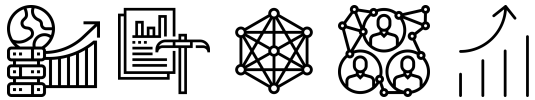


# CSC 5741
# Lecture 2: Python for Data Mining and Machine Learning

Lighton Phiri <lighton.phiri@unza.zm>
Department of Library and Information Science
University of Zambia

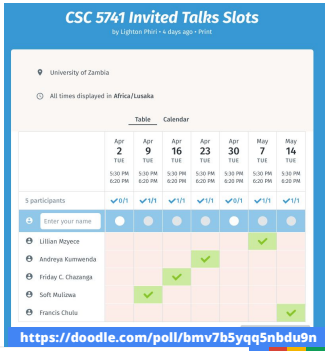
## Announcements—March 26, 2019

- CSC 5741 Moodle site up and running
  - All assignments will be available and submitted via The Moodle
- Mini Projects
  - Open date: March 29, 2019
  - Due date: May 17, 2019
- Paper reading #01
  - Open date: March 29, 2019
  - Due date: April 5, 2019
  - Paper suggestion rotation—Two every week
- Invited Academic Talk
  - April 2, 2019—Lillian Mzyece



## Announcements—April 2, 2019 (1/3)

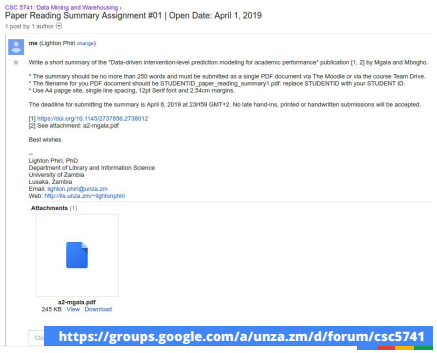
- No talk today—April 2, 2019—as none of the invited guests are able to show up
  - I could give another talk if people wish :P
- Mulizwa will come through to give a talk on April 9, 2019—next week
  - Biography & abstract will be sent
  - Please show up on time and ask a lot of questions





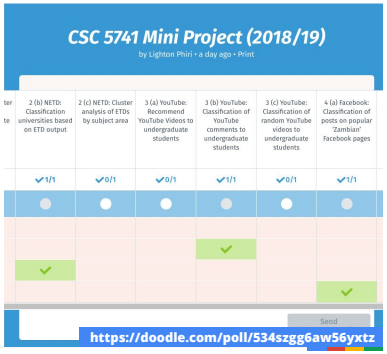

## Announcements—April 2, 2019 (2/3)

- Paper reading summaries are due on April 8, 2019
  - Please follow the reading assignment specifications



## Announcements—April 2, 2019 (3/3)

- **Please make your selections as soon as possible**
  - You might want to make selections to work in packs
- **Errata**
  - Distribution of marks for presentation was less than 100%
  - Page limit for technical report is four (4) pages not six (6) pages



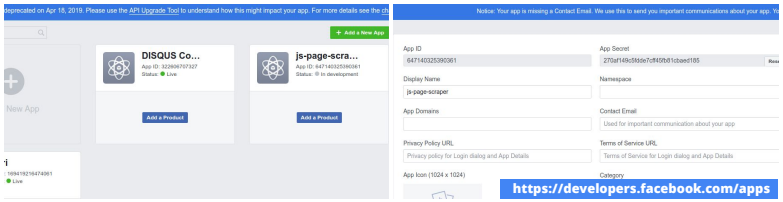



## Todos | Showcases—April 2, 2019 (1/6)


- **Extraction of records of NETD and NDLTD Union Catalog is best done using the OAI-PMH protocol**
  - Use the 'resumptionToken' to loop through batches of records



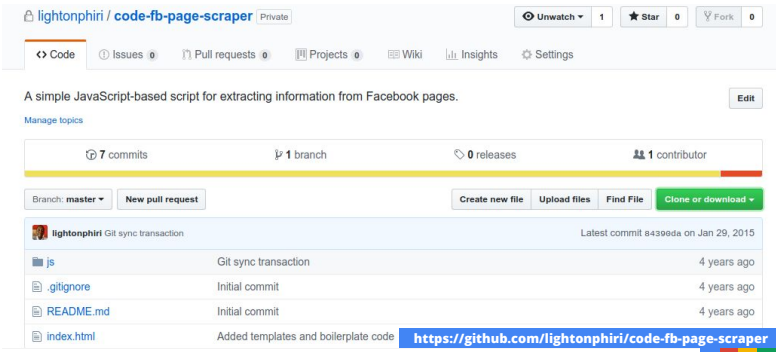
## Todos | Showcases—April 2, 2019 (2/6)




- **Extraction of data from Facebook is best done using Facebook's Graph API**
  - You will need to get access tokens for this to work

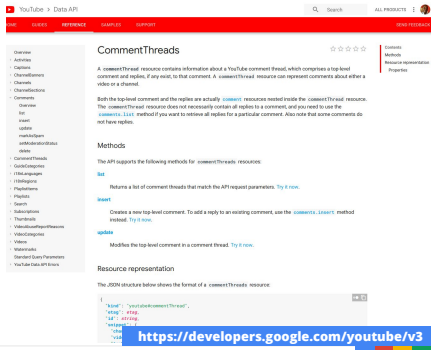


## Todos | Showcases—April 2, 2019 (3/6)



## Todos | Showcases—April 2, 2019 (4/6)

- YouTube data is accessible via the YouTube Data API
  - Those working with comments will need to both the 'Comments' and 'CommentThreads' resources

