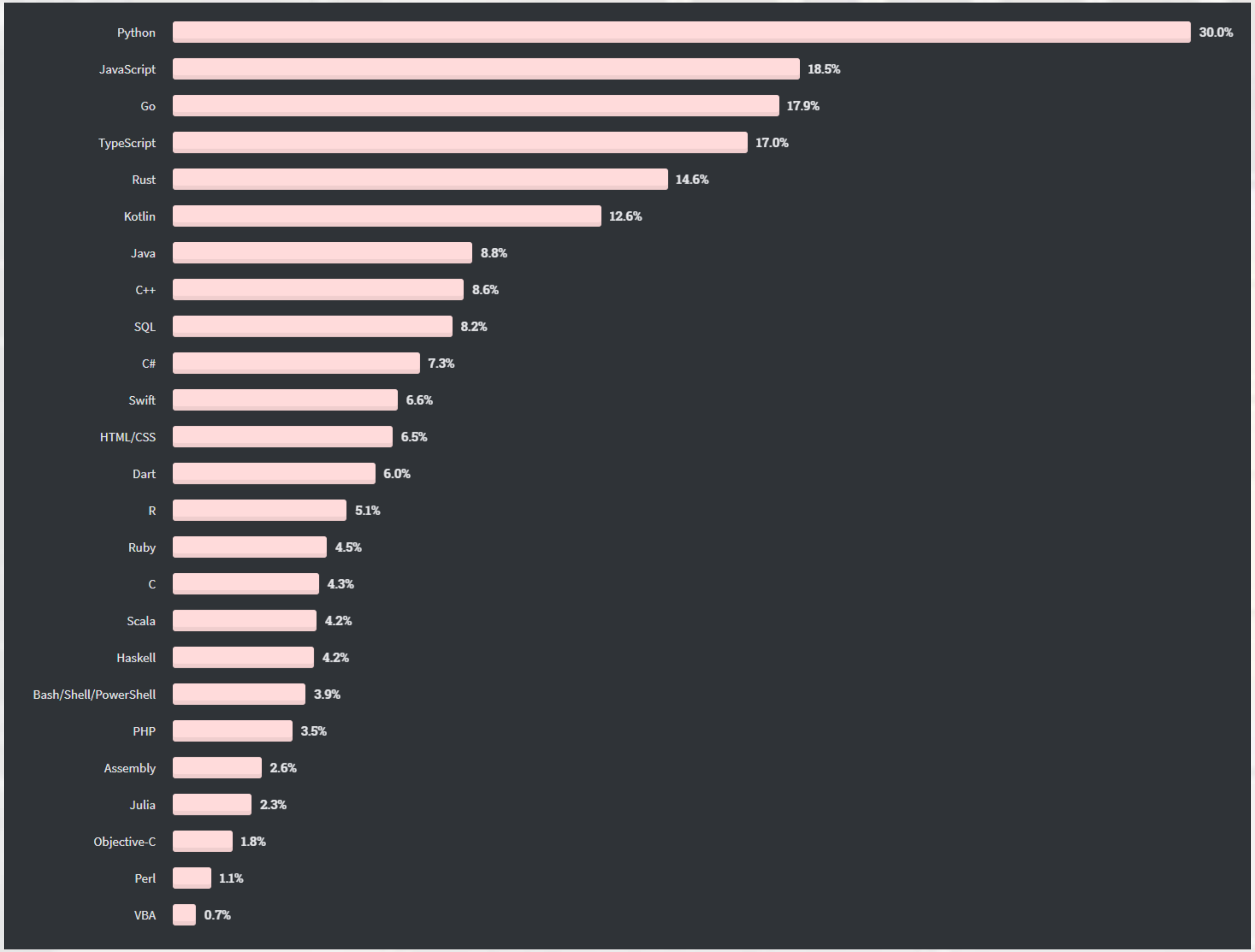
- A very good general resource for beginners
  - https://opentechschool.github.io/python-beginners/en/getting_started.html#what-is-python-exactly

https://towardsdatascience.com/what-are-the-skills-needed-to-become-a-data-scientist-in-2018-d037012f1db2



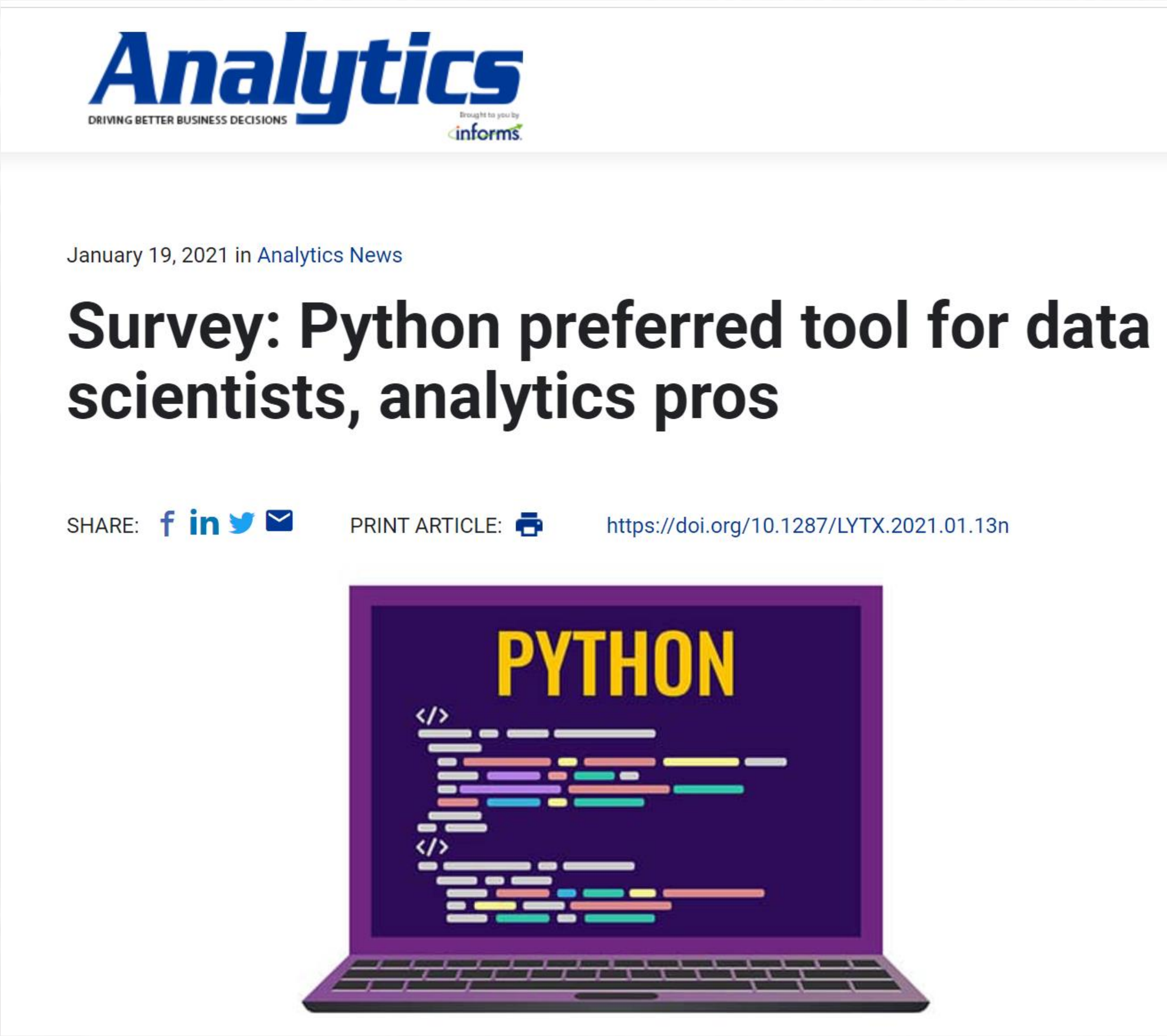
https://stackoverflow.blog/2017/09/06/incredible-growth-python/

Most Wanted Languages



Stack Overflow Developer Survey 2020

In 2020 nearly 65,000 developers told us how they learn and level up, which tools they're using, and what they want.



