As of 2022-03-09

DATASOCIETY:

Course Catalog

Built for MeldR

Table of Contents

INTRODUCTION

Table of Contents

DATA SCIENCE (PYTHON) (cont'd)

Table of Contents

FRONT END 80

Table of Contents

FRONT END (cont'd)

Table of Contents

BACK END (cont'd)

Table of Contents

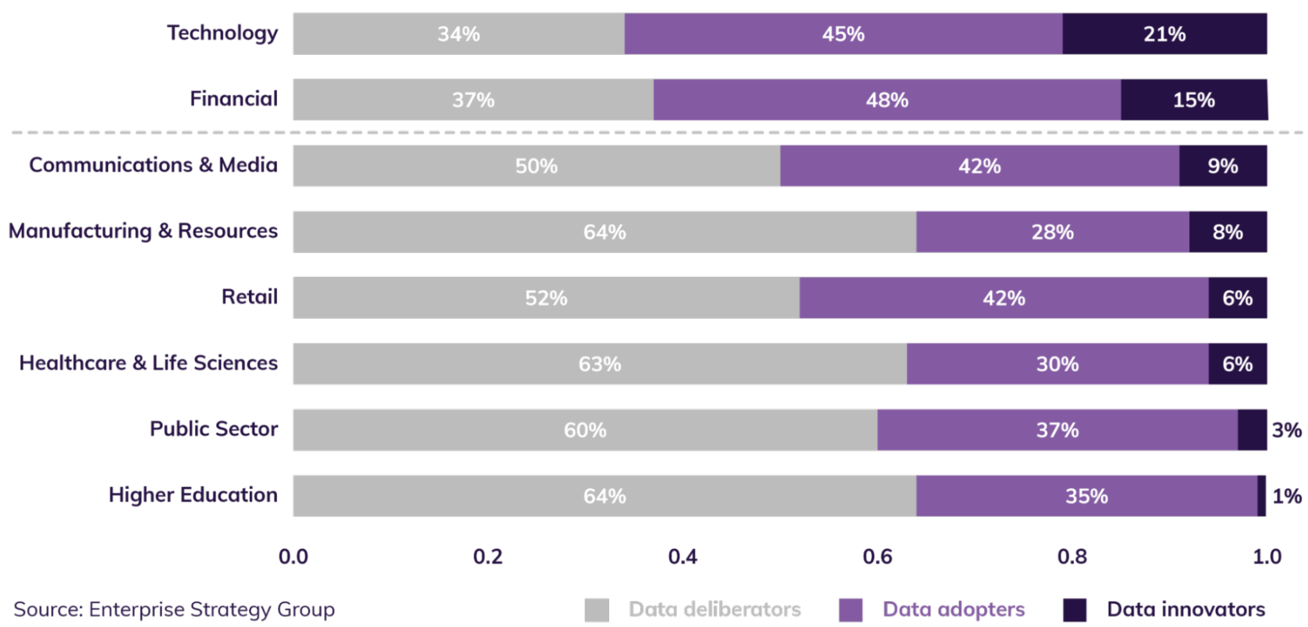
MACHINE LEARNING OPERATIONS (MLOPS) (cont'd)

Preparing for a Data Economy

In the last 15 years, 52% of the Fortune 500 companies have disappeared.

In 2015, the average life expectancy of a company was 15 years, compared to 75 years in 1955. The world is changing more rapidly than ever - organizations need to live and breathe data to stay competitive and effective.

Percent of respondents in each maturity stage, by industry



Source: Enterprise Strategy Group

Businesses that integrate data into their operations and lead with analytics grow their annual revenue by 5.3 percent and decrease operation costs by 4.85 percent. The organizations that build their data infrastructures and cultivate data-literate workforces will thrive in the new data economy.

HIRING ONE DATA SCIENTIST

TRAINING 30 STAFF MEMBERS

By implementing a data analytics training program, you'll create a workforce that is savvy about data and has demonstrated its dedication to the organization's mission. Data literacy improves efficiency across all levels of an organization, from planning project resources to automating processes.

How to Review this Catalog

There are a lot of courses here – how do I know where to start?

If you feel overwhelmed going through this catalog, you're not alone. Whether you're just exploring our offerings or looking for a specific topic, the following guidelines may be helpful to you in determining which courses are most relevant to your organization's needs:

1. For staff who don't directly manage data but need to understand common data terms and the importance of data, look to our Data Literacy pathway.
2. For analysts who want to automate data processes and build foundational machine learning processes, look to our Data Science pathway.
3. For programmers who have a foundation in machine learning and want to build out their predictive capabilities, look to our Data Science pathways.
4. For data scientists who want to embed advanced modeling, look to our Artificial Intelligence pathway.

Data Society can customize all programs and course offerings to fit your use cases and schedule. If you're not sure what makes sense for you or if you want to discuss our customization options, reach out to us at **hello@datasociety.com** or call us at **202-600-9635**.

Included Services

- VIRTUAL LIVE-STREAMED INSTRUCTION
- COURSE REFERENCE GUIDES AND CHEAT SHEETS
- PRACTICAL EXERCISES TO SHARPEN SKILLS
- REUSABLE DATA SETS / CODING TEMPLATES
- COMPANION STEP-BY-STEP WORKBOOKS
- CAPSTONE PROJECT SUPPORT AND GUIDANCE

Topics and Tools We Cover

This catalog represents a subset of programs that we provide, but it doesn't capture the full breadth of our content. Curious about what else we teach? Here are some more topics and tools.

TOOLS	TECHNIQUES
● Amazon Web Services (AWS) ● Microsoft Azure ● SQL / Relational databases ● Python ● R ● Scala ● Spark ● TensorFlow ● Keras	● Data science for executives ● Data visualization ● Data storytelling ● Foundational statistics ● Programming fundamentals ● Data cleaning ● Data transformation ● Unsupervised machine learning ● Supervised machine learning ● Text mining ● Network analysis ● Neural networks ● Deep learning ● Cyber security

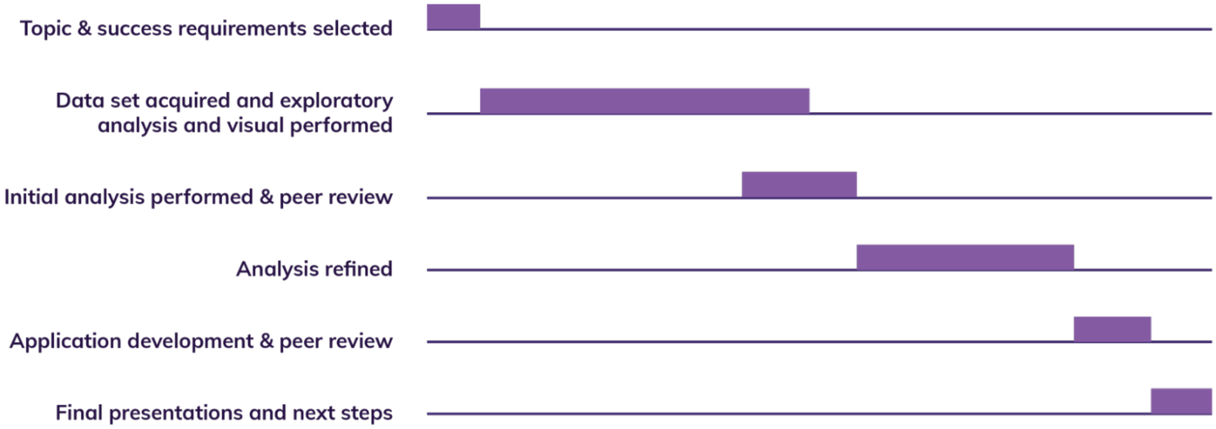
If you'd like to learn more about the other training programs that we teach or if you want to discuss our customization options, reach out to us at **hello@datasociety.com** or call us at **202-600-9635**.

Capstone Project Plan

The capstone project is a core component of demonstrating the competency and application of employees' skills and can be integrated into your learning pathway. We work closely with students to help them develop a project plan that is scoped appropriately and facilitates success.

Our program will track student progress and set aside time to meet with students individually on scheduled intervals. One of the most critical aspects of the capstone facilitation is ensuring that students frame the right questions and can access the correct datasets. We recommend that students define their key question and hypothesis early on to maximize the time they can spend building and refining the analysis.

One of the main objectives of this program is to teach students the right frameworks and skill sets to understand how to apply their knowledge in different scenarios and problem-solve effectively.



While Data Society instructors will provide students with regular support, we recommend that all participants set aside at least 4-6 hours per week during the program for independent work on the capstone project

Data Science Literacy

Data Science Literacy

A pathway ideal for those who would like to get a bird's eye view of the world of Data Science, specifically for those who are managing teams with Data Scientists, or leaders who want to run data-driven companies.

Courses Include

- **Data Science for Executives**
- **Data Science for Managers**
- **Fundamentals of Data Literacy**
- **Data Visualization Design & Storytelling**

DATA SOURCES & EXAMPLES USED

- Industry or domain-specific examples
- Customized data that maximizes the attendees' engagement and retention of skills and can be applied directly within your organization

PATH DURATION

- 42 Hours, approximately
- Available in flexible schedule format

MODALITY

- Proposed live-streaming (in-person available post-COVID-19)
- Instructor-led (TAs are optional)

RECOMMENDATIONS

- Up to 30 learners per cohort
- Review individual course prerequisites and software requirements

Data Science for Executives

6 hours of instruction

A course that focuses on basics of data analytics and building data driven culture across organizations with informed leadership.

OBJECTIVES	PREREQUISITES
1. Understand and navigate through data components and empower individuals to implement strategic data-driven culture and innovation across the organization	None

SYLLABUS & TOPICS COVERED

- Intro To Data Analytics
  - Define data and understand use cases of data analytics
  - Assess data quality
  - Finding data resources
- Plan And Build Data Driven Culture
  - Assess and build data driven culture, thinking and innovation
  - Plan data projects

SOFTWARE REQUIREMENTS

TBD

Data Science for Managers

12 hours of instruction

This course is designed for managers seeking to bolster their data literacy with a deep dive into data science tools and teams, project life cycles, and methods.

OBJECTIVES	PREREQUISITES
1. To understand the structure of data science projects from start to finish, and to make more informed decisions about how to identify data-driven solutions, structure the teams, allocate resources, and interpret results	None

SYLLABUS & TOPICS COVERED

- Overview Of Data Analytics
  - Data and its importance
  - Types and use cases of data
  - Data analytics maturity model and stages
- Data Quality And Governance
  - Data governance
  - Data ethics
  - Structuring data teams
  - Building a data driven culture
- Intro To Data Science
  - Process of data science
  - Putting together a data project
- Foundational Data Science Methods
  - Intro to machine learning
  - Types of machine learning models
- Advanced Data Science Methods
  - Text mining
  - Graph analysis
  - Neural networks
- Data Visualization And Story Telling
  - Intro to data visualization
  - Types of data visualization
  - Storytelling using data

SOFTWARE REQUIREMENTS

TBD

Fundamentals of Data Literacy

12 hours of instruction

This course is designed for anyone seeking to bolster their data literacy with a deep dive into data science tools and teams, project life cycles, and methods.

OBJECTIVES	PREREQUISITES
1. This course will demystify the structure of data science projects from start to finish, helping students to make more informed decisions about how to identify data-driven solutions, structure their teams, allocate resources, and interpret results	Data Science for Executives

SYLLABUS & TOPICS COVERED

- Overview Of Data Analytics
  - Data and its importance
  - Types and use cases of data
  - Data analytics maturity model and stages
- Data Quality And Governance
  - Assess data quality
  - Data governance
  - Building a data driven culture
- Overview Of Data Science
  - Process of data science
- Foundational Data Science Methods
  - Intro to machine learning
  - Types of machine learning models
- Advanced Data Science Methods
  - Text mining
  - Graph analysis
  - Neural networks
- Data Visualization And Story Telling
  - Intro to data visualization
  - Types of data visualization
  - Storytelling using data

SOFTWARE REQUIREMENTS

TBD

Data Visualization Design & Storytelling

12 hours of instruction

This course focuses on the fundamentals of data visualization, which helps support data-driven decision-making and to create a data-driven culture.

OBJECTIVES	PREREQUISITES
1. To recognize misleading or inaccurate charts and graphs, understand the design principles involved in creating compelling and accurate visualizations, and create a narrative that accurately supports the data, by revealing actionable insights	Fundamentals of Data Literacy

SYLLABUS & TOPICS COVERED

- Intro To Data Visualization
  - Types of data
  - Elements of a data chart
  - Types of visualizations
  - Process of choosing the right visual
- Visual Design Theory
  - Visual design theory
  - Common mistakes
  - Misleading statistics and visual distortions
- Storytelling With Data
  - Intro to storytelling
  - Process of storytelling with data
  - Storytelling activity

SOFTWARE REQUIREMENTS

TBD

DATA SOCIETY:

Beginner

Beginner

A pathway that gives foundational knowledge of Version control, Python programming, Backend engineering and Frontend engineering

Courses Include

- **Git**
- **Intro to Python**
- **Intermediate Python**
- **Internet Basics**
- **Domain & Hosting**
- **Understanding Different OS Concepts**
- **Operating System Basics, Terminal & Shell Scripts**

DATA SOURCES & EXAMPLES USED

- Industry or domain-specific examples
- Customized data that maximizes the attendees' engagement and retention of skills and can be applied directly within your organization

PATH DURATION

- 28 Hours, approximately
- Available in flexible schedule format

MODALITY

- Proposed live-streaming (in-person available post-COVID-19)
- Instructor-led (TAs are optional)

RECOMMENDATIONS

- Up to 30 learners per cohort
- Review individual course prerequisites and software requirements

Git

4 hours of instruction

A course that builds a foundational understanding of Git and version control systems. By the end of the course students will be able to use key Git commands and interact with remote repositories like GitHub, Bitbucket, and GitLab. Students will also get acquainted with the concept of continuous integration and deployment (CI/CD).

OBJECTIVES	PREREQUISITES
1. Describe the use cases for version control 2. Describe the core concepts of CI/CD and Git's part in it 3. Create, modify, and update remote repositories and use Git to track their history	None

SYLLABUS & TOPICS COVERED

- Basics
  - The what and why of version control
  - Git overview
  - Installing git
- Remote Repositories
  - What is a repository?
  - Forks, clones, branches, and commits
  - Pushing, pulling, PR's, and merging
  - Other basic git commands
  - Understanding and using gitignore
- CI/CD
  - What is CI/CD?
  - Understanding environment, repository, and deployment variables

SOFTWARE REQUIREMENTS

browser, text editor

Intro to Python

4 hours of instruction

This course will cover how to leverage Python's capabilities to manipulate and explore data. By the end of this program, students will be able to outline use cases for Python and gain foundational skills to automate tedious data processes.

OBJECTIVES	PREREQUISITES
1. Program in Python 2. Perform data manipulation using python 3. Understand data structures	Git

SYLLABUS & TOPICS COVERED
• Basics   • Data Science industry overview   • Python as a programming language and the tools used to write and execute Python code   • Basic operations and data types • Basic Data Structures   • Base data types vs data structures   • Lists, tuples, sets, and dictionaries

SOFTWARE REQUIREMENTS
Anaconda, Python, Jupyter Notebooks

Intermediate Python

4 hours of instruction

This course will unravel different functionalities that Python offers. The students will be able to write modular code, conditional statements, loops, function definitions and list comprehensions.

OBJECTIVES	PREREQUISITES
1. Program proficiently in Python 2. Automate data cleaning and processing using Python 3. Implement modular code, conditional statements, loops, function definitions and list comprehensions	Intro to Python

SYLLABUS & TOPICS COVERED
• Control Flow   • Control flow structures and practice of writing of modular code   • Conditional statements and blocks   • For loops, while loops, and list comprehensions • Functions   • Function definition and use cases   • Function implementation in Python

SOFTWARE REQUIREMENTS
Anaconda, Python, Jupyter Notebooks

Internet Basics

4 hours of instruction

A course that builds a foundational understanding of what you see when opening a browser or communicating with others through the web.

OBJECTIVES	PREREQUISITES
1. Describe what Internet is and how it works 2. Be able to navigate the world of browsers, domains, and websites	Git

SYLLABUS & TOPICS COVERED
• How It Works   • The role and function of the Internet   • Key components that make up communication with others through the web possible • HTTP   • Protocols of communication over the world wide web   • Definition and role of HTTP • Browsers   • Definition and role of a browser   • Key browser capabilities   • Different browsers and their advantage   • Account and password management on browsers

SOFTWARE REQUIREMENTS
browser, text editor

Domain & Hosting

4 hours of instruction

A course that builds a foundational understanding of the domain name system, how to host a webpage and add a custom domain name.

OBJECTIVES	PREREQUISITES
1. Describe what DNS and Hosting are 2. Be able to create and host a webpage 3. Be able to set up a custom domain	Internet Basics

SYLLABUS & TOPICS COVERED

- DNS
  - DNS definition and use cases
  - DNS servers and their function
- Domain Name
  - Domain name structure
  - The role and function of subdomains
- Hosting
  - DNS definition and use cases
  - DNS servers and their function
  - Domain name structure
  - The role and function of subdomains
  - Web hosting service and its function
  - Types of hosting services and their use cases
  - Set up and host a single web page
  - Set up and add a custom domain name

SOFTWARE REQUIREMENTS

browser, text editor

Understanding Different OS Concepts

4 hours of instruction

A course that builds foundational knowledge of what an operating system is. It walks through the different core concepts of OS and its inner workings.