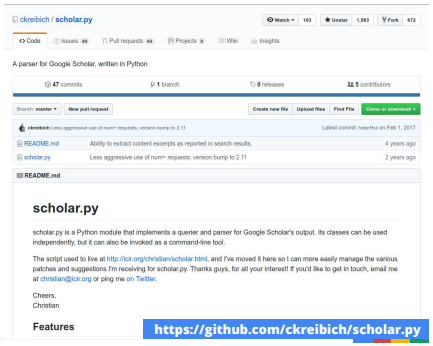



https://developers.google.com/youtube/v3



## Todos | Showcases—April 2, 2019 (5/6)

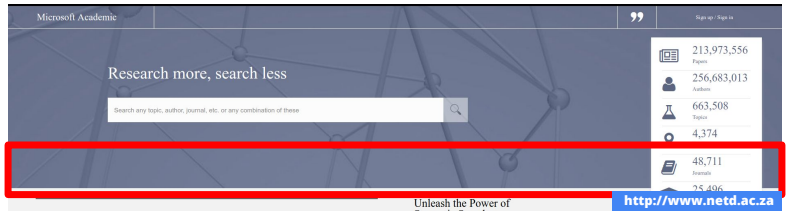
- Google Scholar does not provide an API, so you are going to have to scrap for data
  - Python Beautiful Soup library might be useful
  - Some projects on GitHub and Bitbucket might be useful when extracting data from Google Scholar




https://github.com/ckreibich/scholar.py



## Todos | Showcases—April 2, 2019 (6/6)




http://www.netd.ac.za

- Read up on how to use Microsoft Academic Graph API to extract data from Microsoft Academic Search



## Announcements—April 9, 2019 (1/3)

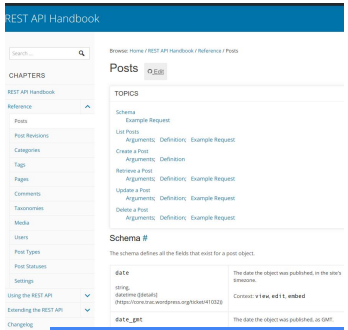
- Paper reading suggestions account towards participation
  - Simple formula will be used to aggregate scores from other activities, e.g. talks.
- Hint: Look for papers aligned to the problem you are solving in the Mini Project

| No. | First Name | Lastname |
|---|---|---|
| 1 | Chola | Paul Modest |
| 2 | Daka | John Chrispin |
| 3 | Lamaswala | Inonge |
| 4 | Mubanga | Mubanga |
| 5 | Mukuma | Nonde |
| 6 | Mulenga | David |
| 7 | Mumbi | Memory |
| 8 | Mutende | Kaumba |
| 9 | Nongola | Justin |
| 10 | Nyambe | Teddy |
| 11 | Phiri | Jonathan |
| 12 | Sampa | Anthny Wila |
| 13 | Shamane | Tasha |

https://groups.google.com/a/unza.zm/forum/?hl=en#!forum/csc5741

## Announcements—April 9, 2019 (2/3)

- Mini Project questions involving Facebook's Graph API can use WordPress REST API
- Anyone managed with Graph API
  - WordPress REST API route: restrict model implementation to popular sites like LusakaTimes, ZambianWatchDog & Mwebantu.