Analytics
DRIVING BETTER BUSINESS DECISIONS
Brought to you by informs

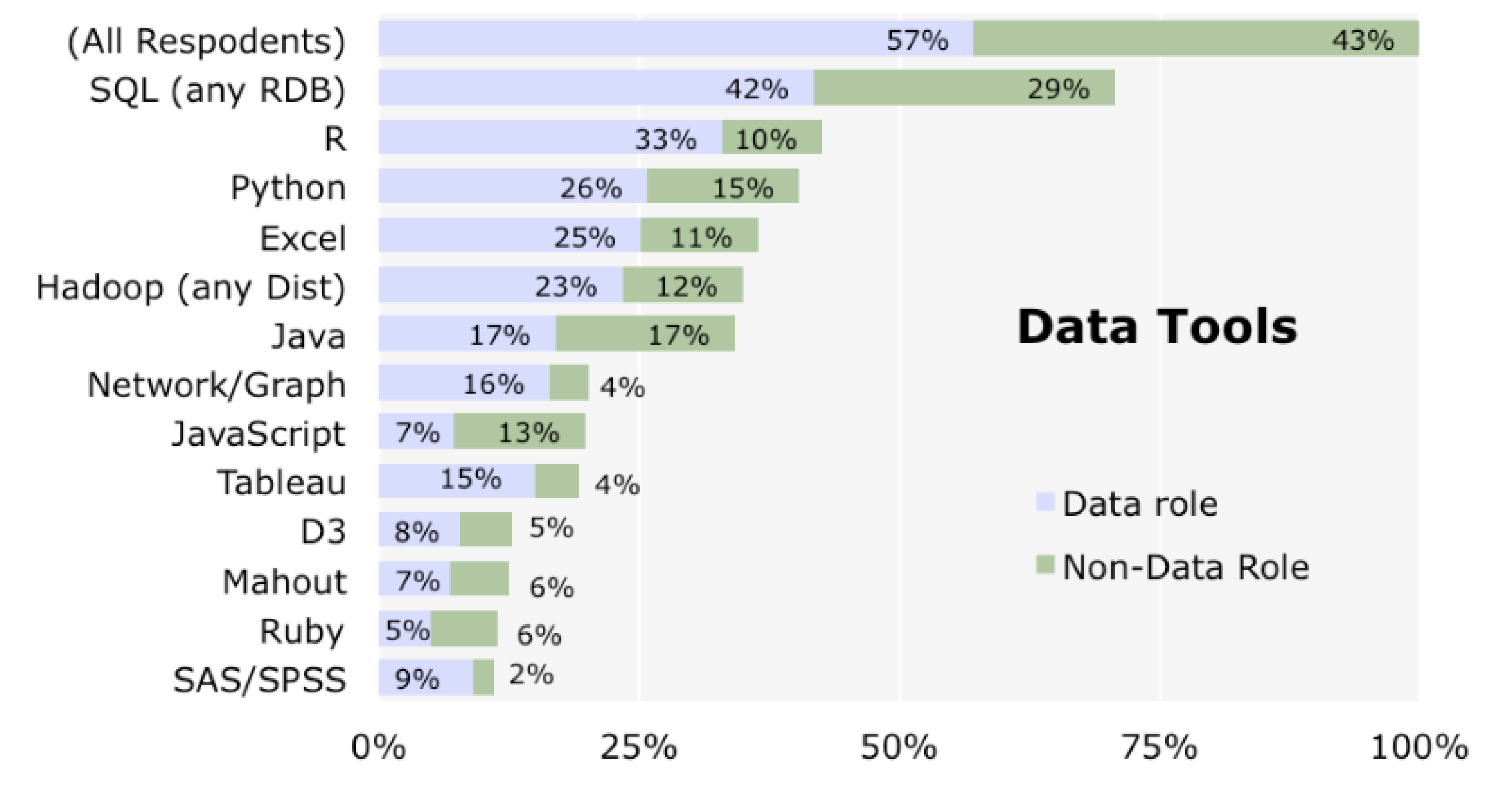
January 19, 2021 in Analytics News

Survey: Python preferred tool for data scientists, analytics pros

SHARE: f in ✉ PRINT ARTICLE: 🖶 https://doi.org/10.1287/LYTX.2021.01.13n

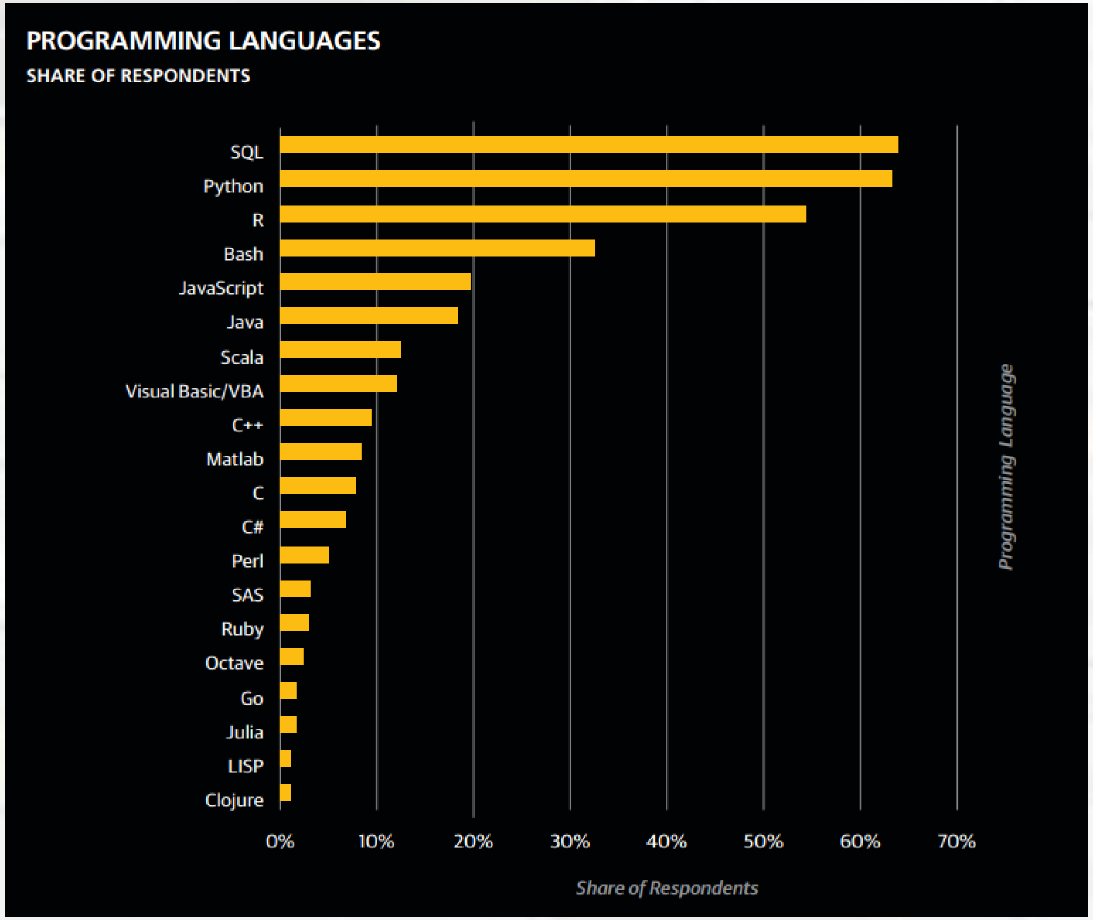
- https://pubsonline.informs.org/do/10.1287/LYTX.2021.01.13n/full/