OBJECTIVES	PREREQUISITES
1. General understanding of Operating Systems, OS system processes 2. Understanding of basic components and types of Operating Systems	Domain & Hosting

SYLLABUS & TOPICS COVERED
• Intro To OS Concepts • I/O Management • Virtualization • Memory/Storage • File Systems • Introto OS Concepts

SOFTWARE REQUIREMENTS
TBD

Operating System Basics, Terminal & Shell Scripts

4 hours of instruction

A course that builds foundational knowledge of how operating system works. It covers the basic terminal and shell scripts

OBJECTIVES	PREREQUISITES
1. General understanding of Operating Systems with the ability to use basic terminal commands 2. Basic understanding of Linux/Unix shell scripting	Understanding Different OS Concepts

SYLLABUS & TOPICS COVERED
• Shell Scripting Basics   • Command Line   • Common Linux/Unix Shell Commands   • Useful Features of the Bash Shell

SOFTWARE REQUIREMENTS
TBD

Data Science (Python)

Data Science (Python)

A pathway that takes students through the courses that span the topics in Data Science. The pathway starts with the very foundations of how the world of machine learning works, what makes up classification or clustering methods and ends with advanced topics in Text Mining, Outlier Detection, and much more. This pathway equips learners with a wide rande of tools and frameworks to become data scientists.

Courses Include

- **Data Wrangling in Python**
- **Storytelling with Data**
- **Intro to Visualization in Python**
- **Interactive Visualization with Plotly**
- **Interactive Visualization with Bokeh**
- **Intro to Classification**
- **Decision Trees**
- **Ensemble Methods**
- **Optimizing Ensemble Methods**
- **Support Vector Machines**
- **Intro to NLP**
- **Topic Modeling in NLP**
- **And more...!**

DATA SOURCES & EXAMPLES USED

- Industry or domain-specific examples
- Customized data that maximizes the attendees' engagement and retention of skills and can be applied directly within your organization

PATH DURATION

- 138 Hours, approximately
- Available in flexible schedule format

MODALITY

- Proposed live-streaming (in-person available post-COVID-19)
- Instructor-led (TAs are optional)

RECOMMENDATIONS

- Up to 30 learners per cohort
- Review individual course prerequisites and software requirements

Data Wrangling in Python

6 hours of instruction

This course will cover how to leverage Python's libraries like NumPy and Pandas to manipulate and explore data.

OBJECTIVES	PREREQUISITES
1. Perform data processing using NumPy and Pandas 2. Clean unstructured data sets using python so that they can be explored and analyzed more effectively 3. Explore the power of dataframes	Intermediate Python

SYLLABUS & TOPICS COVERED

- Data Wrangling With Numpy
  - NumPy use cases and object types
  - NumPy array manipulation
- Data Wrangling With Pandas
  - Pandas use cases and basic operations
  - Dataframe definition and manipulation

SOFTWARE REQUIREMENTS

Anaconda, Python, Jupyter Notebooks

Storytelling with Data

4 hours of instruction

Students will learn about the power of data analysis and narrating a story through data visuals. Students will take their Python skills to the next level and learn how to effectively communicate the results of their analysis with a few lines of code.

OBJECTIVES	PREREQUISITES
1. Introduce the main elements of building a story with data 2. Learn about audience engagement and visual building blocks 3. Understand color theory and its best practices	Data Wrangling in Python

SYLLABUS & TOPICS COVERED
• Storytelling With Data • Outline storytelling with data • Discuss the different elements of data visualization

SOFTWARE REQUIREMENTS
Python, Anaconda, Jupyter Notebook or JupyterLab

Intro to Visualization in Python

4 hours of instruction

This course covers the basics of data visualization and exploratory data analysis. It helps students learn different plots and their use cases.

OBJECTIVES	PREREQUISITES
1. Introduce the main elements of building a story with data 2. Build interactive data visualizations in Python 3. Select the appropriate data visualization for the scenario	Intro to Python

SYLLABUS & TOPICS COVERED

- Exploratory Data Analysis
  - Exploratory data analysis use cases
  - Perform EDA on data
- Static Plots
  - Data visualization basics and use cases
  - Build plots with matplotlib

SOFTWARE REQUIREMENTS

Python, Anaconda, Jupyter Notebook or JupyterLab

Interactive Visualization with Plotly

4 hours of instruction

Students will learn about the power of data visualization with Plotly. By the end of the course, students will be able to create interactive visualizations and publish dynamic visualizations to websites.

OBJECTIVES	PREREQUISITES
1. Introduce difference between static and interactive visualizations 2. Build static data visualizations in Python 3. Build interactive data visualizations in Python 4. Select the appropriate data visualization for the scenario	Intro to Visualization in Python

SYLLABUS & TOPICS COVERED
• Interactive Plots   • Interactive visualization use cases   • Visualizing data with cufflinks and plotly

SOFTWARE REQUIREMENTS
Python, Anaconda, Jupyter Notebook or JupyterLab

Interactive Visualization with Bokeh

4 hours of instruction

Students will learn interactive plotting techniques using Bokeh. By the end of the course, students will be able to generate simple plots and integrate widgets with Bokeh.

OBJECTIVES	PREREQUISITES
1. Construct interactive choropleth maps 2. Generate simple plots using Bokeh 3. Integrate widgets to bokeh and plotly graphs	Interactive Visualization with Plotly

SYLLABUS & TOPICS COVERED
• Interactive Plots   • Interactive visualization use cases   • Organize and visualize data with bokeh

SOFTWARE REQUIREMENTS
Python, Anaconda, Jupyter Notebook or JupyterLab

Intro to Classification

4 hours of instruction

This course covers an introduction to classification algorithms to categorize a given set of data into classes.This course includes different classification algorithms like kNN. The course includes the background, how to build, evaluate and interpret these models.

OBJECTIVES	PREREQUISITES
1. Identify opportunities to use supervised learning 2. Build classification models to anticipate events and behaviors 3. Evaluate accuracy of classification algorithms	Data Wrangling in Python

SYLLABUS & TOPICS COVERED

- kNN
  - Supervised learning and its use cases
  - The theory behind kNN algorithm
  - Implementation of kNN on a dataset
  - Performance optimization for kNN

SOFTWARE REQUIREMENTS

Anaconda, Python, Jupyter Notebooks

Logistic Regression

4 hours of instruction

This course covers the supervised learning approach in which the computer program learns to categorize a given set of data into classes. It can be performed on structured data. This course includes different classification algorithms like logistic regression. The course includes the background, how to build, evaluate and interpret these models.

OBJECTIVES	PREREQUISITES
1. Understand Logistic regression and its use cases 2. Implement a logistic predictive model 3. Understand and improve performance of the model	Intro to Classification

SYLLABUS & TOPICS COVERED

- Logistic Regression
  - Logistic regression use cases and theory behind it
  - Data transformation necessary for logistic regression
  - Implementation of logistic regression on a dataset
  - Model performance evaluation and tuning

SOFTWARE REQUIREMENTS

Anaconda, Python, Jupyter Notebooks

Decision Trees

4 hours of instruction

This course covers the supervised learning approach of decision trees. It includes the concepts, how to build, evaluate and interpret the models.

OBJECTIVES	PREREQUISITES
1. Understand concepts and mathematics of decision trees 2. Implement, evaluate and optimize the decision tree model	Logistic Regression

SYLLABUS & TOPICS COVERED
• Decision Trees   • Decision trees use cases and theory behind it   • Data transformation necessary for decision trees   • Implementation of decision trees on a dataset   • Model performance evaluation and tuning

SOFTWARE REQUIREMENTS
Anaconda, Python, Jupyter Notebooks

Ensemble Methods

4 hours of instruction

This course covers an overview of ensemble learning methods like random forest and boosting. At the end of this course, students will be able to implement and compare random forest algorithm and boosting.

OBJECTIVES	PREREQUISITES
1. Build random forest and gradient boosting models 2. Compare the different methods and evaluate performance	Decision Trees

SYLLABUS & TOPICS COVERED

- Random Forest
  - Ensemble methods use cases
  - Random Forests algorithm in a nutshell
  - Implement Random Forests on a dataset
- Gradient Boosting
  - Gradient boosting algorithm in a nutshell
  - Implement gradient boosting on a dataset

SOFTWARE REQUIREMENTS

Python, Anaconda, Jupyter Notebook or JupyterLab

Optimizing Ensemble Methods

4 hours of instruction

This course covers advanced topics in optimizing ensemble learning methods – specifically random forest and gradient boosting. Students will learn to implement base models and perform hyperparameter tuning to enhance the performance of models.

OBJECTIVES	PREREQUISITES
1. Understand and interpret model results for a random forest classifier and gradient boosting algorithm 2. Fine tune parameter weights and perform advanced feature selection	Ensemble Methods

SYLLABUS & TOPICS COVERED

- Optimize RF
    - The need for algorithm performance optimization
    - Optimize Random Forest performance
- Optimize GBM
    - Optimize gradient boosting algorithm performance

SOFTWARE REQUIREMENTS

Python, Anaconda, Jupyter Notebook or JupyterLab

Support Vector Machines

4 hours of instruction

This course covers the basics of Support Vector machine algorithm. It helps students implement and optimize the model for a dataset.

OBJECTIVES	PREREQUISITES
1. Implement support vector machine algorithm 2. Understand hyperplanes and maximal margin classifier 3. Optimize the algorithm using grid search	Intro to Classification

SYLLABUS & TOPICS COVERED

- Support Vector Machines
    - Support vector machines algorithm use case and logic behind it
    - Implement SVM on a dataset

SOFTWARE REQUIREMENTS

Python, Anaconda, Jupyter Notebook or JupyterLab

Intro to NLP

4 hours of instruction

This course covers the basics of natural language processing, equipping learners with the ability to clean and process large amounts of text data required for text analysis.

OBJECTIVES	PREREQUISITES
1. Describe how text mining can be used effectively in commercial applications and industry 2. Process, clean, and format text data for analysis 3. Extract key summary metrics and words from a corpus of documents	Data Wrangling in Python

SYLLABUS & TOPICS COVERED

- Intro To NLP
  - NLP definition and use cases
  - Text analysis terminology
  - What is text processing
  - Text processing steps
  - Term Document matrix
  - Word distribution in a corpus
- Text Processing
  - What is text processing
  - Text processing steps
  - Term Document matrix
  - Word distribution in a corpus

SOFTWARE REQUIREMENTS

Anaconda, Python, Jupyter Notebooks

Topic Modeling in NLP

6 hours of instruction

This course intermediate concepts in natural language processing, equipping learners with the ability to clean and process large amounts of text data, segregating text into different groups and topics, as well as finding similarities between different documents. As natural language can be vague and subjective, the course also presents ways to evaluate and interpret these language models.

OBJECTIVES	PREREQUISITES
1. Understand and implement bag of words and term frequency inverse document frequency (TF-IDF) 2. Process, clean, and format text data for analysis 3. Extract key summary metrics and words from a corpus of documents 4. Perform latent dirichlet allocation (LDA) for topic modelling	Intro to NLP

SYLLABUS & TOPICS COVERED

- Tf Idf
  - "Bag-of-words" approach
  - Weighting terms in a corpus
  - Implementation of Tf-Idf weighting
- Topic Modeling
  - "Bag-of-words" approach
  - Weighting terms in a corpus
  - Implementation of Tf-Idf weighting
  - Topic modeling
  - Latent Dirichlet Allocation as topic modeling algorithm
  - Implementation of LDA

SOFTWARE REQUIREMENTS

Anaconda, Python, Jupyter Notebooks

Clustering in NLP

4 hours of instruction

This course covers the clustering concepts of natural language processing, equipping learners with the ability to cluster text data into groups and topics by finding similarities between different documents.

OBJECTIVES	PREREQUISITES
1. Understand measures of similarity and distance 2. Learn and implement cosine similarity on text documents 3. Understand how similar documents can be clustered into topics	Topic Modeling in NLP

SYLLABUS & TOPICS COVERED

- Cosine Similarity
  - Measures of similarity and distance
  - Theory and implementation of cosine similarity find most similar documents
- Clustering Documents
  - Hierarchical clustering algorithm in a nutshell

SOFTWARE REQUIREMENTS

Anaconda, Python, Jupyter Notebooks

Word Embeddings in NLP

4 hours of instruction

This course covers the intermediate concepts of natural language processing like creating word embeddings, feature engineering and word embeddings for finding text features for model development.

OBJECTIVES	PREREQUISITES
1. Understand feature engineering in text analysis 2. Create word embeddings and learn to use pre-trained embeddings like GloVe 3. Compute text similarity based on the embeddings	Topic Modeling in NLP

SYLLABUS & TOPICS COVERED

- Word Embeddings
    - Feature engineering in text analysis
    - Word embeddings: creating new ones vs using pre-trained

SOFTWARE REQUIREMENTS

Anaconda, Python, Jupyter Notebooks

Sentiment Analysis in NLP

4 hours of instruction

This course covers the intermediate concepts of natural language processing like sentiment analysis. By the end of this course students will be able to build classification model for textual data.

OBJECTIVES	PREREQUISITES
1. Understand the concept of sentiment analysis 2. Initialize, build and train a logistic regression model that can be used to find sentiments in a document	Word Embeddings in NLP

SYLLABUS & TOPICS COVERED

- Sentiment Analysis
    - Sentiment analysis in a nutshell
    - Classifying text according to sentiment

SOFTWARE REQUIREMENTS

Anaconda, Python, Jupyter Notebooks

Intro to Clustering

4 hours of instruction

This course covers the unsupervised learning method called clustering which is used to find patterns or groups in data without the need for labelled data. This course includes applying clustering on numerical data using centroid-based methods like K-Means, equipping learners to build, evaluate and interpret these models.

OBJECTIVES	PREREQUISITES
1. Mine data to find latent patterns and groups on numerical data using K-Means clustering 2. Evaluate the accuracy and effectiveness of clustering 3. Identify use cases where clustering analyses are relevant and where they are not applicable	Data Wrangling in Python

SYLLABUS & TOPICS COVERED

- K Means
  - Unsupervised learning and its use cases
  - The theory behind KMeans algorithm
  - Implementation of KMeans on a dataset

SOFTWARE REQUIREMENTS

Anaconda, Python, Jupyter Notebooks

Intermediate Clustering

4 hours of instruction

This is an intermediate course that covers clustering algorithms which is used to find patterns or groups in data without the need for labelled data. This course includes different methods of clustering on numerical data including density-based and hierarchical-based clustering and how to build, evaluate and interpret these models.

OBJECTIVES	PREREQUISITES
1. Mine data to find latent patterns and groups on numeric data using DBSCAN and Hierarchical clustering 2. Evaluate the accuracy and effectiveness of clustering 3. Understand the purpose and implications of what clustering methods can and cannot achieve	Intro to Clustering

SYLLABUS & TOPICS COVERED

- Hierarchical
  - The theory behind Hierarchical clustering
  - Implementation of Hierarchical clustering and comparison to other methods
- DBSCAN
  - The theory behind DBSCAN algorithm
  - Implementation and optimization of DBSCAN

SOFTWARE REQUIREMENTS

Anaconda, Python, Jupyter Notebooks

Advanced Clustering

4 hours of instruction

This course covers the advanced concepts in clustering which is used to find patterns or groups in data without the need for labelled data. This course includes applying different methods of clustering like K-Modes, MeanShift and K-Prototypes.

OBJECTIVES	PREREQUISITES
1. Mine data to find latent patterns and groups on categorical data using KModes, MeanShift and KPrototypes 2. Evaluate the accuracy and effectiveness of the models 3. Identify use case of each of the clustering methods	Intermediate Clustering

SYLLABUS & TOPICS COVERED

- K Modes
  - KModes use cases and theory
  - Implementation of KModes on a dataset
- Mean Shift
  - MeanShift use cases and theory
  - Implementation of MeanShift on a dataset
- K Prototypes
  - KPrototypes use cases and theory
  - Implementation of KPrototypes on a dataset

SOFTWARE REQUIREMENTS

Anaconda, Python, Jupyter Notebooks

Feature Engineering

4 hours of instruction

This course helps students to identify the most impactful features for your model. It will build upon foundational machine learning techniques to hone predictive skills and discover critical danger points in patterns. By the end of this course, students will be able to determine key features in models.