https://developer.wordpress.org/rest-api



## Announcements—April 9, 2019 (3/3)


```
id: 229497,
date: "2019-04-09T07:33:43",
date_gmt: "2019-04-09T05:33:43",
guid: {
    rendered: "https://www.lusakatimes.com/?p=229497"
},
modified: "2019-04-09T07:33:43",
modified_gmt: "2019-04-09T05:33:43",
slug: "faz-div-1-wrap-kansanshi-top-zone-2",
status: "publish",
type: "post",
link: "https://www.lusakatimes.com/2019/04/09/faz-div-1-wrap-kansanshi-top-zone-2/",
title: {
    rendered: "FAZ DIV 1 WRAP: Kansanshi top Zone 2"
},
content: {
    rendered: "<p>Kansanshi Dynamos have won the first half of the 2019 FAZ Division Zone 2 season with 36 points after a 2-1 win over Zesco Luapula in the Week 15 m
    a three point gap half way into the season.</p> <p>The Solwezi side benefited from Zesco's own goal to triumph in their latest match at home in Solwezi.</p> <p>G
    home in Ndola.</p> <p>Gomes remain stuck on 33 points from 15 matches played while Konkola are six points behind in third place.</p> <p>Fourth placed Indeni and
    are leading with a one point gap after overcoming Lundazi United 1-0 to amass 33 points while National Assembly are second on the table.</p> <p>Zambeef are toppi
    placed Kabwe Youth are three points behind.</p> <p>Zone 4 leaders Young Green Eagles have 31 points, two above second placed Zesco Shockers after 15 matches play
    Nkwazi 2-0 Riflemen</p> <p>Chipata City Council 1-0 Petauke United</p> <p>Lundazi United 0-1 Young Green Buffaloes</p> <p>Happy Hearts 0-2 Police College</p> <p>
    Rangers 2-2 Kafue Celtic</p> <p>Paramilitary 1-1 Lusaka City Council</p> <p>National Assembly 2-1 Romeki FC</p> <p>ZONE TWO</p> <p>Ndola United 3-1 Roan United</
    0-0 Indeni</p> <p>Nchanga Rangers 2-0 FQM FMs</p> <p>Gomes 0-1 Konkola Blades</p> <p>ZNS Luamfumu 2-0 Trident</p> <p>Kansanshi Dynamos 1-0 Zesco Luapula</p> <p>Ka
    <p>Intersport Youth 1-2 Riverside United</p> <p>Kateshi Coffee Bullets 0-0 Kabwe Youth Soccer Academy</p> <p>Malalo Police 0-1 Zambeef FC</p> <p>Mungwi Hotspurs
    Eagles</p> <p>Tazara Express 2-0 Tazara Rangers</p> <p>Real Nakonde 3-1 Mpulungu Harbour</p> <p>Kabwe Rangers 2-0 Chindwin Sentries</p> <p>ZONE FOUR</p> <p>Mazab
    Medics</p> <p>Sinazongwe United 1-0 Zesco Shockers</p> <p>Choma football Stars 0-2 Livingstone Pirates</p> <p>Kalomo Jetters 1-0 New Monze Swallows</p> <p>Young
    Stars</p> <p>Blue Arrows 0-0 Katima Border Stars</p> ",
    protected: false
},
excerpt: {
    rendered: "<p>Kansanshi Dynamos have won the first half of the 2019 FAZ Division Zone 2 season with 36 points after a 2-1 win over Zesco Luapula in the Week 15 m
    three point gap half way into the season. The Solwezi side benefited from Zesco's &hellip;</p> ",
    protected: false
},
```


https://www.lusakatimes.com/wp-json/wp/v2/posts



## Lecture Series Outline

- Part I: Getting Started With Python
- Part II: pandas, matplotlib and scikit-learn
- Part III: Academic Talk [Trial]
- Part IV: Paper Reading [Trial]
- Part V: About Next Week



## Lecture Series Outline

- Part I: Getting Started With Python
  - Introduction
  - Installation and Setup
  - Basics
  - Data Structures
  - Flow Control
  - Functions and Modules
- Part II: pandas, matplotlib and scikit-learn
- Part III: Academic Talk [Trial]
- Part IV: Paper Reading [Trial]
- Part V: About Next Week

## Getting Started With Python (1/3)

```
Python 3.6.7 (default, Oct 22 2018, 11:32:17)
[GCC 8.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import this
The Zen of Python, by Tim Peters

Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated.
Flat is better than nested.
Sparse is better than dense.
Readability counts.
Special cases aren't special enough to break the rules.
Although practicality beats purity.
Errors should never pass silently.
```



## Getting Started With Python (2/3)

- Python is an interpreted language
- Python is a scripting language
- Python is a general purpose language
- Python is an Object Oriented language
- [...]
- [...]
- We recommend using Python 3



## Getting Started With Python (3/3)

- Python statements can be executed directly from the interpreter
- Python scripts can be executed as shell commands

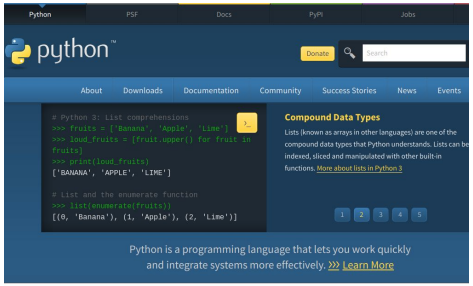


## Installation and Setup

- [...]

## Installation and Setup (1/2)

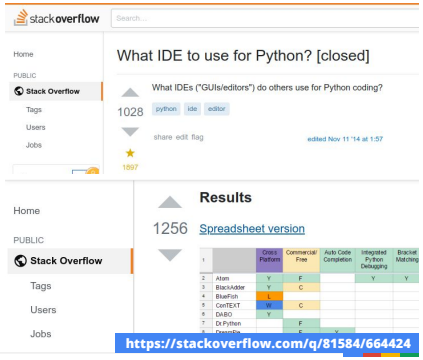
- **Download and install the latest version of Python 3**
  - Installers also available on course Web page, in the resources directory
- **Download and install the latest version of pip**






## Installation and Setup (2/2)

- **Any text editor will be sufficient for scripting.**
  - Vim, Notepad [...]
- **On IDEs**
  - There are plenty of IDEs to choose from
  - In the recent past, I have worked with Wing 101 and Kate






## Basics

- **No need to specify data types on variable declaration**
- **Indentation is important**



## Identifiers (1/2)

- **Python is case-sensitive, meaning uppercase and lowercase are considered as different**
  - age is different from AGE
  - favourite_course is different from Favourite_Course
- **Variable names, like other identifiers, follow rules**
  - can use letters, numbers or underscores
  - can't use other punctuation
  - can't start with a number
  - can't use Python keywords (reserved words)
- **The assignment operator in Python is the equals sign =**
  - **>>> age = 19**