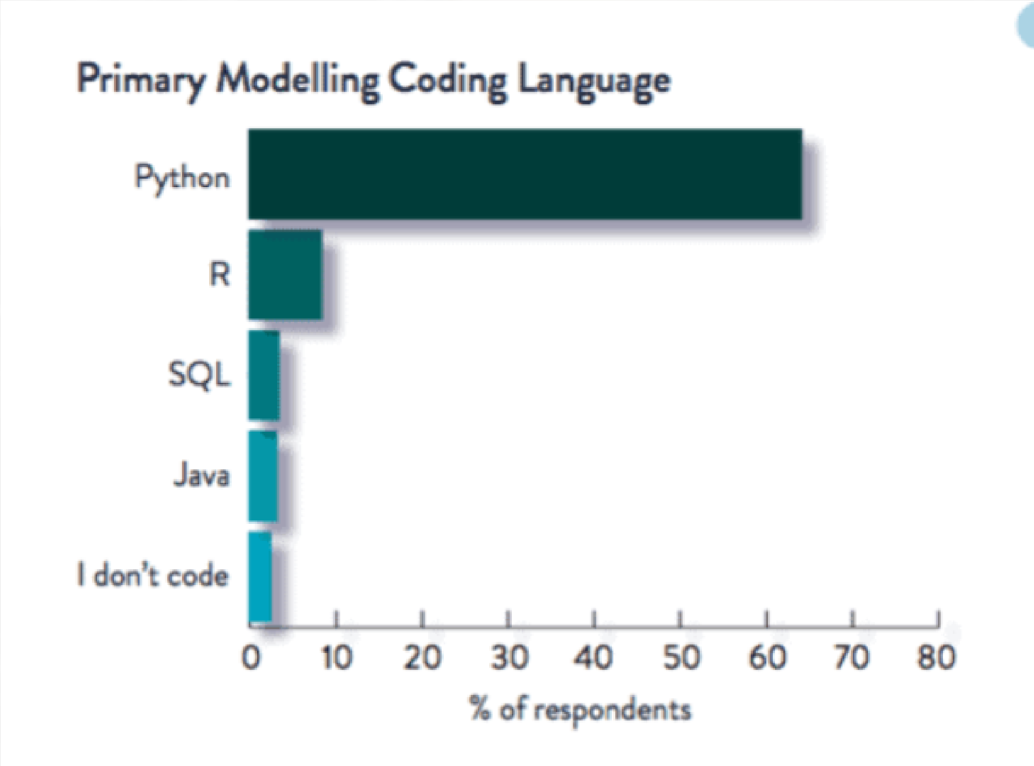
O’Reilly Data Science Salary Survey 2013

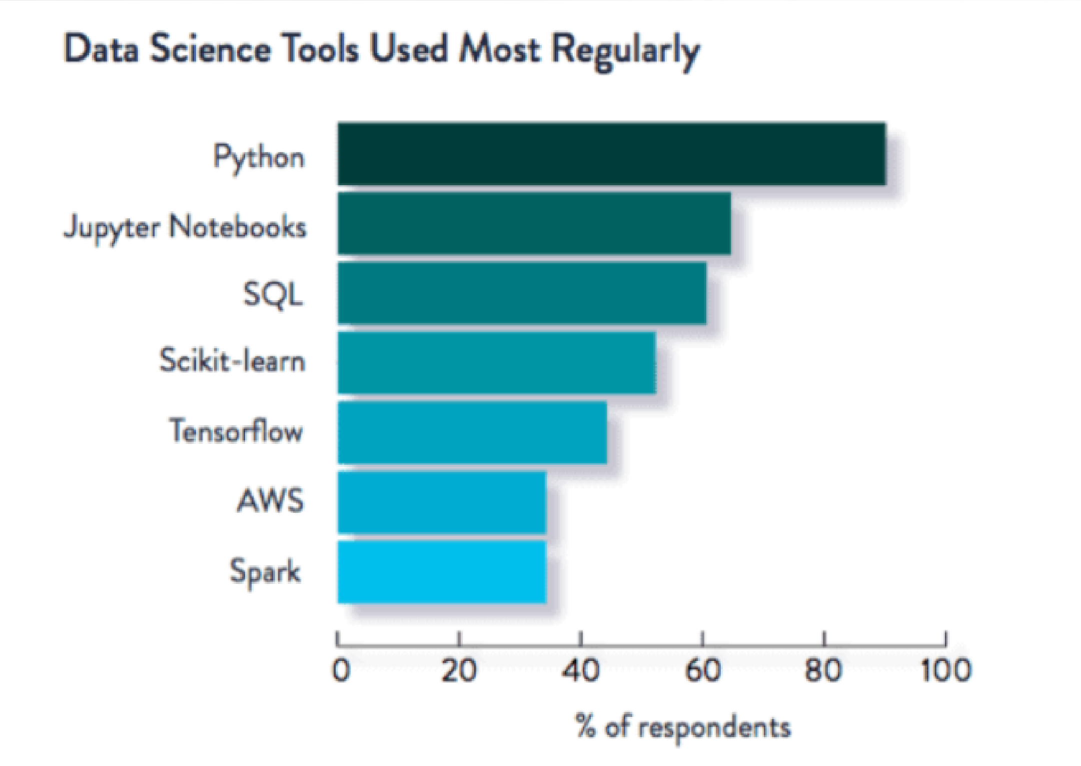


O’Reilly Data Science Salary Survey 2017

- Python users jumped from 58% in 2016 to 63% in 2017
- 63% are using Python versus 54% R users