OBJECTIVES	PREREQUISITES
1. Define use cases for feature engineering 2. Identify and evaluate the most impactful numerical and categorical variables	Intro to Clustering

SYLLABUS & TOPICS COVERED
• Feature Engineering • Feature engineering definition and use cases • Feature engineering on different types of data • Implement feature engineering techniques on numeric, categorical, temporal, and spatial data

SOFTWARE REQUIREMENTS
Python & Anaconda

Intro to Outlier Detection

4 hours of instruction

Detecting outlier data points are powerful machine learning techniques. This class will build upon foundational machine learning techniques to discover critical danger points in patterns. By the end of this course, students will use techniques like DBSCAN and SMOTE to identify anomalous data points

OBJECTIVES	PREREQUISITES
1. Define use cases for anomaly and outlier detection 2. Understand the concepts of DBSCAN and SMOTE models and how these models can be used for anomaly and outlier detection	Intro to Clustering

SYLLABUS & TOPICS COVERED

- Anomaly Basics
  - Anomaly detection basics and use cases
  - Definition of different types of outliers / anomalies
- DBSCAN
  - Use DBSCAN as an anomaly detection technique
- SMOTE
  - Role of SMOTE in anomaly detection

SOFTWARE REQUIREMENTS

Python & Anaconda

Intermediate Outlier Detection

4 hours of instruction

Detecting outlier data points are powerful machine learning techniques. This course covers how techniques like Local Outlier Factor and Isolation Forest play a role in anomaly and outlier detection. By the end of the course, students will learn to implement these techniques to identify anomalous data points

OBJECTIVES	PREREQUISITES
1. Understand the concepts and working of specific outlier detection algorithms like LOF and Isolation Forest 2. Implement and optimize these models to identify anomalies in a dataset	Intro to Outlier Detection

SYLLABUS & TOPICS COVERED
• LOF • Local Outlier Factor algorithm use cases and logic behind it • Implement and optimize LOF on a dataset • Isolation Forest • Isolation forest algorithm use cases and logic behind it • Implement and optimize Isolation Forest on a dataset

SOFTWARE REQUIREMENTS
Python & Anaconda

Outlier Detection for Time Series

4 hours of instruction

Detecting outlier data points are powerful machine learning techniques. This covers covers the concepts and models used in Time Series Analysis, and how these models can be used in anomaly detection. By the end of this course, students will be able to use ARIMA model to identify anomalous data points in a time series dataset

OBJECTIVES	PREREQUISITES
1. Introduce the concepts used in Time Series analysis 2. Understand the working of time series models and how they can be used in anomaly and outlier detection 3. Develop accurate anomaly detection model using ARIMA	Intermediate Outlier Detection

SYLLABUS & TOPICS COVERED

- Time Series
  - Time series analysis basics
  - AR, MA, and ARIMA models in a nutshell
  - ARIMA use cases and its role in anomaly detection for time series data

SOFTWARE REQUIREMENTS

Python & Anaconda

Intro to Time Series Analysis

8 hours of instruction

Learn how to apply time series basics and concepts to create accurate forecasts for their organizations and make better decisions when developing strategies.

OBJECTIVES	PREREQUISITES
1. Introduce time series analysis and understand the components and core concepts used in time series analysis 2. Process, clean, and format time series data for analysis	Data Wrangling in Python

SYLLABUS & TOPICS COVERED

- Basics
  - Time series analysis definition and use cases
  - What makes data a time series
  - Basics and components of time series modeling
  - Visualize time series data
- Concepts
  - Core concepts in time series analysis: random walk, stationarity, moving averages, trend, and seasonality
  - Deconstruct time series into its components

SOFTWARE REQUIREMENTS

Python & Anaconda

ARIMA

4 hours of instruction

Learn how to apply seasonal analysis and ARIMA models and how to decompose and identify seasonal and non-seasonal factors all while learning the nuances of building sophisticated time series models.

OBJECTIVES	PREREQUISITES
1. Build time series models to identify and predict trends 2. Define key metrics of complex time series models 3. Develop models that account for seasonal trends and other factors 4. Build and evaluate ARIMA models	Intro to Time Series Analysis

SYLLABUS & TOPICS COVERED

- Modeling
  - AR, MA, and ARIMA models in a nutshell
  - Implement AR, MA, and ARIMA on a time series

SOFTWARE REQUIREMENTS

Python & Anaconda

Simple Linear Regression

4 hours of instruction

This course covers a supervised regression technique called Simple Linear which is used to model a relationship between a single feature and a continuous target variable. Students will learn how this relationship is then used to predict changes in the target variable. The course includes the background, how to build, evaluate and interpret these Simple Linear regression models.

OBJECTIVES	PREREQUISITES
1. Understand the working and identify use cases of simple linear regression models 2. Build and evaluate simple regression models 3. Assess statistical significance and validate these models for explanatory power and bias	Data Wrangling in Python

SYLLABUS & TOPICS COVERED

- Simple Linear
  - Simple linear regression definition and use cases
  - Theory behind simple linear regression
  - Simple linear regression implementation on a dataset

SOFTWARE REQUIREMENTS

Anaconda, Python, Jupyter Notebooks

Multiple Linear Regression

4 hours of instruction

This course covers a supervised regression technique called Multiple regression which is used to model a relationship between a certain number of features and a continuous target variable. Students will learn how this relationship is then used to predict changes in the target variable. The course includes the foundations of Multiple regression models, how to build, evaluate and interpret these models.

OBJECTIVES	PREREQUISITES
1. Understand the theory behind Multiple Linear Regression and identify its use cases 2. Build, optimize and evaluate these multiple regression models 3. Assess statistical significance and validate models for explanatory power and bias 4. Cover techniques used to identify influential points and correlated variables	Simple Linear Regression

SYLLABUS & TOPICS COVERED

- Multiple Linear
  - Multiple linear regression definition and use cases
  - Theory behind multiple linear regression
  - Multiple linear regression implementation on a dataset

SOFTWARE REQUIREMENTS

Anaconda, Python, Jupyter Notebooks

Nonlinear Regression

4 hours of instruction

This course covers a supervised regression technique called Non Linear regression which is used to model a relationship between a certain number of features and a continuous target variable. Students will learn how this relationship is then used to predict changes in the target variable. The course includes the background, how to build, evaluate and interpret these Non Linear Regression models.

OBJECTIVES	PREREQUISITES
1. Understand the working and identify use cases of nonlinear regression models 2. Build, optimizing and evaluate these nonlinear regression models 3. Assess statistical significance and validate models for explanatory power and bias	Multiple Linear Regression

SYLLABUS & TOPICS COVERED
• Nonlinear   • Optimizing and evaluating nonlinear regression models   • Understanding how to model interactions

SOFTWARE REQUIREMENTS
Anaconda, Python, Jupyter Notebooks

Recommender Systems

6 hours of instruction

This course provides an overview of how recommender systems work and teaches students how to build effective models. By the end of this course, students will be able to explain the key assumptions underlying recommender systems and build and evaluate them based on real data.

OBJECTIVES	PREREQUISITES
1. Identify and define use cases for recommender systems 2. Build and evaluate content-based recommender systems 3. Build and evaluate item-based filtering algorithm	Data Wrangling in Python

SYLLABUS & TOPICS COVERED

- Basics
  - Recommender systems use cases and logic behind them
  - Data processing for recommender system
- Content Based Recommenders
  - Content-based recommender system use cases and logic behind it
  - Generate recommendations using content-based recommender system
- Collaborative Filtering
  - Collaborative filtering use cases and logic behind it
  - SVD and its role in recommender systems
  - Generate recommendations using collaborative filtering recommender system

SOFTWARE REQUIREMENTS

Python, Anaconda, Jupyter Notebook or JupyterLab

Intro to Network Analytics

4 hours of instruction

This course introduces the concepts of networks, network graphs and network types. By the end of this course, students will be able to understand and visualize the connections between network nodes.

OBJECTIVES	PREREQUISITES
1. Understand what networks are and their use cases 2. Build and visualize foundational network graphs 3. Measure and evaluate the strengths and attributes of networks	Data Wrangling in Python

SYLLABUS & TOPICS COVERED
• Intro To Networks   • Network analysis case studies   • Definition of a network graph and its components   • Build and visualize a network graph from a dataset

SOFTWARE REQUIREMENTS
Python, Anaconda, Jupyter Notebook or JupyterLab

Intermediate Network Analytics

4 hours of instruction

This is an intermediate network analytics course. By the end of this course, students will be able to quantitatively measure and visualize network nodes.

OBJECTIVES	PREREQUISITES
1. Measure and evaluate a network 2. Build and visualize foundational network graphs 3. Measure and evaluate the strengths and attributes of networks	Intro to Network Analytics

SYLLABUS & TOPICS COVERED
• Network Metrics   • Define ways to measure and evaluate a network   • Calculate and interpret various centrality metrics • Network Visualization   • Trim a network based on its attributes   • Visualize the resulting network graph • Network Resilience   • What network resilience is and how to test it   • Perform a network resilience test

SOFTWARE REQUIREMENTS
Python, Anaconda, Jupyter Notebook or JupyterLab

Network Analytics & Community Detection

4 hours of instruction

This course covers the basic concepts of community detection and how it relates to network analysis. The students will be able to implement and compare different community detection algorithms.

OBJECTIVES	PREREQUISITES
1. Define what community detection is and its use cases 2. Introduce and implement Louvain community detection algorithm 3. Explain and implement label propagation algorithm	Intermediate Network Analytics

SYLLABUS & TOPICS COVERED

- Community Detection
  - Community detection use cases and methods
  - Implement and compare different community detection algorithms

SOFTWARE REQUIREMENTS

Python, Anaconda, Jupyter Notebook or JupyterLab

DATA SOCIETY

Data Science (R & SQL)

Data Science (R & SQL)

A pathway that takes students through the courses that span the topics in Data Science in R and SQL. The pathway starts with the very foundations of R and SQL and walks the learners through key machine learning algorithms implemented in one of the most widely used languages in statistics and research community - R. This pathway equips learners with a wide range of tools and frameworks to become data scientists and communicate their findings through effective visuals and model summaries.

Courses Include

- **Intro to SQL**
- **Intermediate SQL**
- **Intro to R**
- **Intermediate R**
- **Data Wrangling in R**
- **Intro to Visualization in R**
- **Interactive Visualization with R**
- **RShiny Apps**
- **Simple Linear Regression in R**
- **Multiple Linear Regression in R**
- **Intro to Clustering in R**
- **Intermediate Clustering in R**
- **Advanced Clustering in R**
- **And more...!**

DATA SOURCES & EXAMPLES USED

- Industry or domain-specific examples
- Customized data that maximizes the attendees' engagement and retention of skills and can be applied directly within your organization

PATH DURATION

- 70 Hours, approximately
- Available in flexible schedule format

MODALITY

- Proposed live-streaming (in-person available post-COVID-19)
- Instructor-led (TAs are optional)

RECOMMENDATIONS

- Up to 30 learners per cohort
- Review individual course prerequisites and software requirements

Intro to SQL

4 hours of instruction

Students will learn fundamental concepts in database design, database modeling techniques, and Structured Query Language (SQL) programming techniques while providing hands-on exercises in which students apply these concepts and techniques to real-world problems. An easy-to-learn programming language that takes care of tedious data collection and cleaning tasks. By the end of this course, students will be able to import and manipulate data in SQL.

OBJECTIVES	PREREQUISITES
1. Introduce conceptual, logical, and physical design of relational database systems and their use case in business environments 2. Create SQL databases and populate and maintain/use data in the database	Git

SYLLABUS & TOPICS COVERED
• Intro To SQL   • Define structure of the data stored in the database   • Manipulate data defined in the database

SOFTWARE REQUIREMENTS
Snowflake, Jupyter Notebook or JupyterLab

Intermediate SQL

4 hours of instruction

Students will learn advanced concepts in database design, database modeling techniques, and Structured Query Language (SQL) programming techniques while providing hands-on exercises in which students apply these concepts and techniques to real-world problems. An easy-to-learn programming language that takes care of tedious data collection and cleaning tasks. By the end of this course, students will be able to import and manipulate data in SQL.

OBJECTIVES	PREREQUISITES
1. Create SQL databases and populate and maintain/use data in the database 2. Program proficiently in SQL 3. Automate data cleaning and processing	Intro to SQL

SYLLABUS & TOPICS COVERED

- Intermediate SQL
    - Use aggregate functions to generate mathematical outputs
    - Group data for easier and specific analysis
    - Implement complex, nested queries

SOFTWARE REQUIREMENTS

Snowflake, Jupyter Notebook or JupyterLab

Intro to R

4 hours of instruction

This course will give students the foundational tools they need to work with data in R.

OBJECTIVES	PREREQUISITES
1. Identify what data science is and why it's important 2. Program proficiently in R 3. Identify basic data structures in R	Git

SYLLABUS & TOPICS COVERED

- Basics
  - Overview of Data Science and its tools
  - R and RStudio as tools in data analysis and their features
  - Basic calculations in R
- Data Types And Data Structures
  - Overview of R data types and data structures
  - Lists, vectors, matrices, and dataframes

SOFTWARE REQUIREMENTS

R & RStudio

Intermediate R

4 hours of instruction

This course builds on the introduction to R and introduces key programming concepts like control flow, functions, and working with R environment.

OBJECTIVES	PREREQUISITES
1. Discuss control flow structures and practice of writing of modular code 2. Implement for loops and functions 3. Create simple programs in R	Intro to R

SYLLABUS & TOPICS COVERED
• Load Data And Environment   • Load data into R   • Manipulate variables within R environment • Control Flow And Functions   • Controlling the flow of a program   • Conditionals, loops, and functions

SOFTWARE REQUIREMENTS
R & RStudio

Data Wrangling in R

4 hours of instruction

This course introduces students to R's tools to work with datasets. By the end of this course students will be able to load, wrangle, transform, and summarize data.

OBJECTIVES	PREREQUISITES
1. Define the functions that provide verbs for the language of data manipulation 2. Summarize columns using the group by function 3. Transform messy data to tidy data using tidyr package	Intermediate R

SYLLABUS & TOPICS COVERED

- Data Wrangling
  - Tidy data best practices
  - Transform data with tidyr
  - Data wrangling basics
  - Use dplyr for data wrangling
- Data Summarization And Transformation
  - Tidy data best practices
  - Transform data with tidyr

SOFTWARE REQUIREMENTS

R & RStudio

Intro to Visualization in R

6 hours of instruction

Students will take their R skills to the next level and learn how to effectively communicate the results of their analysis with a few lines of code. They'll develop a powerful toolkit to visualize data with static graphs.

OBJECTIVES	PREREQUISITES
1. Explain use cases for Exploratory Data Analysis (EDA) 2. Create various static visualizations with R	Data Wrangling in R

SYLLABUS & TOPICS COVERED

- Exploratory Data Analysis
  - Exploratory data analysis use cases
  - Perform EDA on data
- Basic Data Visualization
  - Data visualization basics and use cases
  - Build plots with base r
- Static Plots
  - The process behind constructing complex plots
  - Visualizing data with ggplot2

SOFTWARE REQUIREMENTS

R & RStudio

Interactive Visualization with R

8 hours of instruction

Students will take their R graphing skills to the next level and learn how to effectively communicate the results of their analysis through interactive graphs. By the end of the course, students will be able to create interactive visualizations for various types of data.

OBJECTIVES	PREREQUISITES
1. Import, export, clean, and manipulate data in R 2. Create interactive visualizations with transformed data 3. Create various interactive charts for different types of data	Intro to Visualization in R

SYLLABUS & TOPICS COVERED

- Interactive Plotting Libraries
  - Interactive visualization use cases
  - Visualizing data with highcharter
- Interactive Plots
  - Various plot types available in highcharter library
  - Creating scatterplots, correlation plots, column plots, and boxplots
  - Saving and viewing interactive plots by using htmlwidgets library
- Summary Data Visualization
  - Transforming and summarizing data for visualization
  - Visualizing summary data
- Interactive Maps
  - Transforming and preparing data for maps
  - Creating maps and displaying spatial data over time
- Network Graphs
  - Transforming and preparing data for network graph visualizations
  - Building and customizing interactive network graphs

SOFTWARE REQUIREMENTS

R & RStudio

RShiny Apps

4 hours of instruction

Students will take their R skills to the next level and learn how to effectively communicate the results of their analysis by creating interactive dashboards in RShiny. By the end of this course students will be able to create and publish dashboards.

OBJECTIVES	PREREQUISITES
1. Identify RShiny tools and discuss how they improve user experience 2. Import, export, clean, and manipulate data in R 3. Create and publish a dashboard using RShiny	Interactive Visualization with R

SYLLABUS & TOPICS COVERED

- R Shiny Overview
  - Interactive dashboards and their role in effectively presenting data to end users
  - RShiny library and framework as a single tool needed to build complex visual displays with interactive features
- R Shiny Setup And Features
  - RShiny structure and key components
  - Controls, widgets, and various types of inputs and their functionality
- R Shiny Reactive Output
  - Introduce the concept of reactivity and user interactions
  - Configuration of server logic based on user input

SOFTWARE REQUIREMENTS

R & RStudio

Simple Linear Regression in R

4 hours of instruction

Learn how to predict trends with regression models, a supervised machine learning technique. This program combines the theoretical basis necessary to perform regression analysis with hands-on practice applying regression to a variety of real-world problems. By the end of this course, students will be able to identify use cases for regression, build simple regression models, and evaluate their results.

OBJECTIVES	PREREQUISITES
1. Identify opportunities and use cases for regression models 2. Build and evaluate simple linear regression models 3. Assess statistical significance and validate models for explanatory power and bias	Data Wrangling in R

SYLLABUS & TOPICS COVERED

- Simple Linear
  - Regression use cases
  - Simple linear regression in a nutshell
  - Implement simple linear regression on a dataset

SOFTWARE REQUIREMENTS

R & RStudio

Multiple Linear Regression in R

4 hours of instruction

This program combines the theoretical basis necessary to perform multiple regression analysis with hands-on practice applying regression to a variety of real-world problems. Multiple regression is used to model a relationship between a certain number of features and a continuous target variable. This relationship is then used to predict changes in the target variable. By the end of this course, students will be able to identify use cases for multiple regression, build regression models, and evaluate their results.