## Identifiers (2/2)

- Python keywords (reserved words) can't be used when naming identifiers
- **>>> import keyword**
- **>>> keyword.kwlist**
- **['False', 'None', 'True', 'and', 'as', 'assert', 'break', 'class', 'continue', 'def', 'del', 'elif', 'else', 'except', 'finally', 'for', 'from', 'global', 'if', 'import', 'in', 'is', 'lambda', 'nonlocal', 'not', 'or', 'pass', 'raise', 'return', 'try', 'while', 'with', 'yield']**



## Comments (1/2)

- Comments are useful in explaining your code, and are ignored by the Python interpreter
- Single line comments are simply indicated with a hash # character
- Everything to the right of the hash is ignored
  - >>> course_code = "csc5741" # creates a variable course_code



## Comments (2/2)

- Multiple line comments are specified between sets of three quotes, ' ' ' or " " "

```
' ' ' Author: Paul Chola
Course: CSC 5741
Lecture #02 ' ' '

" " " Author: Tasha Shamane
Course: CSC 5741
Lecture #02 " " "
```



## Data Types (1/3)

- Variables don't require explicit type declaration in Python, as in other programming languages
  - **>>> x = 5**
- There are a few basic data types in Python

| | |
|---|---|
| **Integers** | **int** |
| **Float** | **float** |
| **String** | **str** |
| **Boolean** | **bool** |

## Data Types (2/3)

- Integer, whole numbers
  - **>>> i = 23**
- Float, floating point numbers
  - full stop indicates decimal point
  - **>>> d = 2.345**
- String, piece of text
  - enclosed in single (‘) or double quotes (“”)
  - **>>> x = ‘CSC 5741’**
  - **>>> y = “CSC 5741”**



## Data Types (3/3)

- Boolean, true or false
  - values True and False, start with capital letter
  - 0, '', [], (), {}, None are considered False, everything else is True
  - **>>> weekday = True**



## Functions (1/3)

- Functions are used to perform simple operations, sometimes on values
- Functions are called with round brackets ()
  - function_name()
- Functions can be passed certain values, which are referred to as parameters (or arguments) separated by commas
  - function_name(parameter)
  - function_name(parameter1, parameter2, ...)



## Functions (2/3)

- Python has many built-in functions, here are some:
  - print() function prints information to the screen
  - input() function gets information from the user
  - type() function returns data type of variable or value
  - **>>> x = 3**
  - **>>> type(x)**
  - **<class ‘int’>**

## Functions (3/3)

```
def csc5741(x, y='Y', z='Z'):
    print(x + ' ' + y)
return 0

csc5741('Xxxx', 'Yyyyy')
```

- All arguments are named
- Naming useful for optional arguments
- Return is optional



## Data Structures

- Tuple
  - var = (1, 2, 3, 4, 5)
- List
  - var = [1, 2, 3, 4, 5]
- Dictionary
  - var = {"one":1, "two":2, "three":3, "four":4, "five":5}
- Set
  - var = {1, 2, 3, 4, 5}



## Loops

```
for i in [1,2,3]:
    print(i)


while i < 5:
    i += 1
    print(i)
```

- No curly braces or "end for"
- Structure is derived from level of indentation
- One statement per line
- No semicolons required



## Modules (1/2)

- Modules facilitate extensibility and reusability
- Modules are collections of functions adding functionality to Python
- Modules can be imported using import keyword
  - Once modules are imported, their functions can be accessed by using the module name
  - The help() function displays what is contained in a module

```
from math import sqrt
import  math
```

## Modules (2/2)

- Single functions can be imported using the from statement
  - **>>> from math import sqrt**
- When using the from statement functions can be accessed without the module name
  - **>>> sqrt(16)**
- Everything from the module can be imported using an asterisk with the from statement
  - **>>> from math import ***



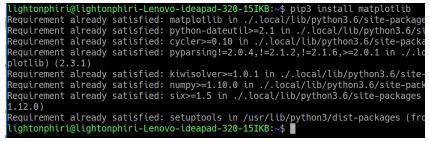
## Lecture Series Outline

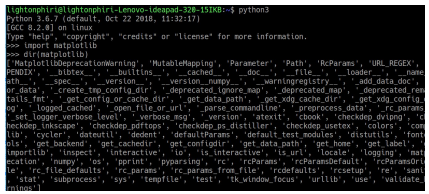
- Part I: Getting Started With Python
- Part II: pandas, matplotlib and scikit-learn
  - matplotlib
  - pandas
  - scikit-learn
- Part III: Academic Talk [Trial]
- Part IV: Paper Reading [Trial]
- Part V: About Next Week



## Matplotlib (1/7)

- The matplotlib library is best installed using pip, as with all libraries or using apt-get, if on Mac or Linux
  - pip3 install matplotlib
  - sudo apt-get install python-matplotlib
- Test installation by importing a library module