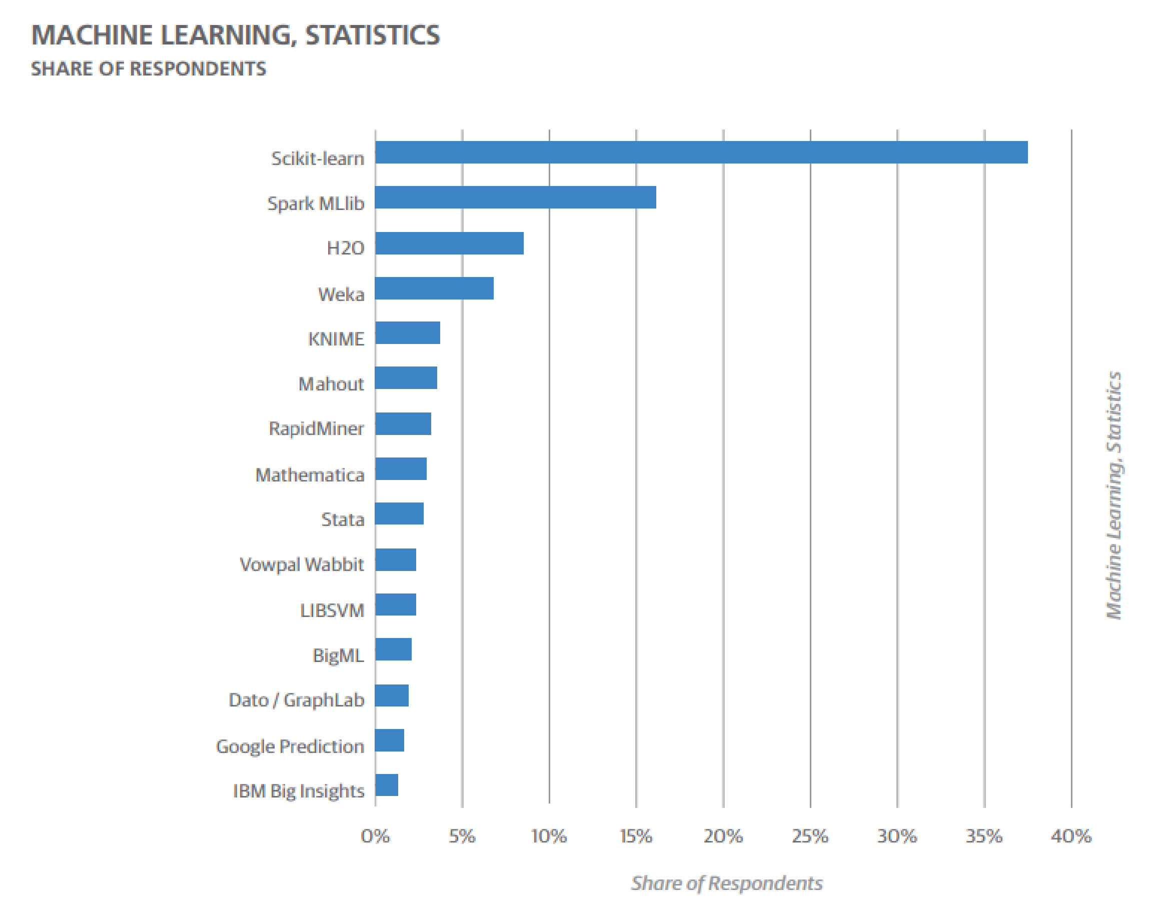




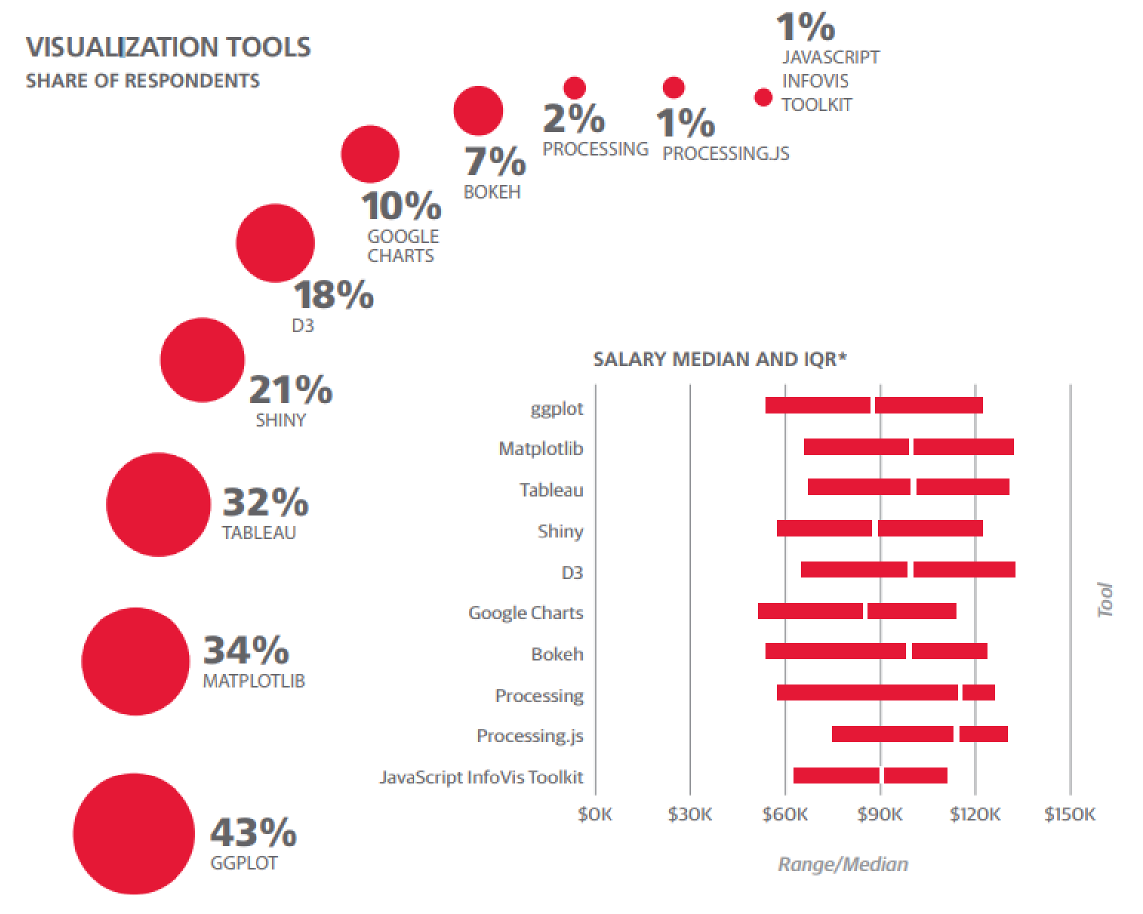
Data Science Salary Report 2020 Europe by Big Cloud & O'Reilly Data Science Salary Survey

UNIVERSITY OF OREGON
Lundquist College of Business

Machine Learning Tools



Data Visualization Tools



About Python

- Created by Guido van Rossum in the late 1980’s
- Python combines the power of general-purpose programming languages with ease of use of domain-specific scripting languages like MATLAB and R.
- Easy to learn
- Easy to interact with

How to interact with Python

- Through Python’s interpreter (Python shell) using your operating system’s (OS) console (command prompt in Widows or terminal in Mac)
  - Open the system console
  - Type python (if you have added python to your computer PATH when you installed Python)
  - Now you should be in Python shell (interpreter), you will see >>> prompt on the last line
  - To leave the Python shell and go back to the console, press Ctrl-Z and then Enter on Windows (or Ctrl-D on OS X or Linux), you could also run the python command exit()
  - Not the best way for larger codes, instead we can use:
- IEDs (Integrated Development Environments) or Code Editors
  - Various IEDs and code editors available for Python
  - In this course we use **Jupyter Notebook**

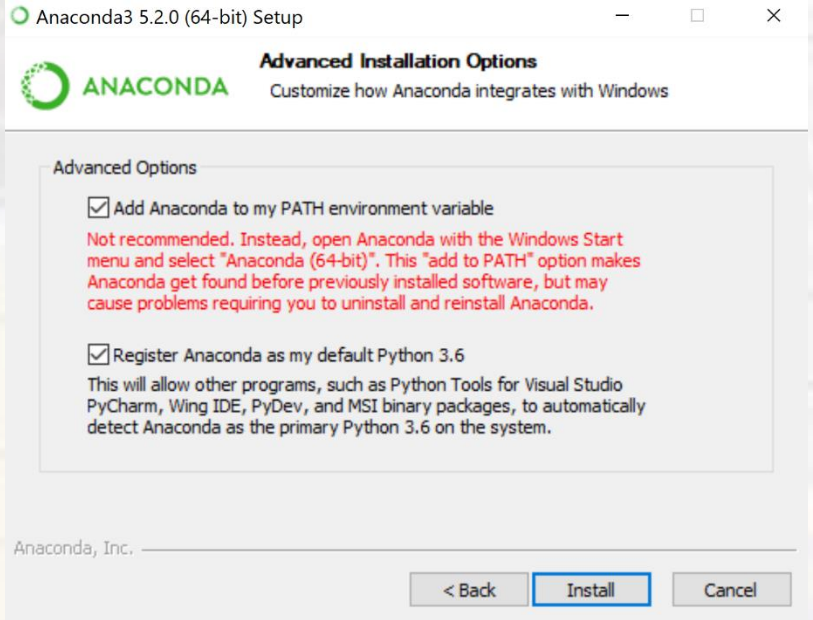
Installing Jupyter Notebook

- Complete installation information in this link:
  - https://jupyter.readthedocs.io/en/latest/install.html#install
- Use **Anaconda** distribution to install both Python and Jupyter
- This distribution automatically install most of the commonly used data analytics packages
- Use the following installation steps:
  - Download Anaconda’s latest Python 3 version
    - https://www.anaconda.com/download/