OBJECTIVES	PREREQUISITES
1. Identify opportunities and use cases for regression models 2. Build and evaluate multiple regression models 3. Assess statistical significance and validate models for explanatory power and bias	Simple Linear Regression in R

SYLLABUS & TOPICS COVERED
• Multiple Linear   • Multiple linear regression in a nutshell   • Implement multiple linear regression on a dataset

SOFTWARE REQUIREMENTS
R & RStudio

Intro to Clustering in R

4 hours of instruction

Learn how to mine data and uncover patterns within it. Clustering is a foundational unsupervised machine learning technique that is key to discovering underlying patterns and trends. This course includes applying clustering on numerical data using centroid-based methods called K-Means, equipping learners to build, evaluate, and interpret these models.

OBJECTIVES	PREREQUISITES
1. Mine data to find latent patterns and groups in different types of data 2. Implement and visualize k-means algorithm 3. Evaluate the effectiveness of clustering 4. Understand the purpose and implications of what clustering methods can and cannot achieve	Data Wrangling in R

SYLLABUS & TOPICS COVERED

- K Means
  - Clustering algorithms use cases
  - k-means algorithm in a nutshell
  - Implement k-means on data

SOFTWARE REQUIREMENTS

R & RStudio

Intermediate Clustering in R

4 hours of instruction

This course covers the unsupervised learning method called clustering which is used to find patterns or groups in data without the need for labelled data. This course includes different methods of clustering on numerical data including density-based and hierarchical-based clustering and how to build, evaluate and interpret these models.

OBJECTIVES	PREREQUISITES
1. Mine data to find latent patterns and groups in different types of data 2. Summarize the process behind agglomerative clustering and discuss the types of linkage methods 3. Summarize the process and implement density-based clustering (DBSCAN) 4. Identify use cases where clustering analyses are relevant and where they are not applicable	Intro to Clustering in R

SYLLABUS & TOPICS COVERED

- Hierarchical
  - Hierarchical clustering algorithm in a nutshell
  - Implement Hierarchical clustering on a dataset
- DBSCAN
  - DBSCAN algorithm use case and logic
  - Implement DBSCAN on a dataset

SOFTWARE REQUIREMENTS

R & RStudio

Advanced Clustering in R

4 hours of instruction

This course covers the unsupervised learning method called clustering which is used to find patterns or groups in data without the need for labelled data. This course includes application of different methods of clustering on categorical or mixed data, equipping learners to build, evaluate, and interpret these models.

OBJECTIVES	PREREQUISITES
1. Mine data to find latent patterns and groups in different types of data 2. Implement Mean-Shift clustering on dataset 3. Describe the process behind k-modes and its use cases 4. Explore k-prototypes to cluster mixed data types 5. Identify use cases where clustering analyses are relevant and where they are not applicable	Intermediate Clustering in R

SYLLABUS & TOPICS COVERED

- Mean Shift
  - Mean-Shift algorithm use case and logic
  - Implement Mean-Shift on a dataset
- K Modes
  - K-Modes algorithm use case and logic
  - Implement K-Modes on a dataset
- K Prototypes
  - K-Prototypes algorithm use case and logic
  - Implement K-Prototypes on a dataset

SOFTWARE REQUIREMENTS

R & RStudio

Intro to Classification in R

4 hours of instruction

Classification algorithms are powerful and intuitive data science tools that can predict behaviors and trends. This course walks students through how to implement algorithms like k-Nearest Neighbors, as well as evaluate and interpret their results. By the end of the course, students will be able to build classification models to anticipate events and assess the accuracy of predictive algorithms.

OBJECTIVES	PREREQUISITES
1. Identify opportunities and use cases for predictive analytics 2. Build kNN classification model to anticipate events and behaviors 3. Evaluate accuracy of kNN algorithm	Data Wrangling in R

SYLLABUS & TOPICS COVERED

- kNN
    - Classification use cases
    - kNN algorithm in a nutshell
    - Implement kNN on a dataset

SOFTWARE REQUIREMENTS

R & RStudio

Logistic Regression in R

4 hours of instruction

This course builds on the foundational knowledge of classification algorithms and takes a deeper dive into logistic regression - one of the most powerful and widely used classification method. By the end of the course, students will be able to build, evaluate, and tune logistic regression models in R.

OBJECTIVES	PREREQUISITES
1. Identify opportunities and use cases for classification algorithms 2. Summarize the process and the math behind logistic regression 3. Assess results of classification model performance 4. Tune the model using grid search cross-validation	Intro to Classification in R

SYLLABUS & TOPICS COVERED

- Logistic Regression
  - Logistic regression use cases and logic behind it
  - Implementation of logistic regression on a dataset
  - Evaluation of the results and tuning the model

SOFTWARE REQUIREMENTS

R & RStudio

Decision Trees in R

4 hours of instruction

This course builds on the foundational knowledge about classification techniques and introduces students to the concept of decision trees. By the end of the course, students will be able to build decision tree models, evaluate their performance and tune them to achieve optimal results.

OBJECTIVES	PREREQUISITES
1. Identify opportunities and use cases for decision trees 2. Build classification models to anticipate events and behaviors 3. Implement decision tree on a dataset and evaluate its results 4. Optimize the decision tree by tuning the hyperparameters	Intro to Classification in R

SYLLABUS & TOPICS COVERED

- Decision Trees
  - Decision trees algorithm use cases and logic behind it
  - Implementation of decision trees algorithm on a dataset
  - Evaluation of model performance
  - Tuning and optimization of the decision tree model

SOFTWARE REQUIREMENTS

R & RStudio

Front End

Front End

A pathway that takes students through the courses that span the topics in Frontend engineering. The pathway starts with very foundations of HTML and CSS and ends with the advanced topics on React and equips the learners with a wide range of tools and frameworks to become frontend engineers.

Courses Include

- **Web Security Knowledge**
- **Intro to HTML, Part 1**
- **Intro to HTML, Part 2**
- **Advanced Git**
- **Intro to CSS, Part 1**
- **Intro to CSS, Part 2**
- **Accessibility**
- **SEO Basics**
- **CSS Architecture**
- **CSS Preprocessors**
- **Intro to JavaScript Syntax & Basic Constructs**
- **Intro to JavaScript: the DOM**
- **Intro to JavaScript: Fetch Async Await**
- **And more...!**

DATA SOURCES & EXAMPLES USED

- Industry or domain-specific examples
- Customized data that maximizes the attendees' engagement and retention of skills and can be applied directly within your organization

PATH DURATION

- 164 Hours, approximately
- Available in flexible schedule format

MODALITY

- Proposed live-streaming (in-person available post-COVID-19)
- Instructor-led (TAs are optional)

RECOMMENDATIONS

- Up to 30 learners per cohort
- Review individual course prerequisites and software requirements

Web Security Knowledge

4 hours of instruction

A course that builds a foundational understanding of web security tools and best practices for developers.

OBJECTIVES	PREREQUISITES
1. Describe the need for securing web content 2. Create and manage web security policies for a given website	Domain & Hosting

SYLLABUS & TOPICS COVERED

- HTTPS
  - What is HTTPS?
  - SSL Certificates
- Content Security Policy
  - The risks of cross-site scripting
  - What is CSP?
  - Making it work
- CORS
  - What is CORS?
  - Understanding CORS policies
  - How and where to set them
- OWASP Security Risks
  - Understanding application security risks
  - Tools, checklists and other resources

SOFTWARE REQUIREMENTS

browser, text editor

Intro to HTML, Part 1

4 hours of instruction

A course that builds a foundational understanding of HTML syntax and allows students to have hands-on experience in building a basic, static webpage.

OBJECTIVES	PREREQUISITES
1. Describe the inner workings of HTML 2. Be able to create and edit webpages	Domain & Hosting

SYLLABUS & TOPICS COVERED

- HTML Basics
  - Understand HTML syntax
  - Build a basic, static webpage
- Semantic HTML
  - Understand rationale for semantic HTML
  - Incorporate into basic, static webpage

SOFTWARE REQUIREMENTS

browser, text editor

Intro to HTML, Part 2

4 hours of instruction

A course that continues to build a foundational understanding of HTML syntax and allows students to have hands-on experience in building a basic, static webpage.

OBJECTIVES	PREREQUISITES
1. Describe the inner workings of HTML 2. Be able to take user inputs from webpages	Intro to HTML, Part 1

SYLLABUS & TOPICS COVERED

- Forms And Validations
  - Build a basic form
  - Understand and incorporate form validation
- Conventions And Best Practices
  - Why are conventions important?
  - Best practices of writing sound HTML

SOFTWARE REQUIREMENTS

browser, text editor

Advanced Git

4 hours of instruction

A course that builds an advanced understanding of Git and version control systems. By the end of the course students will be able to use key Git commands and interact with remote repositories like GitHub, Bitbucket, and GitLab for efficient tracking and maintaining of software code.

OBJECTIVES	PREREQUISITES
1. Describe the need for advanced version control 2. Create tags to mark important checkpoints in Git history 3. Learn to navigate to different checkpoints in Git history	Domain & Hosting

SYLLABUS & TOPICS COVERED

- Tagging
  - Create and delete tags
  - Checkout and push tags to remote
- Select Changes
  - Cherry pick commits
  - Create and apply diff changes
- Rebase
  - Rebase vs merge
  - Squash commits
- Staging Logging
  - Interactive staging
  - Logging

SOFTWARE REQUIREMENTS

browser, text editor

Intro to CSS, Part 1

4 hours of instruction

A course that builds a foundational understanding of CSS syntax and allows students to have hands-on experience in styling web pages.

OBJECTIVES	PREREQUISITES
1. Describe the inner workings of CSS basic concepts and layouts and how they are used to style and modify webpages 2. Be able to style webpages	Intro to HTML, Part 2

SYLLABUS & TOPICS COVERED

- CSS Basics
    - What is CSS
    - Classes and Identifiers
- CSS Layouts
    - Floats
    - Positioning
    - Display
    - Box Model
    - CSS Grid
    - Flexbox

SOFTWARE REQUIREMENTS

browser, text editor

Intro to CSS, Part 2

4 hours of instruction

A course that continues to build on the foundational understanding of CSS syntax and allows students to work with responsive design and media queries.

OBJECTIVES	PREREQUISITES
1. Describe the inner workings of responsive design and the ins and outs of media queries and how it is used to style and modify webpages 2. Be able to style webpages	Intro to CSS, Part 1

SYLLABUS & TOPICS COVERED

- Responsive Design
    - What is responsive design?
    - Breakpoints
    - Dev tools for responsive design
- Media Queries
    - What is a Media Query?
    - Media Query use cases and basics

SOFTWARE REQUIREMENTS

browser, text editor

Accessibility

4 hours of instruction

A course that builds a foundational understanding of the concept of website accessibility and allows students to explore and implement the standards needed to create accessible websites.

OBJECTIVES	PREREQUISITES
1. Explore and implement the standards needed to create accessible websites 2. Get familiar with popular accessibility tools and frameworks 3. Implement a set of accessibility standards on the example of a simple static website	Intro to HTML, Part 2

SYLLABUS & TOPICS COVERED
• Web Accessibility Standards   • Intro to web accessibility   • Understand accessibility standards • Web Accessibility Tools   • Accessibility tools (including Lighthouse)   • Implementation of core standards on the example of a static webpage

SOFTWARE REQUIREMENTS
browser, text editor

SEO Basics

4 hours of instruction

A course that builds a foundational understanding of basic concepts of Search Engine Optimization (SEO) and best practices of effective SEO.

OBJECTIVES	PREREQUISITES
1. Describe the basic concepts of SEO and why it is important 2. Explore best SEO practices and the common do's and don'ts of the SEO	Intro to HTML, Part 2

SYLLABUS & TOPICS COVERED

- SEO Core Concepts
  - What is SEO?
  - Basic SEO strategies
- SEO Best Practices
  - Best practices of effective SEO
  - SEO do's and don'ts

SOFTWARE REQUIREMENTS

browser, text editor

CSS Architecture

4 hours of instruction

A course that builds on the foundations of CSS and dives into the exploration of CSS architecture.

OBJECTIVES	PREREQUISITES
1. Learners will improve their CSS skills and build on the foundational concepts 2. They will be able to navigate through the basic constructs of CSS architecture	Intro to CSS, Part 2

SYLLABUS & TOPICS COVERED

- CSS Architecture Basics
  - CSS architecture best practices
  - Separation, categorization, and modularization of code
- BEM Basics
  - Component definitions
  - Block, element, modifier as the key to the clean and efficient CSS

SOFTWARE REQUIREMENTS

browser, VSCode or another IDE, terminal

CSS Preprocessors

4 hours of instruction

A course that builds on the architecture of CSS and provides learners with a toolset for creating custom stylesheets and enhancing the look of standard webpages.

OBJECTIVES	PREREQUISITES
1. Learners will improve their CSS skills and build on the foundational concepts and the knowledge of CSS architecture 2. They will be able to build custom stylesheets that will enhance interactive websites	CSS Architecture

SYLLABUS & TOPICS COVERED
• Sass   • Preprocessors as the solution to the growing stylesheet problem   • Sass as a stylesheet language that compiles to CSS   • Sass examples • Post CSS   • PostCSS as a tool for transforming styles with JS plugins   • PostCSS examples

SOFTWARE REQUIREMENTS
browser, VSCode or another IDE, terminal

Intro to JavaScript Syntax & Basic Constructs

4 hours of instruction

A course that builds a foundational understanding of JavaScript basic constructs and syntax, and allows students to have hands-on experience building JavaScript driven websites.

OBJECTIVES	PREREQUISITES
1. Describe the core constructs of JavaScript and how it is used to create dynamic modern websites 2. Be able to create JavaScript-driven websites	Intro to CSS, Part 2

SYLLABUS & TOPICS COVERED

- Data Types And Operators
    - Data types
    - Operators
- Syntax Basics
    - Statements and expressions
    - Functions

SOFTWARE REQUIREMENTS

browser, text editor, VSCode

Intro to JavaScript: the DOM

4 hours of instruction

A course that builds on the basic constructs and syntax of JavaScript, and allows students to have hands-on experience of exploring the Document Object Model (DOM) structure.

OBJECTIVES	PREREQUISITES
1. Describe the core concepts of the DOM and its components 2. Be able to create JavaScript-driven websites	Intro to JavaScript Syntax & Basic Constructs

SYLLABUS & TOPICS COVERED
• DOM Basics • Intro to DOM and window objects • Getting and manipulating DOM elements

SOFTWARE REQUIREMENTS
browser, text editor, VSCode

Intro to JavaScript: Fetch Async Await

4 hours of instruction

A course that dives into the exploration of the frontend APIs, asynchronous calls and the concepts of modular JavaScript.

OBJECTIVES	PREREQUISITES
1. Explore the idea of asynchronous calls to frontend APIs 2. Use Ajax and Fetch API to implement a request cycle 3. Be able to build modular JavaScript code with the help of its functional and object-oriented tools	Intro to JavaScript: the DOM

SYLLABUS & TOPICS COVERED

- Ajax And Fetch API
  - What is an API?
  - RESTful conventions
  - Intro to HTTP request cycle
  - HTTP request structure
  - Intro to asynchronous functions
  - Using AJAX and Fetch

SOFTWARE REQUIREMENTS

browser, text editor, VSCode

Intro to JavaScript: Basic Concepts

6 hours of instruction

A course that finalizes the series of introductory JavaScript courses and introduces the students to the basic concepts in the JavaScript ecosystem.

OBJECTIVES	PREREQUISITES
1. Describe the basic concepts of JavaScript and best practices to create dynamic modern websites 2. Be able to create JavaScript-driven websites	Intro to JavaScript: Fetch Async Await

SYLLABUS & TOPICS COVERED

- Modular JavaScript And E S6
    - What are modules
    - Exporting and Importing
    - Functional vs Object-oriented Programming
    - The concept of "this"
- Hoisting And Event Bubbling
    - What is hoisting?
    - Event bubbling basics
    - Implementation of complex events that drive other events
- Scope And Prototypes
    - What is scope in JavaScript
    - The concept of a prototype object and how to use it

SOFTWARE REQUIREMENTS

browser, text editor, VSCode

Package Managers

4 hours of instruction

A course that builds a foundational understanding of package management tools for developers.

OBJECTIVES	PREREQUISITES
1. Describe the need for package management 2. Create and manage a simple code base using popular package managers	Intro to JavaScript: Basic Concepts

SYLLABUS & TOPICS COVERED

- Packages And Dependencies
    - Package vs dependency and why we need to maintain them
    - Package management frameworks
- NPM
    - What is NPM?
    - How to use package
    - json and package-lock
    - json
    - Listing and installing packages and dependencies
    - Installing from GIT/GIST and NPM registry
- YARN
    - What is yarn?
    - How to use yarn
    - json and yarn-lock
    - json

SOFTWARE REQUIREMENTS

browser, text editor, VSCode

Web Components

4 hours of instruction

A course that builds a foundational understanding of Web Components. Web Components give us the power to extend the web with our own rich, standards-based components.

OBJECTIVES	PREREQUISITES
1. Learn how to use the Shadow DOM 2. Define inert templates with the template tag 3. Extend HTML by registering your custom elements	Intro to JavaScript: Basic Concepts

SYLLABUS & TOPICS COVERED

- HTML Templates
  - Common Approaches
  - Template Characteristics
  - Defining and Cloning
  - Injecting Dynamic Data
  - Nested Templates
- Custom Elements
  - Core Functionality
  - Registering and Instantiating Custom Elements
  - Instantiating Extended Custom Elements
  - Lifecycle Callback Methods
  - Naming Approaches
- Shadow DOM
  - Light DOM vs
  - Shadow DOM
  - Shadow DOM Alternatives
  - Shadow Host and Shadow Boundary
  - JavaScript Is Not Encapsulated
  - Content Insertion Point
  - Insertion Points vs
  - Distributed Nodes
  - Content Selectors
  - Shadow Insertion Points
  - Listing Distributed Nodes and Destination Insertion Points
  - Event Retargeting
  - Events That Stop

SOFTWARE REQUIREMENTS

browser, text editor

Unit Testing in Jest

4 hours of instruction

Jest is a tool for testing React applications. This course will teach you basic and intermediate Jest testing techniques, including running tests, snapshot testing, testing React components, and module mocking.