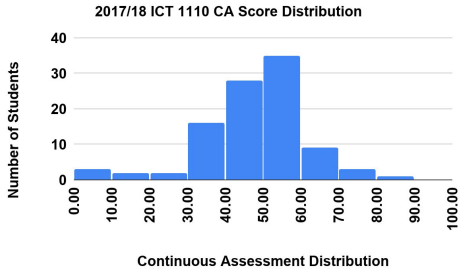





## Matplotlib (2/7)

- Basic elements of a plot
  - Plot title
  - Axis labels
  - Legend



## Matplotlib (3/7)

- Creating plots is a four-step process
  1) Import matplotlib
  2) Draw the plot
  3) Specify plot aesthetics
  4) Render plot



## Matplotlib (4/7)

- Creating plots is a four-step process
  1) Import matplotlib
     import matplotlib.pyplot as plt
  2) Draw the plot
  3) Specify plot aesthetics
  4) Render plot



## Matplotlib (5/7)

- Creating plots is a four-step process
  1) Import matplotlib
     import matplotlib.pyplot as plt
  2) Draw the plot
     plt.plot([...])
     plt.hist([...])
  3) Specify plot aesthetics
  4) Render plot
- Illustration
  - Simple plots
  - Plots using pandas dataframe



## Matplotlib (6/7)

- Creating plots is a four-step process
  1) Import matplotlib
     import matplotlib.pyplot as plt
  2) Draw the plot
     plt.plot([...])
     plt.hist([...])
  3) Specify plot aesthetics
     plt.xlabel("[...]")
     plt.ylabel("[...]")
  4) Render plot
- Illustration
  - Simple plots
  - Plots using pandas dataframe

## Matplotlib (7/7)

- **Creating plots is a four-step process**
  1) Import matplotlib
     import matplotlib.pyplot as plt
  2) Draw the plot
     plt.plot([...])
     plt.hist([...])
  3) Specify plot aesthetics
     plt.xlabel("[...]"); plt.ylabel("[...]")
     plt.legend()
  4) Render plot
     plt.show()
- **Illustration**
  - Simple plots
  - Plots using pandas dataframe



## Matplotlib—Exercise

- **JCTR regularly compiles Basic Needs Basket statistics for major towns in The Republic of Zambia. Using the Lusaka April 2018 BNB dataset (dataset available on https://goo.gl/1zSigy):**
  - Draw a line plot showing the trends of the Lusaka BNB between November 2016 and April 2018
  - Draw a bar plot showing BNB costs across Zambia in April 2018
  - Draw a pie chart showing the cost of basic food items in Lusaka for the month of April 2018



## Lecture Series Outline

- **Part I: Getting Started With Python**
- **Part II: pandas, matplotlib and scikit-learn**
  - matplotlib
  - pandas
  - scikit-learn
- **Part III: Academic Talk [Trial]**
- **Part IV: Paper Reading [Trial]**
- **Part V: About Next Week**



## Pandas (1/9)

- **Why use pandas instead a spreadsheet for data analysis**
  - Efficiency as data scales
  - Very user-friendly
  - Dataframe similar to spreadsheet

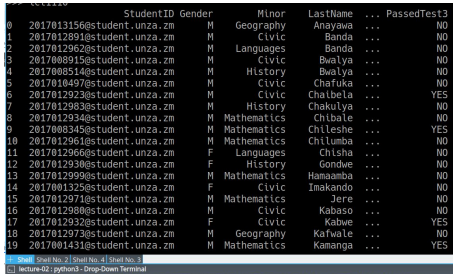
## Pandas (2/9)

- **Why use pandas instead a spreadsheet for data analysis**
  - Efficiency as data scales
  - Very user-friendly
  - Dataframe similar to spreadsheet



## Pandas (3/9)

- **Pandas DataFrame**
  - Two dimensional labeled data structure
  - DataFrame can be viewed as a representation of a Spreadsheet worksheet

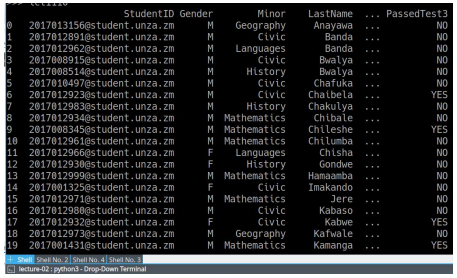


| | StudentID | Gender | Minor | LastName | ... | PassedTest3 |
|---|---|---|---|---|---|---|
| 0 | 2017013156@student.unza.zm | M | Geography | Anayawa | ... | NO |
| 1 | 2017012891@student.unza.zm | M | Civic | Banda | ... | NO |
| 2 | 2017012962@student.unza.zm | M | Languages | Banda | ... | NO |
| 3 | 2017008915@student.unza.zm | M | Civic | Bwalya | ... | NO |
| 4 | 2017008514@student.unza.zm | M | History | Bwalya | ... | NO |
| 5 | 2017010497@student.unza.zm | M | Civic | Chafuka | ... | NO |
| 6 | 2017012923@student.unza.zm | M | Civic | Chaibela | ... | YES |
| 7 | 2017012983@student.unza.zm | M | History | Chakulya | ... | NO |
| 8 | 2017012934@student.unza.zm | M | Mathematics | Chibale | ... | NO |
| 9 | 2017008345@student.unza.zm | M | Mathematics | Chileshe | ... | YES |
| 10 | 2017012961@student.unza.zm | M | Mathematics | Chilumba | ... | NO |
| 11 | 2017012966@student.unza.zm | F | Languages | Chisha | ... | NO |
| 12 | 2017012930@student.unza.zm | F | History | Gondwe | ... | NO |
| 13 | 2017012999@student.unza.zm | M | Mathematics | Hamaamba | ... | NO |
| 14 | 2017001325@student.unza.zm | F | Civic | Imakando | ... | NO |
| 15 | 2017012971@student.unza.zm | M | Mathematics | Jere | ... | NO |
| 16 | 2017012980@student.unza.zm | M | Civic | Kabaso | ... | NO |
| 17 | 2017012932@student.unza.zm | F | Civic | Kabwe | ... | YES |
| 18 | 2017012973@student.unza.zm | M | Geography | Kafwale | ... | NO |
| 19 | 2017001431@student.unza.zm | M | Mathematics | Kamanga | ... | YES |



