**Data Scientist** **specialization**:

### WHAT YOU WILL LEARN

* Describe common Python functionality and features used for data science.
* Explain distributions, sampling, and t-tests
* Query DataFrame structures for cleaning and processing
* Understand techniques such as lambdas and manipulating csv files
* Use GitHub to manage data science projects.

There are 10 courses in this program

* **Course 1: The Data Scientist’s Toolbox**
  + In this course you will get an introduction to the main tools and ideas in the data scientist's toolbox. The course gives an overview of the data, questions, and tools that data analysts and data scientists work with. There are two components to this course. The first is a conceptual introduction to the ideas behind turning data into actionable knowledge. The second is a practical introduction to the tools that will be used in the program like version control, markdown, git, GitHub, Python, and [Anaconda](https://www.anaconda.com/distribution/#windows).

### WHAT YOU WILL LEARN

* + - Create a Github repository
    - Explain essential study design concepts
    - Set up Python, Anaconda, Github
    - Understand the data, problems, and tools that data analysts work with
  + **Data Science Fundamentals** "3h"
    - In this module, we'll introduce and define data science and data itself. We'll also go over some of the resources that data scientists use to get help when they're stuck.
      * What is Data Science?
      * What is Data?
      * Data Science Process

## Python and Anaconda "2h"

* + - In this module, we'll help you get up and running with both Python and Anconda. Along the way, you'll learn some basics about both and why data scientists use them.
      * [Installing Anaconda](https://www.coursera.org/lecture/data-scientists-tools/installing-r-y6mU2)
      * Python Packages

## Version Control and GitHub "2h"

* + - During this module, you'll learn about version control and why it's so important to data scientists. You'll also learn how to use Git and GitHub to manage version control in data science projects.
      * [Version Control](https://www.coursera.org/lecture/data-scientists-tools/version-control-PjlHw)
      * Github and Git
      * Linking Github and Python IDE
      * Projects under Version Control

## Python Markdown, Scientific Thinking, and Big Data "5h"

* + - During this final module, you'll learn to use Python Markdown and get an introduction to three concepts that are incredibly important to every successful data scientist: asking good questions, experimental design, and big data.
      * [Python Markdown](https://www.coursera.org/lecture/data-scientists-tools/r-markdown-134kE)
      * Types of Data Science Questions
      * Experimental Design
      * Big Data
* **Course 2: Python Programming**
  + This course will introduce the learner to the basics of the python programming environment, including fundamental python programming techniques such as lambdas, reading and manipulating csv files, and the numpy library. The course will introduce data manipulation and cleaning techniques using the popular python pandas data science library and introduce the abstraction of the Series and DataFrame as the central data structures for data analysis, along with tutorials on how to use functions such as groupby, merge, and pivot tables effectively. By the end of this course, students will be able to take tabular data, clean it, manipulate it, and run basic inferential statistical analyses. This course should be taken before any of the other Applied Data Science with Python courses: Applied Plotting, Charting & Data Representation in Python, Applied Machine Learning in Python, Applied Text Mining in Python, Applied Social Network Analysis in Python.

### WHAT YOU WILL LEARN

* + - Describe common Python functionality and features used for data science
    - Explain distributions, sampling, and t-tests
    - Query DataFrame structures for cleaning and processing
    - Understand techniques such as lambdas and manipulating csv files

## Week 1: "3h"

## In this week you'll get an introduction to the field of data science, review common Python functionality and features which data scientists use. "1h"

* + - * Python Functions
      * Python Types and Sequences
      * Python More on Strings
      * Python Demonstration: Reading and Writing CSV files
      * Python Dates and Times
      * Advanced Python Objects, map()
      * Advanced Python Lambda and List Comprehensions
      * Advanced Python Demonstration: The Numerical Python Library (NumPy)

## Week 2: "3h"

* + - In this week of the course you'll learn the fundamentals of one of the most important toolkits Python has for data cleaning and processing -- pandas. You'll learn how to read in data into DataFrame structures, how to query these structures, and the details about such structures are indexed. The module ends with a programming assignment and a discussion question.
      * [Introduction](https://www.coursera.org/lecture/python-data-analysis/introduction-JvRF7)
      * The Series Data Structure
      * Querying a Series
      * The DataFrame Data Structure
      * DataFrame Indexing and Loading
      * Querying a DataFrame
      * Indexing Dataframes
      * Missing Values

## Week 3: "3h"

* + - In this week you'll deepen your understanding of the python pandas library by learning how to merge DataFrames, generate summary tables, group data into logical pieces, and manipulate dates. We'll also refresh your understanding of scales of data, and discuss issues with creating metrics for analysis. The week ends with a more significant programming assignment.
      * [Merging Dataframes](https://www.coursera.org/lecture/python-data-analysis/merging-dataframes-08sf6)
      * Pandas Idioms
      * Group by
      * Scales
      * Pivot Tables
      * Date Functionality