OBJECTIVES	PREREQUISITES
1. Explore the anatomy of a Jest unit test, and begin writing code and subsequently, the tests for that code using the Jest framework 2. Be able to write test cases to validate that the software performs as designed	Intro to JavaScript: Basic Concepts

SYLLABUS & TOPICS COVERED

- Install
  - What is Jest?
  - Installing Jest
  - Configuring Jest
- Unit Testing
  - What is Jest?
  - Installing Jest
  - Configuring Jest
  - Testing Async Code with Jest (Part 1)
  - Testing Async Code with Jest (Part 2)
  - Mocking with Jest (Part 1)
  - Mocking with Jest (Part 2)
  - Unit Testing with Jest (Part 1)
  - Unit Testing with Jest (Part 2)
  - Matcher Functions
- Testing Async Mocking
  - Testing Async Code with Jest (Part 1)
  - Testing Async Code with Jest (Part 2)
  - Mocking with Jest (Part 1)
  - Mocking with Jest (Part 2)

SOFTWARE REQUIREMENTS

Text editor

Cypress

4 hours of instruction

Cypress is an end-to-end testing framework for your web application. This course explores its features, core concepts, its ecosystem, and how to write tests.

OBJECTIVES	PREREQUISITES
1. Explore the use of Cypress in writing automated tests in an easy, quick and reliable way, all in JavaScript 2. Be able to set up an end-to-end testing with Cypress	Unit Testing in Jest

SYLLABUS & TOPICS COVERED
• Introduction   • Introduction to Cypress   • Setting up Cypress • Core Concepts   • Writing and organizing tests   • Cypress test runner   • Cypress Studio • Tooling   • Plugins and tools for Cypress   • Code coverage

SOFTWARE REQUIREMENTS
browser, text editor

Test-driven Development

6 hours of instruction

This course describes Test-driven development (TDD) as a software development process that follows a short, repeating cycle of turning requirements into test cases, then improving the code to pass the tests.

OBJECTIVES	PREREQUISITES
1. Understand the purpose and approach of test-driven development 2. Gain experience turning requirements into tests and on into working code	Unit Testing in Jest

SYLLABUS & TOPICS COVERED

- Introduction
    - What is TDD?
    - Benefits of Using TDD
- Core Concepts
    - How do we use TDD to create clean code?
    - Translating Business Requirements into Functional Requirements for Tests
- Types
    - Existing Types of Testing Where does TDD Fit?
    - Powerful Assert Statement
- Practical TDD
    - Translating requirements to testable assertions
    - How unit tests supplement docs
    - Generating docs with Pylint
    - Breaking down complex requirements and supplementing docs
    - Mocks in python with the unittest
    - mock module
    - Gaining Deeper Knowledge About Mock Objects

SOFTWARE REQUIREMENTS

Python, text editor

Build Tools: Task Runners

4 hours of instruction

A course that introduces learners to build tools for JavaScript, specifically task runners. The courses covers popular task runners and what works best for JavaScript.

OBJECTIVES	PREREQUISITES
1. Learners will understand what build tools are, specifically task runners 2. Understand when to use task runners and how to use NPM scripts	Package Managers

SYLLABUS & TOPICS COVERED

- Introduction To Task Runners
  - Learn about task runners and the role they play when building a react application
  - Understand what some popular task runner tools are, such as Grunt and Gulp, and the difference between them and module bundlers
  - Introduction to NPM scripts and the overall benefit of using NPM scripts versus other popular task runners
- Npm Scripts
  - Introduction to how to build a simple NPM script
  - How to use an NPM script as a build tool
  - Working with scripts such as testing, chaining and more
  - Using your scripts for deployments

SOFTWARE REQUIREMENTS

browser, VSCode or another IDE, terminal

Build Tools: Module Bundlers

4 hours of instruction

A course that introduces you to build tools for JavaScript, specifically module bundlers. Learn about some popular module bundlers and how to use a module bundler when building a React app.

OBJECTIVES	PREREQUISITES
1. Learners will understand what build tools are, specifically module bundlers 2. Introduce Webpack, Parcel and Rollup as module bundlers 3. Understand how to use a module bundler, walking through the steps of the process	Build Tools: Task Runners

SYLLABUS & TOPICS COVERED
• Introduction To Module Bundlers • Introduction to module bundlers and why they are beneficial • Webpack and Parcel • Setting up Parcel and using Parcel with middleware • Using Webpack • When to use Webpack instead of Parcel • Implement Webpack and walk through to all the steps and tools needed

SOFTWARE REQUIREMENTS
browser, VSCode or another IDE, terminal

Build Tools: Linters & Formatters

4 hours of instruction

A course that introduces learners to build tools for JavaScript, specifically linters and formatters. Learn how to use linters and formatters when building a React app.

OBJECTIVES	PREREQUISITES
1. Learners will understand the final layer of build tools, linters and formatters 2. Understand when and how to add linters and formatters to your react app 3. Learn about ESLint and Prettier as two popular libraries to use for linters and formatters	Build Tools: Module Bundlers

SYLLABUS & TOPICS COVERED

- Adding Linters And Formatters To Your App
  - What are linters and formatters and when to use them
  - Introduction to and installing ESLint and Prettier
  - Configure ESLint and Prettier
- Using ES Lint And Prettier
  - Run linter and formatters and walk through various scenarios
  - Format a project
  - Best practices and automating the process with husky

SOFTWARE REQUIREMENTS

browser, VSCode or another IDE, terminal

Intro to React: JSX & Components

6 hours of instruction

A course that establishes foundational knowledge of building interactive websites with React framework, focused on the basic concepts

OBJECTIVES	PREREQUISITES
1. Learners will explore how React works and how it fits into the frontend development framework 2. They will be able to build basic reusable components with JSX	Package Managers

SYLLABUS & TOPICS COVERED

- Create React App
  - An overview of React
  - How to use create-react-app
- JSX
  - Overview of class components
  - Build a class component
  - What is a component?
  - Component hierarchy
  - How to build a component
  - An overview of React
  - How to use create-react-app
  - Overview of functional components
  - Build a functional component
  - An overview of JSX
  - JSX vs
  - HTML
- Components
  - Overview of class components
  - Build a class component
  - What is a component?
  - Component hierarchy
  - How to build a component
  - An overview of React
  - How to use create-react-app
  - Overview of functional components
  - Build a functional component
  - An overview of JSX
  - JSX vs
  - HTML
- Class Components
  - Overview of class components

SOFTWARE REQUIREMENTS

browser, VSCode or another IDE, terminal

Intro to React: Props, State, & Conditional Rendering

4 hours of instruction

A course that establishes foundational knowledge of building interactive websites with React framework, focused on passing data between and conditionally rendering components

OBJECTIVES	PREREQUISITES
1. Learners will explore how to pass data between different React components as well as how to conditionally render components	Intro to React: JSX & Components

SYLLABUS & TOPICS COVERED

- Props Vs State
  - Overview of props vs
  - state
  - Pass data between components
  - Update data within a single component
- Conditional Rendering
  - Conditionally render a single component
  - Overview of props vs
  - state
  - Pass data between components
  - Update data within a single component

SOFTWARE REQUIREMENTS

browser, VSCode or another IDE, terminal

Intro to React: Component Lifecycle

2 hours of instruction

A course that establishes foundational knowledge of building interactive websites with React framework, focused on the component lifecycle

OBJECTIVES	PREREQUISITES
1. Learners will explore the various lifecycle methods of class components and see how they relate to functional components	Intro to React: Props, State, & Conditional Rendering

SYLLABUS & TOPICS COVERED

- Component Life Cycle
  - Overview of lifecycle methods in class components
  - Concepts of componentDidMount
  - Lifecycle methods in class components translated to functional components
  - useEffect

SOFTWARE REQUIREMENTS

browser, VSCode or another IDE, terminal

Intro to React: Lists, Keys & Hooks

4 hours of instruction

A course that establishes foundational knowledge of building interactive websites with React framework, focused on generating components from lists and using hooks

OBJECTIVES	PREREQUISITES
1. Learners will explore how to map over lists and pass keys 2. Learners will understand how to use basic hooks	Intro to React: Component Lifecycle

SYLLABUS & TOPICS COVERED
• Lists And Keys   • Review how to transform arrays/lists   • Pass in keys when rendering multiple components • Composition Vs Inheritance   • Overview of composition and Inheritance   • Benefits of using one over the other • Basic Hooks   • useState   • useEffect   • useContext   • Overview of other common hooks

SOFTWARE REQUIREMENTS
browser, VSCode or another IDE, terminal

Modern CSS: Styled Components

4 hours of instruction

A course that introduces you to styled components as a form of Modern CSS and helps you understand how to use them in React.

OBJECTIVES	PREREQUISITES
1. Understand what Modern CSS is, styled components and how to use them within their React application	Intro to React: Lists, Keys & Hooks

SYLLABUS & TOPICS COVERED

- Using Styled Components
  - Introduce styled components as a CSS-in-JS tool
  - Understand the basics of styled components and install
- Implementing Styled Components
  - Implement styled components within your react app
  - Adapt styles based on props
  - Using the CSS helper and StyleSheetManager
  - Techniques for easier debugging with styled components

SOFTWARE REQUIREMENTS

browser, VSCode or another IDE, terminal

Modern CSS: Modules

4 hours of instruction

A course that introduces you to modules as a form of Modern CSS and helps you understand how to use them in React.

OBJECTIVES	PREREQUISITES
1. Learners will understand how to use modules within Modern CSS to work within their React application 2. Best practices of working with Modern CSS Modules	Modern CSS: Styled Components

SYLLABUS & TOPICS COVERED

- Introduction To Modules
  - Introduce modules and compare them to styled components
  - Challenges of CSS at scale
  - Benefits of using modules and why we need them
  - Use CSS Modules to understand their impact
- Continue Using Modules
  - Special instances when using modules
  - Promoting reuse within CSS using Modules
  - Working with Pre-Processors and Modules

SOFTWARE REQUIREMENTS

browser, VSCode or another IDE, terminal

Modern CSS: Styled JSX

4 hours of instruction

Learn to use the styled-jsx library for creating encapsulated and scoped CSS to style your components.

OBJECTIVES	PREREQUISITES
1. Gain an understanding of the styled-jsx library 2. Learn to style components using the styled-jsx library	Modern CSS: Modules

SYLLABUS & TOPICS COVERED
• Styled JSX • Overview of styled-jsx library • How to style components • Understanding the scope of style definitions

SOFTWARE REQUIREMENTS
browser, VSCode or another IDE, terminal

Modern CSS: Emotion

4 hours of instruction

This course is an introduction to Emotion, a library designed for writing css styles with JavaScript. It provides powerful and predictable style composition in addition to a great developer experience with features such as source maps, labels, and testing utilities.

OBJECTIVES	PREREQUISITES
1. Learners will learn how to use Emotion an application with a consistent CSS composition 2. Learners will also explore the various applications of Emotion and primary methods for using Emotion	Modern CSS: Styled JSX

SYLLABUS & TOPICS COVERED

- Intro To Emotion
  - Need for Emotion library
  - Styling with string or object styles
  - Defaulting and extending variables
- Emotion With CSS
  - Applying styles directly with CSS prop
  - Primary methods for using Emotion
  - Deciding when to use Emotion for writing CSS

SOFTWARE REQUIREMENTS

browser, VSCode or another IDE, terminal

React Ecosystem: React Router

4 hours of instruction

A course that builds on the React Ecosystem. It walks through React Router, a fully-featured client and server-side routing library for React.

OBJECTIVES	PREREQUISITES
1. Learners will improve their React skills and be able to implement React Router in the React application 2. Understand server-side rendering (SSR) and static site generation (SSG) concepts	Intro to React: Lists, Keys & Hooks

SYLLABUS & TOPICS COVERED

- Routes
  - Intro to React Router
  - Browser Router implementation
  - Routes, Private Routes, Redirect, NavLink
- SSR Vs SSG
  - What are SSR and SSG?
  - Working with SSR, SSG

SOFTWARE REQUIREMENTS

TBD

React Ecosystem: API Calls

4 hours of instruction

A course that builds on the React Ecosystem. It walks through how to integrate APIs using various libraries into a React application

OBJECTIVES	PREREQUISITES
1. Learners will improve their React skills and be able to make API calls from a React application 2. Understand API request format and request methods available in React	React Ecosystem: React Router

SYLLABUS & TOPICS COVERED
• API Calls   • Fetch   • Axios   • Apollo

SOFTWARE REQUIREMENTS
TBD

React Ecosystem: Forms

4 hours of instruction

A course that builds on the foundations of React framework and expands learners' skills to more advanced concepts.

OBJECTIVES	PREREQUISITES
1. Learners will improve their React skills and learn about best practices when building forms in React	React Ecosystem: API Calls

SYLLABUS & TOPICS COVERED
- Forms   - Form elements   - Controlled components   - state/useState   - Successfully update/keep track of data using state or useState   - React Hook Form

SOFTWARE REQUIREMENTS
browser, VSCode or another IDE, terminal

React Ecosystem: State Management & Redux

4 hours of instruction

A course that builds on the React Ecosystem. It explains how state management works in React and goes over the Redux state management library

OBJECTIVES	PREREQUISITES
1. Learners will be introduced to Context and Redux 2. Learners will be able to implement state management in React using either Context API or Redux library	React Ecosystem: Forms

SYLLABUS & TOPICS COVERED

- Context Hooks
  - Context API
  - Creating, accessing and updating context
  - Intro to hooks, and common hooks
  - Hooks with context
- Redux
  - Context API
  - Creating, accessing and updating context
  - Intro to hooks, and common hooks
  - Hooks with context
  - Introduction to Redux
  - Redux store, actions, reducers
  - Redux thunk middleware
  - Redux with hooks

SOFTWARE REQUIREMENTS

browser, VSCode or another IDE, terminal

React Ecosystem: Styling

4 hours of instruction

A course that expands on your react knowledge to make your own styled components and leverage material UI library

OBJECTIVES	PREREQUISITES
1. Create your own styled components with your own encapsulated style 2. Leverage Material UI library to build faster and interactive React UI	React Ecosystem: State Management & Redux

SYLLABUS & TOPICS COVERED

- Styled Components
  - Install Styled Components
  - First Styled Component
- Material UI
  - Introduce and Install MUI
  - Customize MUI

SOFTWARE REQUIREMENTS

browser, text editor

React Testing Library

4 hours of instruction

A course that introduces to testing simple and complex React applications with React Testing Library.

OBJECTIVES	PREREQUISITES
1. Learners will be introduced to React Testing Library and how to implement testing in their applications 2. Learners will get to explore React Testing best practices: how to test behavior instead of implementation	React Ecosystem: Styling

SYLLABUS & TOPICS COVERED

- Introduction
  - Testing React with Jest and Testing Library
  - JEST and JEST DOM assertions
  - Watch mode and how tests work
  - ESLint for Testing Library and Jest-DOM
- Mocking Service
  - Introduction to Mock Service Worker and Handlers
  - Setting up the Mock Service Worker Server

SOFTWARE REQUIREMENTS

Text editor, React, vscode

Enzyme

4 hours of instruction

A course that explores Enzyme, which is a JavaScript utility for React applications. The course equips users to simulate runs and test React components' outputs.

OBJECTIVES	PREREQUISITES
1. Learn how to install Enzyme and use it for testing the different components of React 2. Write unit and integration tests using Enzyme	React Testing Library

SYLLABUS & TOPICS COVERED
• Testingwith Enzyme • Enzyme overview and basic usage • Installing Enzyme • Running Enzyme tests

SOFTWARE REQUIREMENTS
TBD

Advanced React, Part 1

4 hours of instruction

A course that builds on the foundations of React framework and expands learners' skills to more advanced concepts like Hooks, Context, Refs and RenderProps

OBJECTIVES	PREREQUISITES
1. Learners will improve their React skills and build on the foundational concepts 2. Understand the concepts of hooks, context, refs and render props and learn how to use them for driving interactive websites	Intro to React: Lists, Keys & Hooks

SYLLABUS & TOPICS COVERED

- Hooks
  - Common hooks
  - Writing your own hooks
- Context
  - Create the context
  - Provide the context
  - Consume the context
- Refs
  - Create refs
  - Using forwardRef
- Render Props
  - Render props for cross-cutting concerns
  - Using props other than render

SOFTWARE REQUIREMENTS

browser, VSCode or another IDE, terminal

Advanced React, Part 2

4 hours of instruction

A course that builds on the foundations of React framework and expands learners' skills to more advanced concepts like code splitting, high order components, portals and error boundaries.