## Pandas (4/9)

- **Pandas series**
  - One dimensional labeled array that can hold any data type.
  - Similar to column in Spreadsheet applications

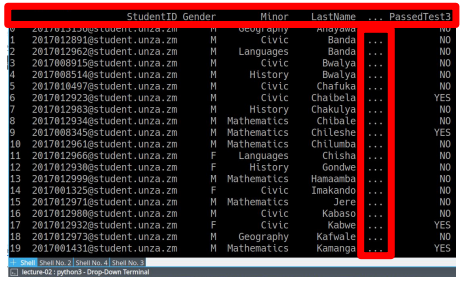


| | StudentID | Gender | Minor | LastName | ... | PassedTest3 |
|---|---|---|---|---|---|---|
| 0 | 2017013156@student.unza.zm | M | Geography | Anayawa | ... | NO |
| 1 | 2017012891@student.unza.zm | M | Civic | Banda | ... | NO |
| 2 | 2017012962@student.unza.zm | M | Languages | Banda | ... | NO |
| 3 | 2017008915@student.unza.zm | M | Civic | Bwalya | ... | NO |
| 4 | 2017008514@student.unza.zm | M | History | Bwalya | ... | NO |
| 5 | 2017010497@student.unza.zm | M | Civic | Chafuka | ... | NO |
| 6 | 2017012923@student.unza.zm | M | Civic | Chaibela | ... | YES |
| 7 | 2017012983@student.unza.zm | M | History | Chakulya | ... | NO |
| 8 | 2017012934@student.unza.zm | M | Mathematics | Chibale | ... | NO |
| 9 | 2017008345@student.unza.zm | M | Mathematics | Chileshe | ... | YES |
| 10 | 2017012961@student.unza.zm | M | Mathematics | Chilumba | ... | NO |
| 11 | 2017012966@student.unza.zm | F | Languages | Chisha | ... | NO |
| 12 | 2017012930@student.unza.zm | F | History | Gondwe | ... | NO |
| 13 | 2017012999@student.unza.zm | M | Mathematics | Hamaamba | ... | NO |
| 14 | 2017001325@student.unza.zm | F | Civic | Imakando | ... | NO |
| 15 | 2017012971@student.unza.zm | M | Mathematics | Jere | ... | NO |
| 16 | 2017012980@student.unza.zm | M | Civic | Kabaso | ... | NO |
| 17 | 2017012932@student.unza.zm | F | Civic | Kabwe | ... | YES |
| 18 | 2017012973@student.unza.zm | M | Geography | Kafwale | ... | NO |
| 19 | 2017001431@student.unza.zm | M | Mathematics | Kamanga | ... | YES |



## Pandas (5/9)

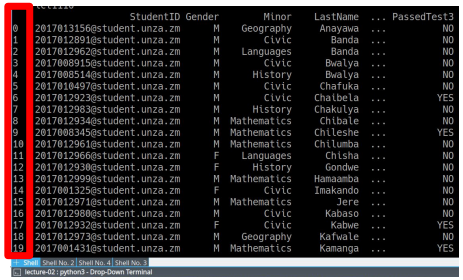
- **Columns**
  - Ellipse indicate more columns. Structure of data frame indicated on last line of output



| | StudentID | Gender | Minor | LastName | ... | PassedTest3 |
|---|---|---|---|---|---|---|
| 0 | 2017013156@student.unza.zm | M | Geography | Anayawa | ... | NO |
| 1 | 2017012891@student.unza.zm | M | Civic | Banda | ... | NO |
| 2 | 2017012962@student.unza.zm | M | Languages | Banda | ... | NO |
| 3 | 2017008915@student.unza.zm | M | Civic | Bwalya | ... | NO |
| 4 | 2017008514@student.unza.zm | M | History | Bwalya | ... | NO |
| 5 | 2017010497@student.unza.zm | M | Civic | Chafuka | ... | NO |
| 6 | 2017012923@student.unza.zm | M | Civic | Chaibela | ... | YES |
| 7 | 2017012983@student.unza.zm | M | History | Chakulya | ... | NO |
| 8 | 2017012934@student.unza.zm | M | Mathematics | Chibale | ... | NO |
| 9 | 2017008345@student.unza.zm | M | Mathematics | Chileshe | ... | YES |
| 10 | 2017012961@student.unza.zm | M | Mathematics | Chilumba | ... | NO |
| 11 | 2017012966@student.unza.zm | F | Languages | Chisha | ... | NO |
| 12 | 2017012930@student.unza.zm | F | History | Gondwe | ... | NO |
| 13 | 2017012999@student.unza.zm | M | Mathematics | Hamaamba | ... | NO |
| 14 | 2017001325@student.unza.zm | F | Civic | Imakando | ... | NO |
| 15 | 2017012971@student.unza.zm | M | Mathematics | Jere | ... | NO |
| 16 | 2017012980@student.unza.zm | M | Civic | Kabaso | ... | NO |
| 17 | 2017012932@student.unza.zm | F | Civic | Kabwe | ... | YES |
| 18 | 2017012973@student.unza.zm | M | Geography | Kafwale | ... | NO |
| 19 | 2017001431@student.unza.zm | M | Mathematics | Kamanga | ... | YES |