Installing Jupyter Notebook

- Follow the instructions and install Anaconda
- Make sure to check the option

  that adds Anaconda to your PATH



Anaconda3 5.2.0 (64-bit) Setup

Advanced Installation Options

Customize how Anaconda integrates with Windows

Advanced Options

☑ Add Anaconda to my PATH environment variable

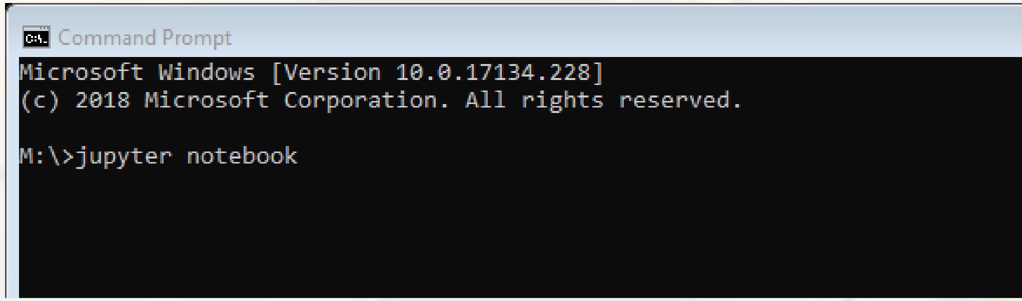
Not recommended. Instead, open Anaconda with the Windows Start menu and select "Anaconda (64-bit)". This "add to PATH" option makes Anaconda get found before previously installed software, but may cause problems requiring you to uninstall and reinstall Anaconda.

☑ Register Anaconda as my default Python 3.6

This will allow other programs, such as Python Tools for Visual Studio PyCharm, Wing IDE, PyDev, and MSI binary packages, to automatically detect Anaconda as the primary Python 3.6 on the system.

- To run the notebook use

  this command (in your console):
  - jupyter notebook, or
  - **Search for jupyter notebook from the start (Windows), or**
  - **Launch jupyter from Anaconda navigator (specially Mac users)**



Command Prompt

```
Microsoft Windows [Version 10.0.17134.228]
(c) 2018 Microsoft Corporation. All rights reserved.

M:\>jupyter notebook
```

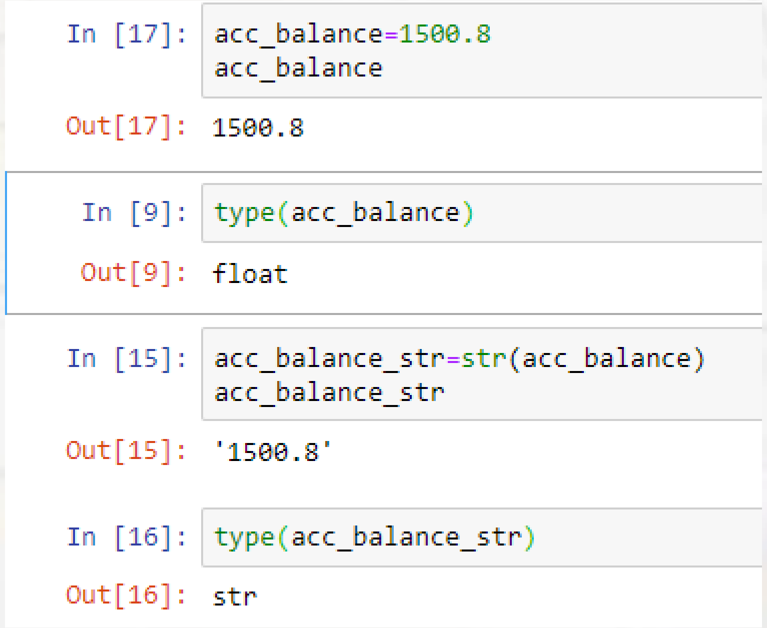
Variables, Types and Values

- Variable names
  - Can be any length
  - Can consist of
    - uppercase and lowercase letters (A-Z, a-z)
    - digits (0-9), and
    - the underscore character (_)
  - First character cannot be a digit (number)
  - Are case sensitive
  - Python 3 has 35 reserved keywords
- Main types used in data science
  - Boolean (bool)… represents logical values can be either *True* or *False*
  - Integer (int) integer numbers
  - String (str)
  - Float (float) numbers with decimals

Type conversion

- Checking type of the variable: `type()`
- str(), converts a value into a string
- Similar functions such as int(), float() and bool()

  convert Python values into any type.



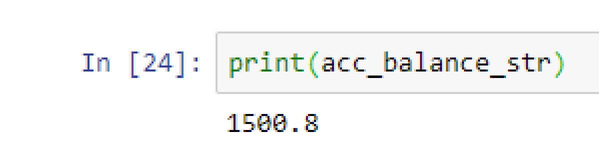
```
In [17]: acc_balance=1500.8
         acc_balance

Out[17]: 1500.8

In [9]:  type(acc_balance)

Out[9]:  float

In [15]: acc_balance_str=str(acc_balance)
         acc_balance_str

Out[15]: '1500.8'

In [16]: type(acc_balance_str)

Out[16]: str
```