OBJECTIVES	PREREQUISITES
1. Learners will improve their React skills and build on the foundational concepts 2. Understand the concepts of code splitting, high order components, portals and error boundaries and learn how to use them for driving interactive websites	Advanced React, Part 1

SYLLABUS & TOPICS COVERED

- Code Splitting
    - React
    - lazy
    - Route-based code splitting
- High Order Components
    - Use HOCs For Cross-Cutting Concerns
    - Pass Unrelated Props Through to the Wrapped Component
    - Maximizing Composability
- Portals
    - Portal Usage
    - Event Bubbling Through Portals
- Error Boundaries
    - Catch JavaScript errors anywhere in their child component tree, log those errors, and display a fallback UI
    - Event handlers

SOFTWARE REQUIREMENTS

browser, VSCode or another IDE, terminal

Progressive Web Applications

4 hours of instruction

A course that introduces the core elements of progressive web apps

OBJECTIVES	PREREQUISITES
1. Introduce the building blocks of progressive web apps 2. Incorporate key features of PWAs into their applications	Intro to React: Lists, Keys & Hooks

SYLLABUS & TOPICS COVERED

- Intro
  - Introduction and characteristics
  - Service Workers
  - Web App Manifest
  - Leverage device features

SOFTWARE REQUIREMENTS

browser, VSCode or another IDE, terminal

Back End

Back End

A pathway that takes students through the courses that span the topics in Backend engineering. The pathway starts with the very foundations of Python for backend applications, how to set up and maintain servers, databases, APIs and ends with the advanced topics on containerization and much more. This pathway equips the learners with a wide range of tools and frameworks to become backend engineers.

Courses Include

- **Backend: Python**
- **Intro to C/C++**
- **Intro to JavaScript Syntax & Basic Constructs**
- **Intro to JavaScript: the DOM**
- **Intro to JavaScript: Fetch Async Await**
- **Intro to JavaScript: Basic**
- **Backend: Express**
- **Databases: Relational**
- **Databases: NoSQL**
- **Databases: Advanced Relational**
- **Graph Databases**
- **RESTful APIs: Python**
- **And more...!**

DATA SOURCES & EXAMPLES USED

- Industry or domain-specific examples
- Customized data that maximizes the attendees' engagement and retention of skills and can be applied directly within your organization

PATH DURATION

- 142 Hours, approximately
- Available in flexible schedule format

MODALITY

- Proposed live-streaming (in-person available post-COVID-19)
- Instructor-led (TAs are optional)

RECOMMENDATIONS

- Up to 30 learners per cohort
- Review individual course prerequisites and software requirements

Backend: Python

4 hours of instruction

A course that teaches foundational knowledge of Python language specifically for backend.

OBJECTIVES	PREREQUISITES
1. Get familiar with different types of Python frameworks for backend engineering 2. Build a simple Flask app	Intermediate Python

SYLLABUS & TOPICS COVERED

- Backend Frameworks
  - Overview of backend framework
  - Most common backend frameworks
- Flask
  - Flask fundamentals
  - Implementation of a simple Flask app

SOFTWARE REQUIREMENTS

Anaconda, Python, Jupyter Notebooks

Intro to C/C++

4 hours of instruction

A course that builds foundational knowledge of the C and C++ programming languages.

OBJECTIVES	PREREQUISITES
1. The course enables learners with basic coding skills 2. The learner will be able to understand the fundamentals of C and C++ 3. Getting familiar with the syntax and writing basic programs in these languages	Operating System Basics, Terminal & Shell Scripts

SYLLABUS & TOPICS COVERED

- C Programming Basics
  - Overview of C language
  - Lexical elements and data types
  - Flow of control and simple functions
- C Plus Plus Programming Basics
  - Overview of C++ language
  - Basic syntax of C++
  - Difference between C and C++

SOFTWARE REQUIREMENTS

TBD

Intro to JavaScript Syntax & Basic Constructs

4 hours of instruction

A course that builds a foundational understanding of JavaScript basic constructs and syntax, and allows students to have hands-on experience building JavaScript driven websites.

OBJECTIVES	PREREQUISITES
1. Describe the core constructs of JavaScript and how it is used to create dynamic modern websites 2. Be able to create JavaScript-driven websites	Intro to CSS, Part 2

SYLLABUS & TOPICS COVERED

- Data Types And Operators
    - Data types
    - Operators
- Syntax Basics
    - Statements and expressions
    - Functions

SOFTWARE REQUIREMENTS

browser, text editor, VSCode

Intro to JavaScript: the DOM

4 hours of instruction

A course that builds on the basic constructs and syntax of JavaScript, and allows students to have hands-on experience of exploring the Document Object Model (DOM) structure.

OBJECTIVES	PREREQUISITES
1. Describe the core concepts of the DOM and its components 2. Be able to create JavaScript-driven websites	Intro to JavaScript Syntax & Basic Constructs

SYLLABUS & TOPICS COVERED

- DOM Basics
    - Intro to DOM and window objects
    - Getting and manipulating DOM elements

SOFTWARE REQUIREMENTS

browser, text editor, VSCode

Intro to JavaScript: Fetch Async Await

4 hours of instruction

A course that dives into the exploration of the frontend APIs, asynchronous calls and the concepts of modular JavaScript.

OBJECTIVES	PREREQUISITES
1. Explore the idea of asynchronous calls to frontend APIs 2. Use Ajax and Fetch API to implement a request cycle 3. Be able to build modular JavaScript code with the help of its functional and object-oriented tools	Intro to JavaScript: the DOM

SYLLABUS & TOPICS COVERED

- Ajax And Fetch API
  - What is an API?
  - RESTful conventions
  - Intro to HTTP request cycle
  - HTTP request structure
  - Intro to asynchronous functions
  - Using AJAX and Fetch

SOFTWARE REQUIREMENTS

browser, text editor, VSCode

Intro to JavaScript: Basic Concepts

6 hours of instruction

A course that finalizes the series of introductory JavaScript courses and introduces the students to the basic concepts in the JavaScript ecosystem.

OBJECTIVES	PREREQUISITES
1. Describe the basic concepts of JavaScript and best practices to create dynamic modern websites 2. Be able to create JavaScript-driven websites	Intro to JavaScript: Fetch Async Await

SYLLABUS & TOPICS COVERED

- Modular JavaScript And E S6
  - What are modules
  - Exporting and Importing
  - Functional vs Object-oriented Programming
  - The concept of "this"
- Hoisting And Event Bubbling
  - What is hoisting?
  - Event bubbling basics
  - Implementation of complex events that drive other events
- Scope And Prototypes
  - What is scope in JavaScript
  - The concept of a prototype object and how to use it

SOFTWARE REQUIREMENTS

browser, text editor, VSCode

Backend: Node.js

4 hours of instruction

A course that teaches foundational knowledge of Node.js to leverage your understanding of JavaScript to create new and powerful applications.

OBJECTIVES	PREREQUISITES
1. Get familiar with Nodejs fundamentals and server-side web development 2. Install and maintain a simple Nodejs application	Intro to JavaScript: Basic Concepts

SYLLABUS & TOPICS COVERED

- Node JS Fundamentals
  - Overview of node
  - js
  - Modules in node
- Implementation
  - Node
  - js installation
  - Events
  - Error Handling

SOFTWARE REQUIREMENTS

TBD

Backend: Express

4 hours of instruction

A course that teaches a learner to write server-side code using Express. It walks learners through features required for effective web and mobile application development. By the end of this course students will be able to create a simple web server application with Express.

OBJECTIVES	PREREQUISITES
1. Get familiar with Express fundamentals 2. Create and maintain a basic web server in Express	Backend: Node.js

SYLLABUS & TOPICS COVERED
• Express Basics   • Overview of Express   • Installing Express • Express Implementation   • Basic routing   • Create a web server

SOFTWARE REQUIREMENTS
TBD

Databases: Relational

4 hours of instruction

An introduction to relational databases and how relational databases can help shape your production-ready application.

OBJECTIVES	PREREQUISITES
1. Learn the fundamentals of databases and the relational model for database management 2. Connect and interact with a database with SQL queries 3. Learn to build your own databases and keep them updated	Backend: Python

SYLLABUS & TOPICS COVERED

- Intro To Databases And The Relational Model
  - Introduction to the relational model
  - Connecting to your database and autoloading tables
  - Understanding and querying your tables
  - Handling a ResultSet
- Creating And Manipulating Databases
  - Creating databases and tables
  - Constraints and data defaults
  - Updating data in your tables
  - Deleting data and tables from your database

SOFTWARE REQUIREMENTS

TBD

Databases: NoSQL

4 hours of instruction

An introduction to NoSQL databases, how they differ from relational and when and why they are beneficial to use.

OBJECTIVES	PREREQUISITES
1. Learn the fundamentals of databases and the non-relational model known as noSQL for database management 2. Understand the flavors of NoSQL and when to apply them to achieve your goals 3. Learn how to handle unstructured data	Backend: Python

SYLLABUS & TOPICS COVERED

- Intro To No SQL Mongo DB
  - Differences between NoSQL and relational databases
  - NoSQL , MongoDB and the document database
  - Working with MongoDB, accessing and manipulating data
- More On Concepts
  - NoSQL and the key value database, Redis
  - NoSQL and the Column family database, Cassandra
  - When to use NoSQL, business case examples
  - Scalability and Speed of NoSQL databases

SOFTWARE REQUIREMENTS

TBD

Databases: Advanced Relational

4 hours of instruction

A deeper dive into the many capabilities of a relational database, how to optimize usage and make sure that your are getting the most use out of your database so that you have a strong base for your applications.

OBJECTIVES	PREREQUISITES
1. Learn advanced features and capabilities of a relational database 2. Understand aspects of optimization within your database to increase speed and scalability	Databases: Relational

SYLLABUS & TOPICS COVERED
• Advanced Relational Database Concepts • Object relational mapping (ORMs) • ACID properties in SQL • Optimizing Databases • Transactions in SQL • Indexing • Normalization

SOFTWARE REQUIREMENTS
TBD

Graph Databases

4 hours of instruction

An introduction to graph databases and why they are so powerful as well as an overview of what you can use graph databases for.

OBJECTIVES	PREREQUISITES
1. Understand how graph databases are setup, what they are best for and when we benefit from their structure 2. Introduction to multiple players within the graph DB world 3. Examples of Cypher as a query language	Databases: Advanced Relational

SYLLABUS & TOPICS COVERED

- Introduction To Graph Databases
  - Introduction to graph databases, a form of NoSQL
  - Understanding the composition of a graph
  - When and why systemizing data in a graph has an advantage
- Working With Graphs
  - Introduce Neo4j and Cypher
  - Understand a simple query in Cypher
  - Demonstrate a case study where a graph database was implemented and how it benefited the situation and why

SOFTWARE REQUIREMENTS

TBD

RESTful APIs: Python

4 hours of instruction

An introduction to fundamental concepts on building RESTful APIs using Python.

OBJECTIVES	PREREQUISITES
1. Understand a variety of API design structures 2. Be able to implement a basic webserver using REST and make requests from a frontend client	Databases: Advanced Relational

SYLLABUS & TOPICS COVERED
• RESTful   • The HTTP request lifecycle   • REST design conventions   • Serving up RESTful data   • RESTful routing and frontend queries

SOFTWARE REQUIREMENTS
TBD

JSON APIs: Python

4 hours of instruction

An introduction to the concept of JSON API and how it can be implemented using Python.

OBJECTIVES	PREREQUISITES
1. Understand how JSON API relates to and differs from RESTful API conventions 2. Practice implementing and successfully querying a JSON API using Python	RESTful APIs: Python

SYLLABUS & TOPICS COVERED

- JSON
  - What is JSON API?
  - How queries are formed, interpreted and served
  - Implementing JSON API

SOFTWARE REQUIREMENTS

TBD

Basic GraphQL: Python

4 hours of instruction

An introduction to GraphQL, what it is good for and how to use it to query or change data.

OBJECTIVES	PREREQUISITES
1. Understand GraphQL and how it works 2. Successfully write queries and mutations and send them to a GraphQL server	JSON APIs: Python

SYLLABUS & TOPICS COVERED

- Introduction
  - What is GraphQL?
  - Queries & Mutations
  - Working with graph data

SOFTWARE REQUIREMENTS

TBD

Authentication Python

4 hours of instruction

An introduction to Authentication concepts and how it can be implemented using Python.

OBJECTIVES	PREREQUISITES
1. Basic understanding of Authentication mechanisms 2. Be able to implement a basic authentication system for APIs/platform	Basic GraphQL: Python

SYLLABUS & TOPICS COVERED
• Authentication   • Basic Authentication   • Token Authentication

SOFTWARE REQUIREMENTS
TBD

Testing in Python Overview

4 hours of instruction

An introduction to software testing and types of testing using Python.

OBJECTIVES	PREREQUISITES
1. Basic understanding of software testing 2. Be able to implement test suites as part of development	Authentication Python

SYLLABUS & TOPICS COVERED

- Testing
  - Intro to Testing
  - Testing framework
  - Types of testing
  - Structuring tests

SOFTWARE REQUIREMENTS

TBD

Unit Testing with Pytest

4 hours of instruction

A course that introduces unit testing and how it can be performed using Pytest in Python.

OBJECTIVES	PREREQUISITES
1. Basic understanding of unit testing 2. Explore features of pytest and be able to implement testing using pytest in python	Testing in Python Overview

SYLLABUS & TOPICS COVERED

- Testing
    - Intro to Pytest
    - Testing using Pytest
    - Features and concepts of Pytest
    - Best practices of unit testing

SOFTWARE REQUIREMENTS

TBD

Integration Testing in Python

4 hours of instruction

An introduction to integration testing and how to perform integration testing using Python.

OBJECTIVES	PREREQUISITES
1. Basic understanding of integration testing 2. Be able to implement test suites as part of development 3. Introduce and perform API, database and UI testing	Unit Testing with Pytest

SYLLABUS & TOPICS COVERED
• Testing • Intro to Integration testing • API testing • Database testing • UI testing using selenium

SOFTWARE REQUIREMENTS
TBD

RESTful APIs: Node.js

4 hours of instruction

An introduction to RESTful conventions for building and querying an API, using Node.js.

OBJECTIVES	PREREQUISITES
1. Understand the HTTP request lifecycle and the design conventions underpinning REST 2. Be able to implement a basic webserver using REST and make requests from a frontend client	Databases: Advanced Relational

SYLLABUS & TOPICS COVERED

- RESTful
  - The HTTP request lifecycle
  - REST design conventions
  - Serving up RESTful data
  - RESTful routing and frontend queries

SOFTWARE REQUIREMENTS

TBD

JSON APIs: Node.js

4 hours of instruction

An introduction to JSON API, using Node.js.

OBJECTIVES	PREREQUISITES
1. Understand how JSON API relates to and differs from RESTful API conventions 2. Practice implementing and successfully querying a JSON API using Nodejs	RESTful APIs: Node.js

SYLLABUS & TOPICS COVERED
• JSON   • What is JSON API?   • How queries are formed, interpreted and served   • Implementing JSON API

SOFTWARE REQUIREMENTS
TBD

Basic GraphQL: Node.js

4 hours of instruction

An introduction to GraphQL, what it is good for and how to use it to query or change data.

OBJECTIVES	PREREQUISITES
1. Understand GraphQL and how it works 2. Successfully write queries and mutations and send them to a GraphQL server	JSON APIs: Node.js

SYLLABUS & TOPICS COVERED

- Introduction
  - What is GraphQL?
  - Queries & Mutations
  - Working with graph data

SOFTWARE REQUIREMENTS

TBD

Authentication Node.js

4 hours of instruction

An introduction to Authentication concepts using Node.js.

OBJECTIVES	PREREQUISITES
1. Basic understanding of Authentication mechanisms 2. Be able to implement a basic authentication system for APIs/platform	JSON APIs: Node.js

SYLLABUS & TOPICS COVERED

- Authentication
  - Basic Authentication
  - Token Authentication

SOFTWARE REQUIREMENTS

TBD

Testing Node.js

4 hours of instruction

An introduction to software testing and types of testing using Node.js.

OBJECTIVES	PREREQUISITES
1. Basic understanding of software testing 2. Be able to implement test suites as part of development	Authentication Node.js

SYLLABUS & TOPICS COVERED
• Testing   • Integration Testing   • Unit Testing   • Functional Testing

SOFTWARE REQUIREMENTS
TBD

Advanced GraphQL: Python

4 hours of instruction

An introduction to GraphQL advanced concepts and tools using Python. Learn how to use Apollo graph platform to manage GraphQL.

OBJECTIVES	PREREQUISITES
1. Understand GraphQL advanced concepts 2. Successfully setting up a GraphQL Apollo server, custom resolvers, and implementing GraphQL advanced features	Authentication Python

SYLLABUS & TOPICS COVERED

- Apollo
  - Apollo
- Advanced
  - GraphQL features
  - Apollo

SOFTWARE REQUIREMENTS

TBD

Advanced GraphQL: Node.js

4 hours of instruction

An introduction to GraphQL advanced concepts and tools using Node.js. Learn how to use Apollo graph platform to manage GraphQL.

OBJECTIVES	PREREQUISITES
1. Understand GraphQL advanced concepts 2. Successfully set up a GraphQL Apollo server, custom resolvers, and implementing GraphQL advanced features	Authentication Node.js

SYLLABUS & TOPICS COVERED

- Apollo
  - Apollo
- Advanced
  - GraphQL features
  - Apollo

SOFTWARE REQUIREMENTS

TBD

Web Security

4 hours of instruction

A course that goes over the basic concepts in web security, and gives a brief overview of the common and emerging web vulnerabilities.