## Week 4: "6h"

* + - In this week of the course you'll be introduced to a variety of statistical techniques such a distributions, sampling and t-tests. The majority of the week will be dedicated to your course project, where you'll engage in a real-world data cleaning activity and provide evidence for (or against!) a given hypothesis. This project is suitable for a data science portfolio, and will test your knowledge of cleaning, merging, manipulating, and test for significance in data. The weekends with two discussions of science and the rise of the fourth paradigm -- data driven discovery.
      * [Introduction](https://www.coursera.org/lecture/python-data-analysis/introduction-CrBP0)
      * Distributions
      * More Distributions
      * Hypothesis Testing in Python
* **Course 3: Getting and Cleaning Data**
  + Before you can work with data you have to get some. This course will cover the basic ways that data can be obtained. The course will cover obtaining data from the web, from APIs, from databases and from colleagues in various formats. It will also cover the basics of data cleaning and how to make data “tidy”. Tidy data dramatically speed downstream data analysis tasks. The course will also cover the components of a complete data set including raw data, processing instructions, codebooks, and processed data. The course will cover the basics needed for collecting, cleaning, and sharing data.

### WHAT YOU WILL LEARN

* Apply data cleaning basics to make data "tidy"
* Obtain usable data from the web, APIs, and databases
* Understand common data storage systems
* Use Python for text and date manipulation
  + look at finding data and reading different file types. "2h"
    - [Obtaining Data Motivation](https://www.coursera.org/lecture/data-cleaning/obtaining-data-motivation-MIAyy)
    - Raw and Processed Data
    - Components of Tidy Data
    - Downloading Files
    - Reading Local Files
    - Reading Excel Files
    - Reading XML
    - Reading JSON
    - The data.table Package
  + Getting and Cleaning Data! The primary goal is to introduce you to the most common data storage systems and the appropriate tools to extract data from web or from databases like MySQL. "1h"
    - [Reading from MySQL](https://www.coursera.org/lecture/data-cleaning/reading-from-mysql-njjbw)
    - Reading from HDF5
    - Reading from The Web
    - Reading From APIs
    - Reading From Other Sources
  + focus on organizing, merging and managing the data you have collected "10h"
    - [Subsetting and Sorting](https://www.coursera.org/lecture/data-cleaning/subsetting-and-sorting-aqd2Y)
    - Summarizing Data
    - Creating New Variables
    - Reshaping Data
    - Managing Data Frames with dplyr - Introduction
    - Managing Data Frames with dplyr - Basic Tools
    - Merging Data
  + finish up with lectures on text and date manipulation in Python. In this final week we will also focus on peer grading of Course Projects. "6h"
    - [Editing Text Variables](https://www.coursera.org/lecture/data-cleaning/editing-text-variables-drpnT)
    - Regular Expressions I
    - Regular Expressions II
    - Working with Dates
    - Data Resources
* **Course 4: Exploratory Data Analysis**
  + This course covers the essential exploratory techniques for summarizing data. These techniques are typically applied before formal modeling commences and can help inform the development of more complex statistical models. Exploratory techniques are also important for eliminating or sharpening potential hypotheses about the world that can be addressed by the data. We will cover in detail the plotting systems in Python as well as some of the basic principles of constructing data graphics. We will also cover some of the common multivariate statistical techniques used to visualize high-dimensional data.