Operators

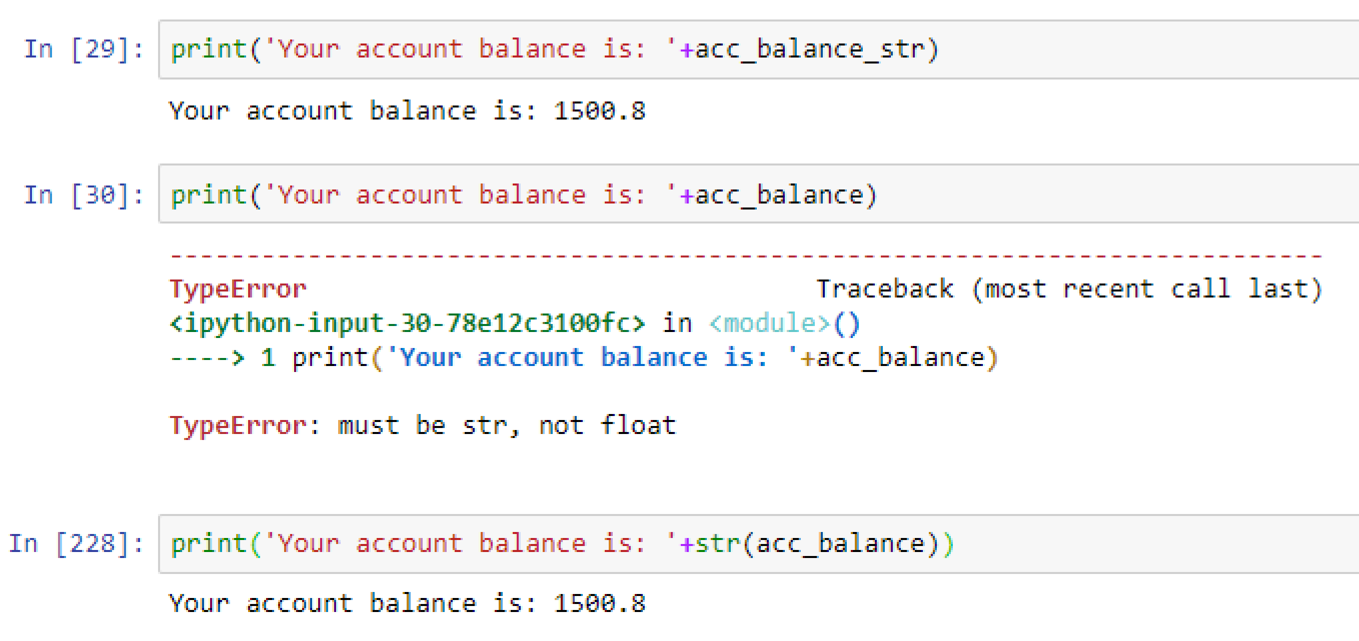
- Python supports the following operators on numbers.
  - + addition
  - - subtraction
  - * multiplication
  - / division
  - ** exponent
  - % remainder, or modulo
- Usual order of operations:
  - Parentheses ()
  - Exponentiation **
  - Multiplication/Division/Remainder
  - Addition/Subtraction

Functions

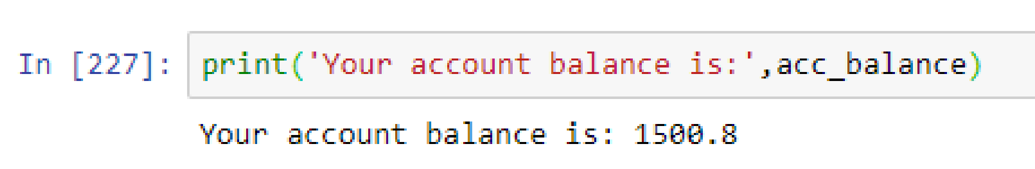
- Python has a couple of built-in functions
- For example: *type(), str(), int(), bool() and float()*
- Another simple built-in function is *print()*



```
In [24]:  print(acc_balance_str)

          1500.8
```



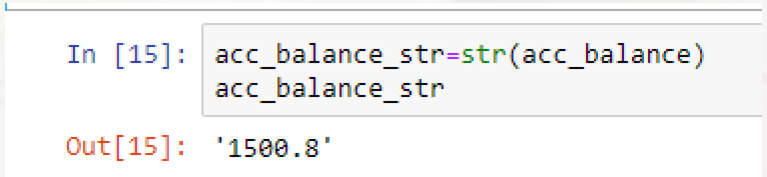
```
In [29]:  print('Your account balance is: '+acc_balance_str)

          Your account balance is: 1500.8

In [30]:  print('Your account balance is: '+acc_balance)

          ---------------------------------------------------------------------------
          TypeError                                 Traceback (most recent call last)
          <ipython-input-30-78e12c3100fc> in <module>()
          ----> 1 print('Your account balance is: '+acc_balance)

          TypeError: must be str, not float

In [228]: print('Your account balance is: '+str(acc_balance))

          Your account balance is: 1500.8
```



```
In [227]: print('Your account balance is:',acc_balance)

          Your account balance is: 1500.8
```

Functions

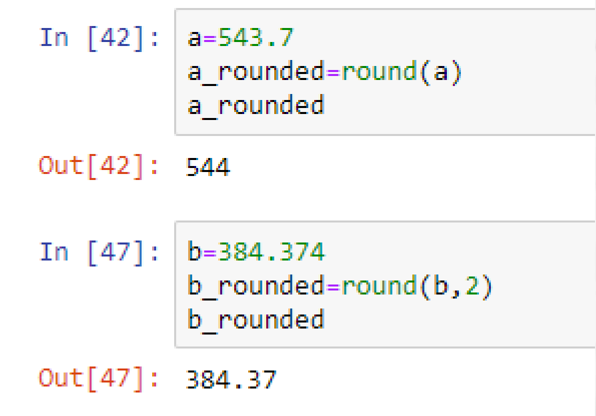
- The general format for calling functions and saving the results into a new variable:
  - new_variable = function_name(input)



```
In [15]: acc_balance_str=str(acc_balance)
         acc_balance_str

Out[15]: '1500.8'
```

- Other built-in functions:
  - *abs(x)* …. returns the absolute value of a number.
  - *max(arg1, arg2,..)* … returns the largest item in an iterable or the largest of two or more arguments.
  - *round(number[, ndigits])* … returns number rounded

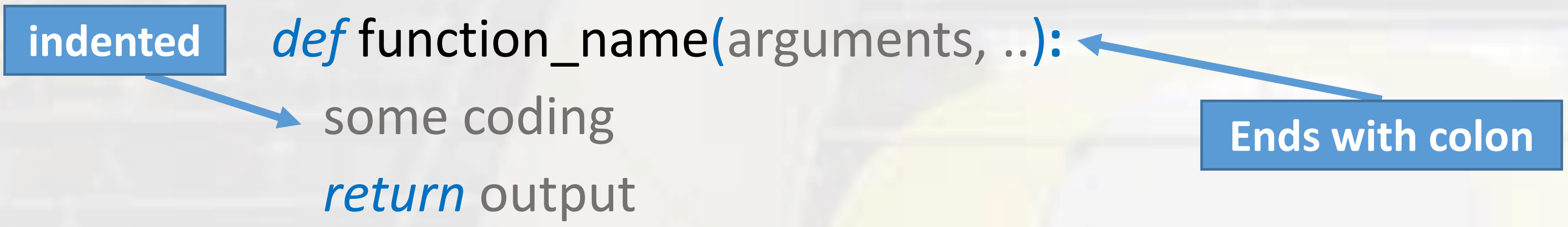
 to ndigits precision after the decimal point. If ndigits is omitted or is None, it returns the nearest integer to its input.



```
In [42]: a=543.7
         a_rounded=round(a)
         a_rounded

Out[42]: 544

In [47]: b=384.374
         b_rounded=round(b,2)
         b_rounded

Out[47]: 384.37
```

Defining new Functions

- We can define new functions using the following format:

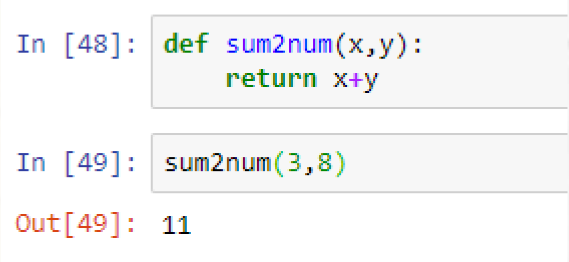


```
def function_name(arguments, ..):
    some coding
    return output
```

indented

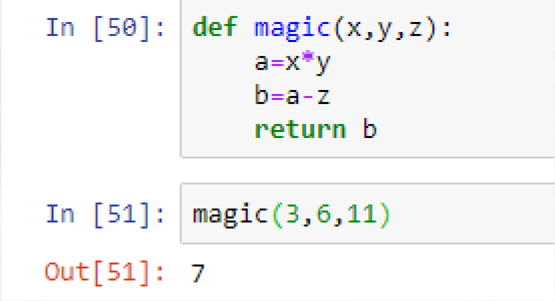
Ends with colon

- Defining a function that sums up two numbers:



```
In [48]: def sum2num(x,y):
             return x+y

In [49]: sum2num(3,8)

Out[49]: 11
```