OBJECTIVES	PREREQUISITES
1. Understand fundamentals of web security 2. Get acquainted with common and emerging vulnerabilities, and tools to deal with these vulnerabilities	Testing Node.js

SYLLABUS & TOPICS COVERED

- Intro To Web Security
    - HTTPS
    - Content security policy
    - CORS
- Common Vulnerabilities
    - OWASP Security Risks
    - Password breaches
    - Authentication
    - Preventing and resolving vulnerabilities

SOFTWARE REQUIREMENTS

TBD

Caching

4 hours of instruction

A course that goes over different types of caching and their use cases on client and server side.

OBJECTIVES	PREREQUISITES
1. Understand server side and client side caching with respect to web applications 2. Learn caching best practices and understand the validity of the data being cached	Web Security

SYLLABUS & TOPICS COVERED

- Server Side
    - Database caching
    - Content Delivery Network (CDN) caching
    - Domain Name System (DNS) Caching
- Client Side
    - HTTP cache headers
    - Browser caching

SOFTWARE REQUIREMENTS

TBD

Design & Development

4 hours of instruction

A course that goes over the principles and architectural patterns of software design and development

OBJECTIVES	PREREQUISITES
1. Understand the principles and architectural patterns underpinning software design and development	Caching

SYLLABUS & TOPICS COVERED

- Principles
  - DRY
  - SOLID
  - KISS
  - YAGNI
- Architectural Patterns
  - Monolithic Apps
  - Microservices
  - SOA
  - Serverless

SOFTWARE REQUIREMENTS

TBD

Search Engines

4 hours of instruction

A course that goes over the basic concepts and terminologies of elasticsearch and how to write queries for elasticsearch

OBJECTIVES	PREREQUISITES
1. Understand the core concepts and terminologies of elasticsearch 2. Learn how to write complex search queries in elasticsearch	Design & Development

SYLLABUS & TOPICS COVERED

- Elastic Search
  - Indexing and documents
  - Clusters and nodes
  - Shards and replicas
  - Complex search queries

SOFTWARE REQUIREMENTS

TBD

Message Brokers

4 hours of instruction

A course that goes over the basic concepts and terminologies of message brokers

OBJECTIVES	PREREQUISITES
1. Understand the need for message brokers and how they work 2. Dive deep into RabbitMQ message broker	Search Engines

SYLLABUS & TOPICS COVERED

- Overview
  - Need for message brokers
  - Message broker architecture
- Rabbit MQ
  - AMQP protocol and message flow
  - Environment variables and config file
  - Messaging patterns
  - Publish / Subscribe

SOFTWARE REQUIREMENTS

TBD

Containerizations vs Virtualizations

4 hours of instruction

An introduction to the concepts of Containerization and Virtualizations, and understanding the differences between them.

OBJECTIVES	PREREQUISITES
1. General understanding of containerization and virtualization concepts 2. How to set up and use Docker for containerization 3. Purposes and advantages of containerization and virtualization	Message Brokers

SYLLABUS & TOPICS COVERED

- Intro To Containerizations
  - What is Containerization?
  - Purposes of Containerization
  - Containerization with Docker
  - Advantages of Containerization
- Intro To Virtualizations
  - What is Virtualization?
  - Purposes of Virtualization
  - Advantages of Virtualization

SOFTWARE REQUIREMENTS

TBD

WebSockets

4 hours of instruction

An introduction to the concept of Websockets and how its used as a communication protocol in client-server communication

OBJECTIVES	PREREQUISITES
1. General understanding of the need for websockets are how they are used for sending and receiving messages over a TCP protocol	Containerizations vs Virtualizations

SYLLABUS & TOPICS COVERED

- Intro To Web Sockets
  - Overview of Websockets
  - Why are Websockets needed?
  - Working of Websockets in a client-server communication
  - Websockets Vs HTTP

SOFTWARE REQUIREMENTS

TBD

Web Servers

4 hours of instruction

An introduction to the concept of Web servers and how they are used to used to satisfy World Wide Web client requests

OBJECTIVES	PREREQUISITES
1. Learn what web servers are and why they are important 2. Gain a general understanding of how a web server works 3. Introduce Apache and Nginx and configure a virtual server using them	WebSockets

SYLLABUS & TOPICS COVERED

- Intro To Web Servers
  - Overview of Web Servers
  - Features of Web Servers
  - How do Web Servers work?
  - Dynamic Vs Static Web Servers
- Apache Server
  - Introduction to Apache Server
  - Configuring a Virtual Server in Apache
- Nginx Server
  - Introduction to Nginx
  - Configuring a Virtual Server in Nginx

SOFTWARE REQUIREMENTS

TBD

Building for Scale

4 hours of instruction

An introduction to the concept of web application scalability and the approaches used to scale an application

OBJECTIVES	PREREQUISITES
1. Understanding the need for scaling a web application 2. Approaches used in web application scalability	Web Servers

SYLLABUS & TOPICS COVERED

- Web Application Scalability
  - Overview of web application scalability
  - Need for scaling a web application
  - Horizontal Scaling
  - Vertical Scaling

SOFTWARE REQUIREMENTS

TBD

Artificial Intelligence

Artificial Intelligence

A pathway that builds on concepts of Data Science pathway and advances the students into the field of Neural Networks and the types of problems these special cutting edge models solves. By the end of this pathway learners will upgrade their Data Science skills to become proficient in Deep Learning.

Courses Include

- **Intro to Neural Networks**
- **Neural Networks & Deep Learning**
- **Convolutional Neural Networks (CNN) for Image Recognition**
- **Deep Learning for Text Analysis**
- **Advanced Deep Learning for Text Analysis**
- **Advanced CNN**
- **Autoencoders**
- **Object Detection**
- **Generative Adversarial Networks**
- **Reinforcement Learning**

DATA SOURCES & EXAMPLES USED

- Industry or domain-specific examples
- Customized data that maximizes the attendees' engagement and retention of skills and can be applied directly within your organization

PATH DURATION

- 42 Hours, approximately
- Available in flexible schedule format

MODALITY

- Proposed live-streaming (in-person available post-COVID-19)
- Instructor-led (TAs are optional)

RECOMMENDATIONS

- Up to 30 learners per cohort
- Review individual course prerequisites and software requirements

Intro to Neural Networks

6 hours of instruction

This course gives students the first preview of the world of Neural Networks and how they work. These state-of-the-art methods build powerful predictive systems and find latent patterns in large amounts of data. By the end of this course, students will learn the foundations of this complex topic and acquire practical skills to build neural networks in order to solve real-world problems.

OBJECTIVES	PREREQUISITES
1. Define core applications and use cases for deep learning 2. Build foundational neural network models	Optimizing Ensemble Methods

SYLLABUS & TOPICS COVERED

- Basics
  - Introduction to neural networks
  - Neural networks use cases
- Intro To TensorFlow
  - TensorFlow and Keras as two leading frameworks for implementing neural networks
  - Overview of TensorFlow / Keras building blocks
- Feed Forward Networks
  - Forward and back propagation
  - Single layer perceptron
  - Implementation of models in TF

SOFTWARE REQUIREMENTS

Python, Anaconda

Neural Networks & Deep Learning

4 hours of instruction

This course builds on the foundations of neural networks and takes through a series of practical examples of how to measure the performance of a neural network algorithm, tune it and accelerate it.

OBJECTIVES	PREREQUISITES
1. Assess the performance of neural networks, choose the right metrics for the given use case 2. Tune neural a network model and find a way to accelerate it	Intro to Neural Networks

SYLLABUS & TOPICS COVERED

- Model Performance And Fit
  - Choosing the right model performance metrics
  - Accuracy, precision, recall, F1 against loss
- Tuning And Accelerating
  - Tuning and acceleration options
  - Using Keras Tuner for model tuning

SOFTWARE REQUIREMENTS

Python, Anaconda

Convolutional Neural Networks (CNN) for Image Recognition

4 hours of instruction

This course starts of a series of topics on neural networks designed to solve a particular family of tasks. In this course students will be able to get an overview of how to work with image data and build Convolutional Neural Networks (CNNs) - the industry standard for tackling image-based data.

OBJECTIVES	PREREQUISITES
1. Define use cases for image analysis 2. Define the concept of a CNN and implement the CNN on the MNIST dataset	Neural Networks & Deep Learning

SYLLABUS & TOPICS COVERED

- Image Analysis With CN Ns
  - Overview of CNNs and their use cases
  - Model inputs and outputs for image analysis type problems
- CNN Architecture
  - CNN architecture
  - Training process of a CNN
- CNN Implementation
  - Image data processing for CNNs
  - Building and implementing simple CNNs
  - Measuring and assessing performance

SOFTWARE REQUIREMENTS

Python, Anaconda, Jupyter Notebook or JupyterLab

Deep Learning for Text Analysis

4 hours of instruction

This course continues on tackling topics in deep learning that address specific problem types. In this course students will be getting to know RNNs and LSTMs - types of neural networks that are often used for solving problems in text analysis.

OBJECTIVES	PREREQUISITES
1. Build basic predictive systems for text using deep learning 2. Build natural language understanding and generation models	Neural Networks & Deep Learning

SYLLABUS & TOPICS COVERED

- Intro To RNN
  - Introduction to RNN
  - Discuss best practices for RNN models
- LSTM
  - LSTMs for text data and their use cases
  - LSTM implementation in TensorFlow

SOFTWARE REQUIREMENTS

Python, TensorFlow, Anaconda, Jupyter Notebook or JupyterLab

Advanced Deep Learning for Text Analysis

4 hours of instruction

This course continues on tackling topics in deep learning for text analysis. In this course students will be getting to know how to use and implement Gated Recurrent Units (GRUs) and model and predict longer sequences of text by leveraging Seq2Seq models.

OBJECTIVES	PREREQUISITES
1. Build advanced deep learning models to predict sequences of text 2. Implement GRU using TensorFlow and predict on test data 3. Explore use cases of Seq2Seq models	Deep Learning for Text Analysis

SYLLABUS & TOPICS COVERED

- GRU
  - Gated recurrent unit (GRU) in TensorFlow
  - Implementing GRU in TensorFlow
  - Implement the concept of stateful GRU
- Seq2Seq
  - Seq2seq theory
  - Implement seq2seq in TensorFlow

SOFTWARE REQUIREMENTS

Python, TensorFlow, Anaconda, Jupyter Notebook or JupyterLab

Advanced CNN

4 hours of instruction

This course build on the subject of Convolutional Neural Networks and dives into the complex pre-trained state-of-the-art CNN architectures. It also gives students a preview of what transfer learning is and why it is such a powerful concept in Deep Learning. By the end of this course students will be able to have implemented and explored pre-trained models such as ResNet, VGG16, and Inception3.

OBJECTIVES	PREREQUISITES
1. Define the need for advanced CNNs 2. Describe optimization using regularization and apply to CNN 3. Implement a VGG16 on the same dataset to compare performance and explore the concept of transfer learning	Convolutional Neural Networks (CNN) for Image Recognition

SYLLABUS & TOPICS COVERED

- Advanced CNN Architectures
  - Use cases for advanced CNN architectures
  - Difference between CNNs and advanced CNNs
- Baseline CNN
  - Implementation of a baseline CNN
  - Measuring performance of a baseline CNN
- Advanced CNN Models
  - Introduction to popular pre-trained advanced CNN models
  - VGG16, Inception, and ResNet architecture and implementation details
  - The concept of transfer learning and it use cases

SOFTWARE REQUIREMENTS

Python, Anaconda, Jupyter Notebook or JupyterLab

Autoencoders

4 hours of instruction

This course takes students through a journey into the world od autoencoders - a set of powerful deep learning models that have a special place in the world of image analysis. By the end of this course students will be able to navigate through the application space of autoencoders and implement autoencoders to perform tasks such as image denoising and more.

OBJECTIVES	PREREQUISITES
1. Define use cases for autoencoders and what tasks they can do in the image analysis space 2. Implement convolutional and denoising autoencoders	Convolutional Neural Networks (CNN) for Image Recognition

SYLLABUS & TOPICS COVERED

- Autoencoder Overview
  - Autoencoders and their use cases in image processing
  - Architecture of a simple autoencoder
- Autoencoder Implementation
  - Implementation of convolutional
  - Implementation of denoising autoencoder

SOFTWARE REQUIREMENTS

Python, Anaconda, Jupyter Notebook or JupyterLab

Object Detection

4 hours of instruction

This course introduces students to a special case of image analysis that addresses the problem of object detection in images. By the end of this course students will be able to create a YOLO - a deep learning model used specifically for such tasks.

OBJECTIVES	PREREQUISITES
1. Define use cases for object detection in the space of image analysis 2. Analyze images for the purpose of automatic object classification	Convolutional Neural Networks (CNN) for Image Recognition

SYLLABUS & TOPICS COVERED

- Object Detection Overview
  - Summarize use cases and the basis of object detection
  - Define YOLO and its core features
- Object Detection Implementation
  - Preprocessing data for object detection
  - Loading pre-trained weights and instantiate a YOLO model
  - Implementation of YOLO model
  - Prediction and evaluation of the results

SOFTWARE REQUIREMENTS

Python, Anaconda, Jupyter Notebook or JupyterLab

Generative Adversarial Networks

4 hours of instruction

This course covers the area in image analysis and computer vision that deals with generative models. By the end of this course students will be able to implement a GAN model to generate new images from a set of training examples.

OBJECTIVES	PREREQUISITES
1. Summarize the basis of generative adversarial networks and its applications 2. Define a combined model and generate images using the created GAN model	Advanced CNN

SYLLABUS & TOPICS COVERED

- GAN Overview
  - Generative modeling and its use cases in computer vision and image processing
  - Summary the basis of generative adversarial networks
- GAN Implementation
  - Definition of discriminator and generator models
  - Implementation of the training process for GANs
  - Generation image samples for modeling

SOFTWARE REQUIREMENTS

Python, Anaconda, Jupyter Notebook or JupyterLab

Reinforcement Learning

4 hours of instruction

This course covers the specialized branch of machine learning and deep learning called reinforcement learning (RL). By the end of this course students will be able to define RL use cases and real world scenarios where RL models are used, they will be able to create a simple RL model and evaluate its performance.

OBJECTIVES	PREREQUISITES
1. Discuss the theory behind reinforcement learning and use cases 2. Apply reinforcement learning theory to deep learning models 3. Implement a reinforcement learning model using TensorFlow	Advanced CNN

SYLLABUS & TOPICS COVERED

- Reinforcement Learning Overview
  - Reinforcement learning (RL) use cases
  - Theoretical concepts behind RL
- Reinforcement Learning Implementation
  - Set up of the RL testing environment
  - Implementation of an RL model using TensorFlow
  - Evaluation of the RL model performance and next steps

SOFTWARE REQUIREMENTS

Python, Anaconda, Jupyter Notebook or JupyterLab

Big Data

Big Data

A pathway that builds on the concepts of Data Science pathway and propels students into the realm of data processing and analysis at scale. By the end of this pathway learners will acquire skills necessary to tackle the most common data engineering tasks and be able to handle large datasets that don't fit in memory.

Courses Include

- **Foundations of Big Data**
- **Big Data Orchestration & Workflow Management**
- **Distributed Data Storage (Hadoop)**
- **Building Scalable Models in PySpark**
- **Intro to Scala Collections**
- **Spark Data Structures & Parallelism**
- **Spark Partitioning & Optimization**

DATA SOURCES & EXAMPLES USED

- Industry or domain-specific examples
- Customized data that maximizes the attendees' engagement and retention of skills and can be applied directly within your organization

PATH DURATION

- 32 Hours, approximately
- Available in flexible schedule format

MODALITY

- Proposed live-streaming (in-person available post-COVID-19)
- Instructor-led (TAs are optional)

RECOMMENDATIONS

- Up to 30 learners per cohort
- Review individual course prerequisites and software requirements

Foundations of Big Data

4 hours of instruction

A theoretical course covering topics on how to handle data at scale and the different tools needed for distributed data storage, analysis, and management. Learners will be able to dive into the vast world of data and computing at scale and get a comprehensive overview of distributed computing.

OBJECTIVES	PREREQUISITES
1. Understand the big data ecosystem and explore tools and methodologies needed for distributed data storage and big data analysis	Optimizing Ensemble Methods

SYLLABUS & TOPICS COVERED
• Intro To Big Data   • Data at scale   • Major sources of big data and industries that deal with it on daily basis • Distributed Data Storage And Analysis   • Need for distributed data storage   • Scalability, fault tolerance, and reliability   • Tools for distributed data storage

SOFTWARE REQUIREMENTS
TBD

Big Data Orchestration & Workflow Management

4 hours of instruction

A theoretical course covering topics on how to handle data at scale and the different tools needed for orchestrating big data systems and manage the workflow. Learners will be able to dive into the vast world of data and computing at scale and get a comprehensive overview of the distributed resource management ecosystem.

OBJECTIVES	PREREQUISITES
1. Understand the big data ecosystem and what role orchestration and resource management play in it 2. Students will explore tools and methodologies needed for distributed workflow management and orchestration	Foundations of Big Data

SYLLABUS & TOPICS COVERED

- Resource Management
  - Need for resource management at scale
  - Resource management for distributed systems
  - Tools for resource management at scale
- Workflow Orchestration Tools
  - Workflow orchestration for distributed systems
  - Workflow orchestration tools at scale

SOFTWARE REQUIREMENTS

TBD

Distributed Data Storage (Hadoop)

6 hours of instruction

A course that covers theory and implementation on a specific cloud platform covering topics on distributed data storage systems. Learners will be able to dive into the nature of storing and processing data at scale using tools like Hadoop on a selected cloud platform. This course will allow students to get a great foundation for creating and managing distributed data storage resources.