### WHAT YOU WILL LEARN

* + - * Apply cluster analysis techniques to locate patterns in data
      * Make graphical displays of very high dimensional data
      * Understand analytic graphics and the base plotting system in Python
      * Use advanced graphing systems such as the Lattice system
  + **Week 1:** This week covers the basics of analytic graphics and the base plotting system in Python. We've also included some background material to help you install Python if you haven't done so already. "20h"
    - Installing Python on Windows
    - Setting Your Working Directory (Windows)
    - Principles of Analytic Graphics
    - Exploratory Graphs
    - Plotting Systems in Python
    - Base Plotting System
    - Base Plotting Demonstration
    - Graphics Devices in Python
  + **Week 2:** Welcome to Week 2 of Exploratory Data Analysis. This week covers some of the more advanced graphing systems available in Python: the Lattice system and the ggplot2 system. While the base graphics system provides many important tools for visualizing data, it was part of the original Python system and lacks many features that may be desirable in a plotting system, particularly when visualizing high dimensional data. The Lattice and ggplot2 systems also simplify the laying out of plots making it a much less tedious process. ""17h"
    - Lattice Plotting System
    - ggplot2
  + **Week 3:** Welcome to Week 3 of Exploratory Data Analysis. This week covers some of the workhorse statistical methods for exploratory analysis. These methods include clustering and dimension reduction techniques that allow you to make graphical displays of very high dimensional data (many many variables). We also cover novel ways to specify colors in Python so that you can use color as an important and useful dimension when making data graphics. All of this material is covered in chapters 9-12 of my book Exploratory Data Analysis with Python. "13h"
    - Hierarchical Clustering
    - K-Means Clustering
    - Dimension Reduction
    - Working with Color in Python Plots
  + **Week 4:** This week, we'll look at two case studies in exploratory data analysis. The first involves the use of cluster analysis techniques, and the second is a more involved analysis of some air pollution data. How one goes about doing EDA is often personal, but I'm providing these videos to give you a sense of how you might proceed with a specific type of dataset. "6h"
    - [Clustering Case Study](https://www.coursera.org/lecture/exploratory-data-analysis/clustering-case-study-IH1Y2)
    - Air Pollution Case Study
* **Course 5: Reproducible Research**
  + This course focuses on the concepts and tools behind reporting modern data analyses in a reproducible manner. Reproducible research is the idea that data analyses, and more generally, scientific claims, are published with their data and software code so that others may verify the findings and build upon them. The need for reproducibility is increasing dramatically as data analyses become more complex, involving larger datasets and more sophisticated computations. Reproducibility allows for people to focus on the actual content of a data analysis, rather than on superficial details reported in a written summary. In addition, reproducibility makes an analysis more useful to others because the data and code that actually conducted the analysis are available. This course will focus on literate statistical analysis tools which allow one to publish data analyses in a single document that allows others to easily execute the same analysis to obtain the same results.

### WHAT YOU WILL LEARN

* + - * Determine the reproducibility of analysis project
      * Organize data analysis to help make it more reproducible
      * Publish reproducible web documents using Markdown
      * Write up a reproducible data analysis using knitr

## Week 1: Concepts, Ideas, & Structure "2h"

## This week will cover the basic ideas of reproducible research since they may be unfamiliar to some of you. We also cover structuring and organizing a data analysis to help make it more reproducible. I recommend that you watch the videos in the order that they are listed on the web page, but watching the videos out of order isn't going to ruin the story.

* + - * What is Reproducible Research About?
      * Reproducible Research: Concepts and Ideas
      * Scripting Your Analysis
      * Structure of a Data Analysis
      * Organizing Your Analysis

## Week 2: Markdown & knitr "3h"

## This week we cover some of the core tools for developing reproducible documents. We cover the literate programming tool knitr and show how to integrate it with Markdown to publish reproducible web documents. We also introduce the first peer assessment which will require you to write up a reproducible data analysis using knitr.

* + - * [Coding Standards in Python](https://www.coursera.org/lecture/reproducible-research/coding-standards-in-r-MW8Sp)
      * Markdown
      * Python Markdown
      * Python Markdown Demonstration
      * Knitr

## Week 3: Reproducible Research Checklist & Evidence-based Data Analysis "1h"

## This week covers what one could call a basic check list for ensuring that a data analysis is reproducible. While it's not absolutely sufficient to follow the check list, it provides a necessary minimum standard that would be applicable to almost any area of analysis.

* + - * [Communicating Results](https://www.coursera.org/lecture/reproducible-research/communicating-results-cRnt1)
      * RPubs
      * Reproducible Research Checklist
      * Evidence-based Data Analysis

## Week 4: Case Studies & Commentaries "3h"

## This week there are two case studies involving the importance of reproducibility in science for you to watch.

* + - * [Caching Computations](https://www.coursera.org/lecture/reproducible-research/caching-computations-5P2rt)
      * Case Study: Air Pollution
      * Case Study: High Throughput Biology
      * Commentaries on Data Analysis
      * Introduction to Peer Assessment 2
* **Course 6: Statistical Inference**
  + Statistical inference is the process of drawing conclusions about populations or scientific truths from data. There are many modes of performing inference including statistical modeling, data oriented strategies and explicit use of designs and randomization in analyses. Furthermore, there are broad theories (frequentists, Bayesian, likelihood, design based, …) and numerous complexities (missing data, observed and unobserved confounding, biases) for performing inference. A practitioner can often be left in a debilitating maze of techniques, philosophies and nuance. This course presents the fundamentals of inference in a practical approach for getting things done. After taking this course, students will understand the broad directions of statistical inference and use this information for making informed choices in analyzing data.
* **Course 7: Regression Models**
* **Course 8: Practical Machine Learning**
* **Course 9: Developing Data Products**
* **Course 10: Data Science Capstone**