- Defining a function that takes three numbers, multiplies the first two, and subtract the third one form it:

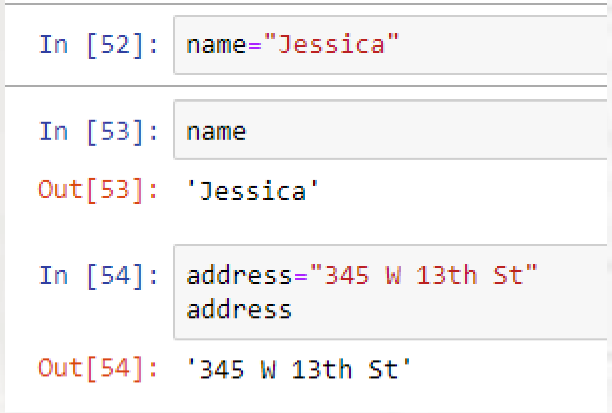


```
In [50]: def magic(x,y,z):
             a=x*y
             b=a-z
             return b

In [51]: magic(3,6,11)

Out[51]: 7
```

- Declaring & Using Functions

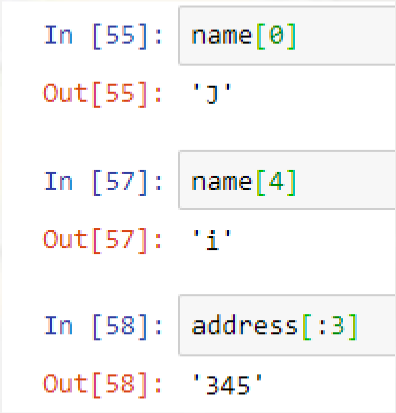
Strings

- Objects that contain sequences of character data



```
In [52]: name="Jessica"

In [53]: name

Out[53]: 'Jessica'

In [54]: address="345 W 13th St"
         address

Out[54]: '345 W 13th St'
```

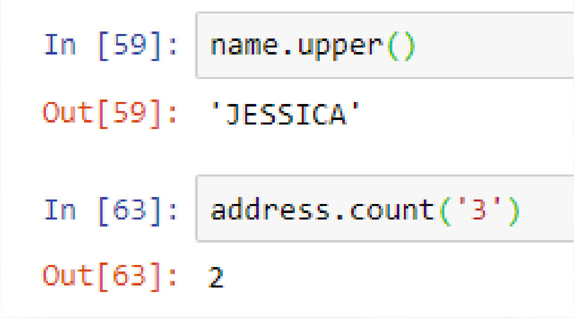
- We can access individual characters in a string using the letter's index
    - <u>first character having an index value of **0**</u>



```
In [55]: name[0]

Out[55]: 'J'

In [57]: name[4]

Out[57]: 'i'

In [58]: address[:3]

Out[58]: '345'
```

Methods

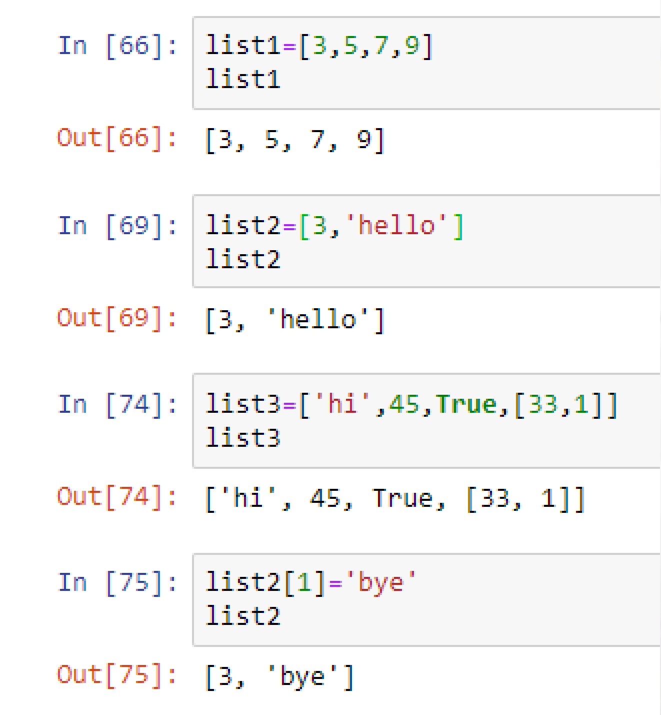
- Functions that belong to objects
- For instance *string* is an object
  - Therefore, there are methods for *strings*
- For instance*:*
- *upper()*
  - Return a copy of the string converted to uppercase.
- *count(sub)*
  - Return the number of occurrences of substring sub



```
In [59]: name.upper()
Out[59]: 'JESSICA'

In [63]: address.count('3')
Out[63]: 2
```

Lists

- Lists are a sequence of values that is similar to a string
- Differences with strings:
  - list is a sequence of any type, while a string is only a sequence of characters
  - lists are mutable and we can change elements
- We create lists using square brackets
- Lists can even contain other lists as an element

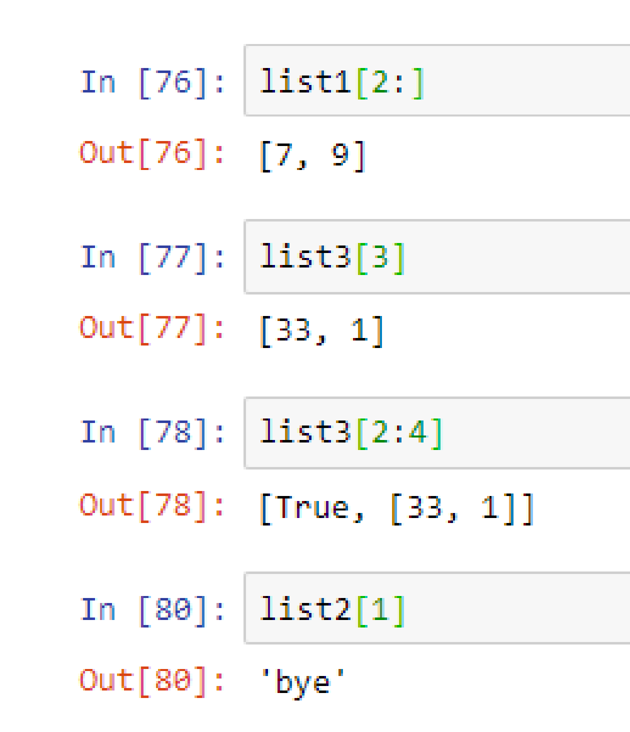


```
In [66]:  list1=[3,5,7,9]
          list1

Out[66]:  [3, 5, 7, 9]

In [69]:  list2=[3,'hello']
          list2

Out[69]:  [3, 'hello']

In [74]:  list3=['hi',45,True,[33,1]]
          list3

Out[74]:  ['hi', 45, True, [33, 1]]

In [75]:  list2[1]='bye'
          list2

Out[75]:  [3, 'bye']
```