OBJECTIVES	PREREQUISITES
1. Understand the need for distributed data storage 2. Overview of components of Hadoop architecture 3. Learn to deploy a Hadoop application on a cloud service 4. Explore cross-functional tools used in conjunction with Hadoop	Foundations of Big Data

SYLLABUS & TOPICS COVERED

- Apache Hadoop
  - Overview of Hadoop's main layers
  - Introduce HDFS(Hadoop Distributed File System)
  - Discuss YARN and MapReduce
- Intro To HDFS
  - Overview and architecture of HDFS
  - Deploy a Hadoop application on a cluster
- Other Tools
  - Hadoop alternatives and other cross-functional tools used with Hadoop

SOFTWARE REQUIREMENTS

AWS, JupyterHub, Hadoop, Java, Python

Building Scalable Models in PySpark

4 hours of instruction

Learn how to optimize your code and to speed up current data processing using PySpark. In this course, students will work through best practices of how and when to use PySpark. They will explore what they can do with PySpark and how to use distributed computing within PySpark.

OBJECTIVES	PREREQUISITES
1. Define use of Spark and PySpark and their role in Big Data analysis 2. Query and analyze data in PySpark 3. Build scalable machine learning models with PySpark	Foundations of Big Data

SYLLABUS & TOPICS COVERED

- Basics
  - Working with data in PySpark
  - RDDs vs
  - DataFrames vs
  - Datasets
  - Optimized queries with Datasets
- SparkSQL
  - SparkSQL and its use cases
  - DataFrame API and operations on DataFrames
- Logistic Regression
  - Logistic regression use cases and theory behind it
  - Logistic regression implementation in Spark
  - Parallel processing in Spark

SOFTWARE REQUIREMENTS

JupyterLab

Intro to Scala Collections

4 hours of instruction

A 4-hour course for intermediate-level data scientists / engineers that covers the key elements and different types of Scala collections.

OBJECTIVES	PREREQUISITES
1. Explain the use cases of scala collections, their differences, and implementation details 2. Outline key variances in mutable vs immutable collections and how to differentiate between linear and indexed collections	Foundations of Big Data

SYLLABUS & TOPICS COVERED
• Linear Vs Indexed Collections • The concept of collections in Scala • Differences between linear vs indexed collections • Implementation of most common linear and indexed collections • Mutable Vs Immutable Collections • Differences between mutable and immutable collections • Implementation of most common mutable and immutable collections • Arrays In Scala • Array as a special collection type in Scala • Implementation of Array collection

SOFTWARE REQUIREMENTS
Apache Spark, JupyterLab

Spark Data Structures & Parallelism

4 hours of instruction

A 4-hour course for intermediate-level data scientists / engineers that covers Spark architecture and fundamentals including RDDs, DataFrames, Datasets.

OBJECTIVES	PREREQUISITES
1. Get familiar with key use cases for Spark and its core features 2. Be able to manipulate data in Spark using RDDs, DataFrames, and Datasets 3. Know how parallel processing works and be able to explore Spark UI to monitor Spark jobs	Intro to Scala Collections

SYLLABUS & TOPICS COVERED

- Spark Basics
  - Spark use cases, architecture, and features
  - Spark architecture and components
- Spark Data Structures Overview
  - RDDs as the core data structure in Spark
  - RDD features
  - Working with RDDs
- Spark Data Frames And Datasets
  - DataFrame features and what makes them different from RDDs
  - Working with DataFrames
- Spark UI
  - Core concepts of parallel processing in Spark
  - Using Spark UI to monitor Spark jobs

SOFTWARE REQUIREMENTS

Apache Spark, JupyterLab

Spark Partitioning & Optimization

6 hours of instruction

A 6-hour course for intermediate-level data scientists / engineers that covers spark partitions, benchmarking, performance optimization and monitoring.

OBJECTIVES	PREREQUISITES
1. Utilize Spark's intrinsic parallelism and its features to optimize performance 2. Leverage SparkUI to monitor Spark jobs	Spark Data Structures & Parallelism

SYLLABUS & TOPICS COVERED

- Optimization Methods
    - Spark partitions
    - Benchmarking performance
    - Caching and persistence
- Implementing Optimization
    - Role of shared variables in Spark
    - Partitioning data in memory vs partitioning on disk
    - Optimizing performance and comparing results

SOFTWARE REQUIREMENTS

Apache Spark, JupyterLab

DevOps

DevOps

A pathway that builds on the core concepts of backend engineering and treats infrastructure and operating system as its playground. By the end of this pathway learners will be able to streamline, connect, deploy, and scale operations within applications and across projects.

Courses Include

- **Understanding Different OS Concepts Python**
- **Introduction to Managing Servers**
- **Web Server & Other Topics**
- **Infrastructure as Code**
- **CI/CD**
- **Testing Automation**

DATA SOURCES & EXAMPLES USED

- Industry or domain-specific examples
- Customized data that maximizes the attendees' engagement and retention of skills and can be applied directly within your organization

PATH DURATION

- 24 Hours, approximately
- Available in flexible schedule format

MODALITY

- Proposed live-streaming (in-person available post-COVID-19)
- Instructor-led (TAs are optional)

RECOMMENDATIONS

- Up to 30 learners per cohort
- Review individual course prerequisites and software requirements

Understanding Different OS Concepts Python

4 hours of instruction

A course that builds foundational knowledge of what an operating system in Python. It walks through the different core concepts of OS and its inner workings.

OBJECTIVES	PREREQUISITES
1. - General understanding of basic components and types of Operating Systems 2. - Be able to create or execute processes through scripting (Python)	Understanding Different OS Concepts

SYLLABUS & TOPICS COVERED

- OS Concepts With Python
  - Process Management
  - Threads and Concurrency
  - Sockets
  - POSIX Basics

SOFTWARE REQUIREMENTS

TBD

Introduction to Managing Servers

4 hours of instruction

A course that builds foundational knowledge of server management and monitoring

OBJECTIVES	PREREQUISITES
1. General understanding of Server Administration, Management, and Monitoring 2. Manage and serve applications and other processes 3. Familiarity with terminal basics such as bash scripting, text manipulation tools, and compiling applications	Understanding Different OS Concepts Python

SYLLABUS & TOPICS COVERED

- Operating Server
    - Linux/Unix
    - Debian or other Linux distros
    - Windows
- Terminal Basics
    - Process Monitoring
    - Network
    - Bash Scripting

SOFTWARE REQUIREMENTS

TBD

Web Server & Other Topics

4 hours of instruction

A course that builds foundational knowledge of web servers and other topics related to servers

OBJECTIVES	PREREQUISITES
1. General understanding of Web Servers 2. Equip learners to configure a Web Server to serve content	Introduction to Network Protocols

SYLLABUS & TOPICS COVERED

- Web Server
  - Caching Server
  - Reverse Proxy
  - Forward Proxy
  - Whitelisting
  - Nginx
  - Apache
  - Firewall
  - Load Balancer
- Other Topics
  - Caching Server
  - Reverse Proxy
  - Forward Proxy
  - Whitelisting
  - Nginx
  - Apache
  - Firewall
  - Load Balancer

SOFTWARE REQUIREMENTS

TBD

Infrastructure as Code

4 hours of instruction

A course that introduces infrastructure as code (IaC) explains how to keep up with the industry standards.

OBJECTIVES	PREREQUISITES
1. Equip learners to define system Infrastructure through industry-standard IaC platforms 2. Enable learners to create and manage application Containers	Web Server & Other Topics

SYLLABUS & TOPICS COVERED

- Containers
  - Docker
- Container Orchestration
- Infrastructure Provisioning
  - GCP Cloud Deployment Manager
  - AWS Cloudformation
- Configuration Management
  - Ansible

SOFTWARE REQUIREMENTS

TBD

CI/CD

4 hours of instruction

A course that teaches continuous integration and continuous deployment to modernize and improve the software development lifecycle.

OBJECTIVES	PREREQUISITES
1. Understanding of a variety of industry-standard CI/CD platforms 2. Enable learners to configure a Continuous Integration / Continuous Deployment Pipeline	Infrastructure as Code

SYLLABUS & TOPICS COVERED
• Introduction To CI/CD   • Overview of CI/CD   • CircleCi/GithubActions • Monitoring Tools   • Infrastructure Monitoring   • Application Monitoring   • Logs Managing

SOFTWARE REQUIREMENTS
TBD

Testing Automation

4 hours of instruction

An introduction to CI/CD and how to perform testing using automated CI/CD pipelines using CircleCI.

OBJECTIVES	PREREQUISITES
1. Basic understanding of CI/CD pipelines 2. Be able to implement test suites using pipelines	CI/CD

SYLLABUS & TOPICS COVERED

- Testing Pipelines
  - Introduce concepts of CI/CD
  - Setup pipelines using Docker and CircleCI
- Circle CI For Testing
  - Introduction to CircleCI as a tool used in testing
  - Set up and validation of pipelines in CircleCI
- Selenium For Testing
  - Introduce Selenium for testing automation
  - Perform Selenium testing using CircleCI

SOFTWARE REQUIREMENTS

TBD

DATA SOCIETY

Machine Learning Operations (MLOps)

Machine Learning Operations (MLOps)

A pathway that builds on the concepts of Data Science, Artificial Intelligence, and DevOps to make up a unique direction for learners that would become proficient ML engineers by the end of the pathway. MLOps pathway is for those who would like to build production-grade AI-driven systems.

Courses Include

- **Intro to MLOps Theory**
- **Automating Data Pipelines & Workflow**
- **ML Introduction & Data Preparation**
- **Model Development & Testing**
- **Model Packaging, Deployment & Monitoring**
- **MLOPs CI/CD Theory**
- **Creating CI/CD Pipeline for Machine Learning (ML)**
- **Testing ML Pipelines**

DATA SOURCES & EXAMPLES USED

- Industry or domain-specific examples
- Customized data that maximizes the attendees' engagement and retention of skills and can be applied directly within your organization

PATH DURATION

- 42 Hours, approximately
- Available in flexible schedule format

MODALITY

- Proposed live-streaming (in-person available post-COVID-19)
- Instructor-led (TAs are optional)

RECOMMENDATIONS

- Up to 30 learners per cohort
- Review individual course prerequisites and software requirements

Intro to MLOps Theory

4 hours of instruction

This theoretical course gives a comprehensive overview of the topics that make up the emerging trend of MLOps. It is suitable for those who are interested in obtaining a big-picture blueprint of the MLOps space and different types of version control within the ML-driven systems.

OBJECTIVES	PREREQUISITES
1. Describe MLOps and its uses 2. Recognize gaps in the machine learning workflow and identify tools to fix them	Optimizing Ensemble Methods

SYLLABUS & TOPICS COVERED

- Intro To ML Ops
    - MLOps definition
    - Importance of MLOps
    - MLOps use cases
    - Importance of version control in machine learning
    - Version control throughout the machine learning workflow
    - Version control tools
- Version Control In ML
    - Importance of version control in machine learning
    - Version control throughout the machine learning workflow
    - Version control tools

SOFTWARE REQUIREMENTS

TBD

Automating Data Pipelines & Workflow

8 hours of instruction

Learn how to automate your entire pipeline using an automation tool. In this course students will learn how to programatically author, schedule and monitor their workflows. Students will also learn how to create an environment to containerize, replicate and deploy a pipeline.

OBJECTIVES	PREREQUISITES
1. Navigate through the growing landscape of automation tools for DataOps and MLOps 2. Acquire foundational knowledge of Airflow components 3. Set up a simple Airflow pipeline 4. Make distinctions between various Airflow set up options on cloud platforms 5. Create and test a data pipeline using Airflow with Docker on a compute instance	Intro to MLOps Theory

SYLLABUS & TOPICS COVERED

- Data Ops Basics
  - What does automation in data science mean? What parts can and should be automated?
  - Comparison of open source frameworks, their pros and cons
  - Apache Airflow origins and use cases
- Airflow Implementation
  - Airflow components: DAGs, Scheduler, Executor, Metadata DB, Airflow UI
  - Set up a simple pipeline on a local machine
- Airflow Executors
  - Airflow with different types of executors
- Airflow With Docker
  - Set up Airflow with Docker
  - Implement advanced Airflow operators
  - Airflow with Kubernetes

SOFTWARE REQUIREMENTS

Apache Airflow, VS Code, Docker, terminal

ML Introduction & Data Preparation

4 hours of instruction

This course kickstarts the series of courses on MLOps technical implementation. It lays the ground for MLOps terminology and lets the students dive into the initial preparatory stages of setting up cloud platform services required to prepare data for the next stages of the ML cycle in production environment.

OBJECTIVES	PREREQUISITES
1. Set up and configuration of services required to perform data preparation 2. Data ingestion, exploring and validation	Intro to MLOps Theory

SYLLABUS & TOPICS COVERED

- Data Processing And ML Pipelines
  - Data engineering overview for ML pipelines
  - Set up and configuration of services required to perform data preparation

SOFTWARE REQUIREMENTS

Access to AWS accounts; computer with terminal; Anaconda, Python; JupyterLab/Jupyter Notebook; VSCode, API Gateway

Model Development & Testing

6 hours of instruction

This practical, hands-on course dives into the next stages in the ML production cycle - model development and testing.

OBJECTIVES	PREREQUISITES
1. By the end of this course, participants will be able to train, test, and evaluate models and pick the best model 2. The students will also set up all the necessary cloud platform services required to perform the above tasks	ML Introduction & Data Preparation

SYLLABUS & TOPICS COVERED

- Components Of ML Pipeline
  - Model development
  - Model training, testing and packaging
- Creating ML Pipeline
  - Model training, feature engineering, boosting
  - Hyperparameter tuning
  - Model evaluation and testing

SOFTWARE REQUIREMENTS

Access to AWS accounts; computer with terminal; Anaconda, Python; JupyterLab/Jupyter Notebook; VSCode, API Gateway

Model Packaging, Deployment & Monitoring

4 hours of instruction

This practical, hands-on course dives into the details of implementation of model deployment - the essential part of the ML cycle in production.

OBJECTIVES	PREREQUISITES
1. By the end of this course, participants will be able to package and deploy models using the infrastructure they have set up using the chosen cloud provider services	Model Development & Testing

SYLLABUS & TOPICS COVERED

- Model Deployment On AWS
  - Set up and configuration of services required to perform model training, evaluation, testing, and packaging
  - Model packaging
  - Model artifact deployment

SOFTWARE REQUIREMENTS

Access to AWS accounts; computer with terminal; Anaconda, Python; JupyterLab/Jupyter Notebook; VSCode, API Gateway

MLOPs CI/CD Theory

4 hours of instruction

This theoretical course gives a comprehensive overview of the components that make up the emerging trend of MLOps. It highlights the importance of Continuous Integration (CI) and Continuous Delivery (CD) in MLOps space and enables teams to adopt automation in model building, testing, and deployment.

OBJECTIVES	PREREQUISITES
1. Explain the role of continuous integration and continuous deployment pipelines in machine learning 2. Describe the components of a typical ML pipeline	Model Packaging, Deployment & Monitoring

SYLLABUS & TOPICS COVERED

- Intro To CI/CD For ML Ops
  - Components of Continuous Integration/Continuous Deployment in ML
  - Importance of CI/CD in MLOps
- CI/CD Pipelines For ML Theory
  - ML pipeline definition
  - Importance of CI/CD in machine learning
  - Components of a typical ML pipeline

SOFTWARE REQUIREMENTS

None

Creating CI/CD Pipeline for Machine Learning (ML)

6 hours of instruction

This practical, hands-on course recaps and ties together all stages of ML cycle in production into an automated CI/CD pipeline.

OBJECTIVES	PREREQUISITES
1. By the end of this course, participants will be able to tie all stages of ML and create automated pipelines on AWS	MLOPs CI/CD Theory

SYLLABUS & TOPICS COVERED

- Constructing CI/CD Pipeline Tasks
  - Identifying tasks for automation based on the business use case
  - Creating a workflow diagram for the CI/CD pipeline
- CI/CD Services For ML
  - Identifying the services and tools needed for the CI/CD pipeline
  - Setup and configuration of services required to create an automated CI/CD pipeline
- Testing And Revising Pipelines For ML
  - Overview of the need for human intervention as part of the pipeline
  - Revising the checkpoints for human intervention and making adjustments to process

SOFTWARE REQUIREMENTS

Access to AWS accounts; computer with terminal; Anaconda, Python; JupyterLab/Jupyter Notebook; VSCode, API Gateway

Testing ML Pipelines

6 hours of instruction

This practical, hands-on course dives into testing of an entire ML pipeline starting from data and model validation and ending with the integration tests of the pipeline as a whole.

OBJECTIVES	PREREQUISITES
1. By the end of this course, participants will be able to test and troubleshoot components of ML pipeline	Creating CI/CD Pipeline for Machine Learning (ML)

SYLLABUS & TOPICS COVERED

- Testing For ML Pipelines
  - Role of tests for ML pipelines
  - Tools and frameworks used to create robust testing frameworks
- Integration Testing Framework
  - Create a list of tests to be performed on the pipeline
  - Build integration testing framework for the ML pipeline

SOFTWARE REQUIREMENTS

Access to AWS accounts; computer with terminal; Anaconda, Python; JupyterLab/Jupyter Notebook; VSCode, API Gateway