## Pandas (6/9)

- **Index**
  - Automatically generated, but can be changed
  - Uniquely identifies rows in the DataFrame

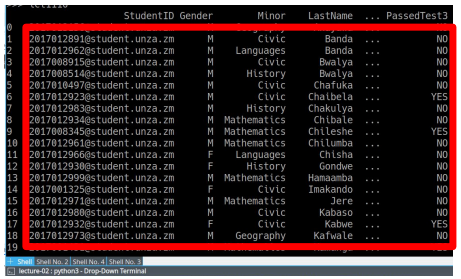


| | StudentID | Gender | Minor | LastName | ... | PassedTest3 |
|---|---|---|---|---|---|---|
| 0 | 2017013156@student.unza.zm | M | Geography | Anayawa | ... | NO |
| 1 | 2017012891@student.unza.zm | M | Civic | Banda | ... | NO |
| 2 | 2017012962@student.unza.zm | M | Languages | Banda | ... | NO |
| 3 | 2017008915@student.unza.zm | M | Civic | Bwalya | ... | NO |
| 4 | 2017008514@student.unza.zm | M | History | Bwalya | ... | NO |
| 5 | 2017010497@student.unza.zm | M | Civic | Chafuka | ... | NO |
| 6 | 2017012923@student.unza.zm | M | Civic | Chaibela | ... | YES |
| 7 | 2017012983@student.unza.zm | M | History | Chakulya | ... | NO |
| 8 | 2017012934@student.unza.zm | M | Mathematics | Chibale | ... | NO |
| 9 | 2017008345@student.unza.zm | M | Mathematics | Chileshe | ... | YES |
| 10 | 2017012961@student.unza.zm | M | Mathematics | Chilumba | ... | NO |
| 11 | 2017012966@student.unza.zm | F | Languages | Chisha | ... | NO |
| 12 | 2017012930@student.unza.zm | F | History | Gondwe | ... | NO |
| 13 | 2017012999@student.unza.zm | M | Mathematics | Hamaamba | ... | NO |
| 14 | 2017001325@student.unza.zm | F | Civic | Imakando | ... | NO |
| 15 | 2017012971@student.unza.zm | M | Mathematics | Jere | ... | NO |
| 16 | 2017012980@student.unza.zm | M | Civic | Kabaso | ... | NO |
| 17 | 2017012932@student.unza.zm | F | Civic | Kabwe | ... | YES |
| 18 | 2017012973@student.unza.zm | M | Geography | Kafwale | ... | NO |
| 19 | 2017001431@student.unza.zm | M | Mathematics | Kamanga | ... | YES |



## Pandas (7/9)

- **Data**



| | StudentID | Gender | Minor | LastName | ... | PassedTest3 |
|---|---|---|---|---|---|---|
| 1 | 2017012891@student.unza.zm | M | Civic | Banda | ... | NO |
| 2 | 2017012962@student.unza.zm | M | Languages | Banda | ... | NO |
| 3 | 2017008915@student.unza.zm | M | Civic | Bwalya | ... | NO |
| 4 | 2017008514@student.unza.zm | M | History | Bwalya | ... | NO |
| 5 | 2017010497@student.unza.zm | M | Civic | Chafuka | ... | NO |
| 6 | 2017012923@student.unza.zm | M | Civic | Chaibela | ... | YES |
| 7 | 2017012983@student.unza.zm | M | History | Chakulya | ... | NO |
| 8 | 2017012934@student.unza.zm | M | Mathematics | Chibale | ... | NO |
| 9 | 2017008345@student.unza.zm | M | Mathematics | Chileshe | ... | YES |
| 10 | 2017012961@student.unza.zm | M | Mathematics | Chilumba | ... | NO |
| 11 | 2017012966@student.unza.zm | F | Languages | Chisha | ... | NO |
| 12 | 2017012930@student.unza.zm | F | History | Gondwe | ... | NO |
| 13 | 2017012999@student.unza.zm | M | Mathematics | Hamaamba | ... | NO |
| 14 | 2017001325@student.unza.zm | F | Civic | Imakando | ... | NO |
| 15 | 2017012971@student.unza.zm | M | Mathematics | Jere | ... | NO |
| 16 | 2017012980@student.unza.zm | M | Civic | Kabaso | ... | NO |
| 17 | 2017012932@student.unza.zm | F | Civic | Kabwe | ... | YES |
| 18 | 2017012973@student.unza.zm | M | Geography | Kafwale | ... | NO |
| 19 | | | | | | |



## Pandas (8/9)

- **Some common operations**
  - Reading data files
    - df.read_csv([...])
    - df.read_html([...])
    - df.read_json([...])
    - df.read_*
  - Inspecting dataframes
    - df.head([...])
    - df.tail([...])
    - df.columns
    - df['[...]']