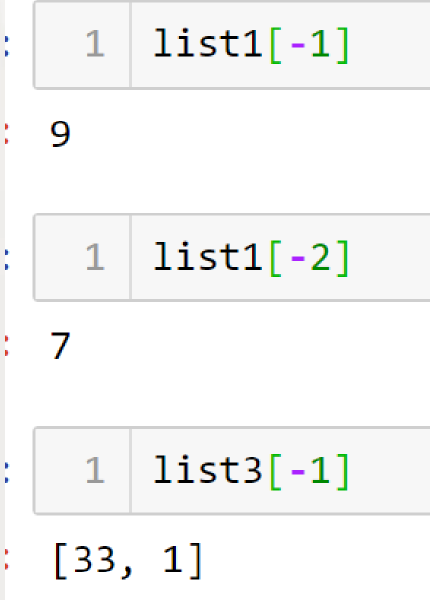
Slicing and dicing lists

- We use indexes similar to strings

- We can also index from the end (right)
- The last element has an index of -1



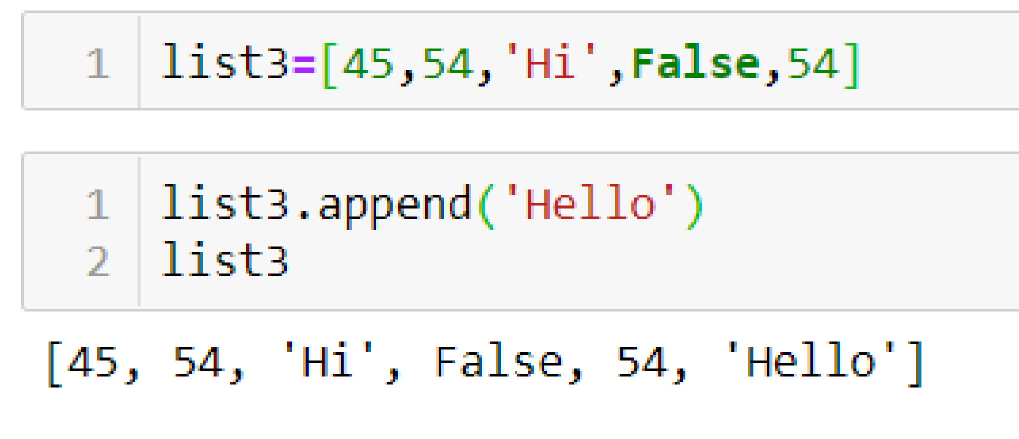
```
In [76]: list1[2:]
Out[76]: [7, 9]

In [77]: list3[3]
Out[77]: [33, 1]

In [78]: list3[2:4]
Out[78]: [True, [33, 1]]

In [80]: list2[1]
Out[80]: 'bye'
```



```
list1[-1]
9

list1[-2]
7

list3[-1]
[33, 1]
```

Adding/removing elements to/from lists

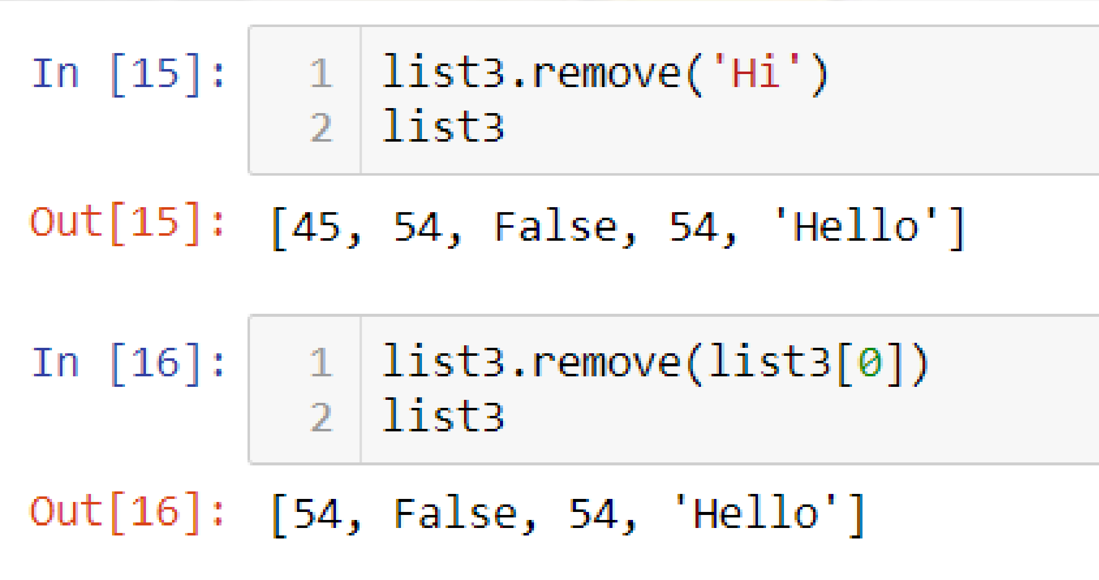
- Listname.append(new elements)
- Listname.remove(elements)



```
list3=[45,54,'Hi',False,54]
```

```
list3.append('Hello')
list3
```

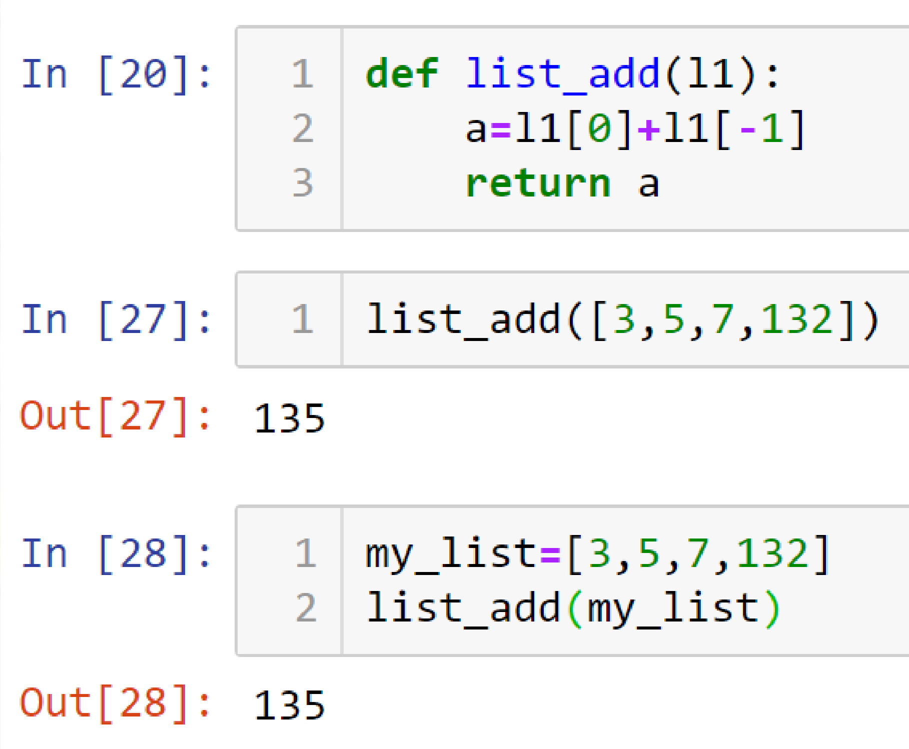
```
[45, 54, 'Hi', False, 54, 'Hello']
```



```
In [15]:  list3.remove('Hi')
          list3

Out[15]:  [45, 54, False, 54, 'Hello']

In [16]:  list3.remove(list3[0])
          list3

Out[16]:  [54, False, 54, 'Hello']
```

Defining new Functions-Example

- Defining a function that takes a list of numbers as input and adds up the first and last elements of the list



```
In [20]:  def list_add(l1):
               a=l1[0]+l1[-1]
               return a

In [27]:  list_add([3,5,7,132])

Out[27]:  135

In [28]:  my_list=[3,5,7,132]
          list_add(my_list)

Out[28]:  135
```