## Pandas (9/9)

- **Some common operations**
  - Converting to different file formats
    - df.to_csv([...])
    - df.to_excel([...])
    - df.to_sql([...])
    - df.to_*
  - Renaming columns
    - df.rename(columns={[...]})
  - Aggregating data
    - df.groupby(['[...]']).mean()
    - df.groupby('[...]').max()

## Pandas—Exercise

- Using pandas and the 2018/19 ICT 1110 assessment scores dataset (dataset available on https://goo.gl/wC1H7Q):
  - Print out the details (using a List) of the student who got the highest CA score
  - Print out the average CA scores for each of the different Minors
  - Print out the average CA scores for each gender
  - Export a summary table, to HTML, of mean CA scores by student Minors
  - Export, to CSV, a dataset consisting of StudentID, Minor and Total CA



## Lecture Series Outline

- Part I: Getting Started With Python
- Part II: pandas, matplotlib and scikit-learn
  - matplotlib
  - pandas
  - scikit-learn
- Part III: Academic Talk [Trial]
- Part IV: Paper Reading [Trial]
- Part V: About Next Week



## Scikit-learn

- Part I: Getting Started With Python
- Part II: pandas, matplotlib and scikit-learn
  - matplotlib
  - pandas
  - scikit-learn
- Part III: Academic Talk [Trial]
- Part IV: Paper Reading [Trial]
- Part V: About Next Week



## Scikit-learn

- Scikit-learn
  - Ensure that the module is installed by using the import statement

```
lightonphiri@lightonphiri-Lenovo-ideapad-320-15IKB:~$ pip3 install sklearn
Collecting sklearn
  Downloading https://files.pythonhosted.org/packages/1e/7a/dbb3be0ce9bd5c8b7e3d
sklearn-0.0.tar.gz
Collecting scikit-learn (from sklearn)
  Downloading https://files.pythonhosted.org/packages/5e/82/c0de5839d613b82bddd0
scikit_learn-0.20.3-cp36-cp36m-manylinux1_x86_64.whl (5.4MB)
    0% ||                                | 20kB 55kB/s eta 0:01:38
```

```
lightonphiri@lightonphiri-Lenovo-ideapad-320-15IKB:~$ python3
Python 3.6.7 (default, Oct 22 2018, 11:32:17)
[GCC 8.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import sklearn
>>> dir(sklearn)
['__SKLEARN_SETUP__', '__all__', '__builtins__', '__cached__', '__check_build', '__doc__',
ame__', '__package__', '__path__', '__spec__', '__version__', '_config', 'base', 'clone',
 'externals', 'get_config', 'logger', 'logging', 're', 'set_config', 'setup_module', 'show
rnings']
>>>
>>>
```

## Lecture Series Outline

- Part I: Getting Started With Python
- Part II: pandas, matplotlib and scikit-learn
- Part III: Academic Talk [Trial]
  - Towards Increased Online Visibility of Research in Zambia
- Part IV: Paper Reading [Trial]
- Part V: About Next Week



## Lecture Series Outline

- Part I: Getting Started With Python
- Part II: pandas, matplotlib and scikit-learn
- Part III: Academic Talk [Trial]
- Part IV: Paper Reading [Trial]
  - L. Phiri (2018) "Research Visibility in the Global South: Towards Increased Online Visibility of Scholarly Research Output in Zambia"
- Part V: About Next Week



## Academic Talk [Trial]

# Research Visibility in the Global South: Towards Increased Online Visibility of Research in Zambia

Lighton Phiri <lighton.phiri@unza.zm>
Department of Library and Information Science
University of Zambia



## Paper Reading Session [Trial]

[1] L. Phiri (2018) "Research Visibility in the Global South: Towards Increased Online Visibility of Scholarly Research Output in Zambia" http://lis.unza.zm/~lightonphiri/papers/paper-icict18-online_visibility.pdf

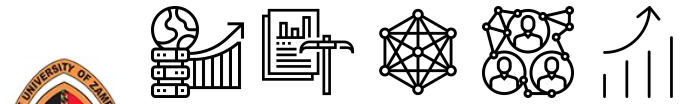
## Bibliography


[1] **Python for Beginners | Python.org**
https://www.python.org/about/gettingstarted

[2] **Pyplot tutorial – Matplotlib 3.0.3 documentation**
https://matplotlib.org/tutorials/introductory/pyplot.html

[3] **10 Minutes to pandas – pandas 0.22.0 documentation**
https://pandas.pydata.org/pandas-docs/version/0.22/10min.html



# CSC 5741
# Lecture 2: Python for Data Mining and Machine Learning


Lighton Phiri <lighton.phiri@unza.zm>
Department of Library and Information Science
University of Zambia