CLARUSWAY

FULL-STACK

MODULES CURRICULUM

CLARUSWAY

WAY TO REINVENT YOURSELF

FULL-STACK

MODULES CURRICULUM

CLARUSWAY

Overview

A full-stack developer is a programmer who has the basic knowledge of all layers of an application with the ability to design complete web applications and websites which requires both front-end and back-end skills.

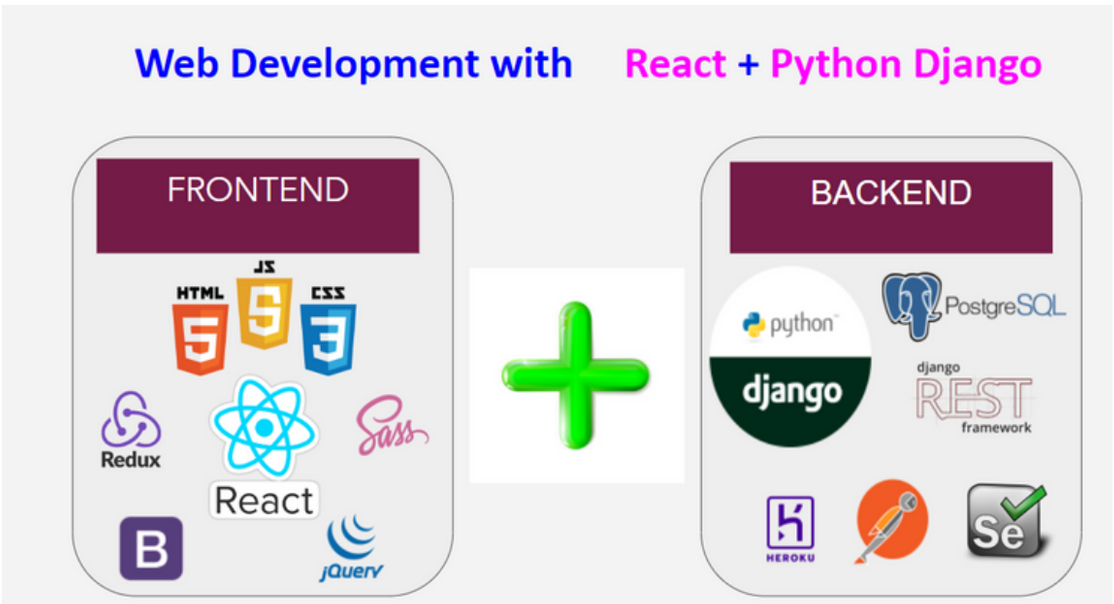
The full-stack developer roles require coding and building applications, websites, or mobile apps, working with multiple programming languages and databases. Our comprehensive program is a good opportunity for those who are interested in software development and want to transform their careers in this field.

Regardless of your previous career, you can step into the IT world with a 17+ weeks dedicated study paired with professional guidance. You can find a respectable job with a good income even before graduation like many of our students. Are you ready to give your career a boost?

The role of software in our daily lives has almost exploded due to the global COVID-19 pandemic. This situation led to an exponential increase in demand for software developers - employment opportunities in this area have increased more than ever and seem to increase more in the coming days. The United States Bureau of Labor Statistics estimates full-stack development employment to increase from 135,000 to over 853,000 by the year 2024.

Clarusway Full Stack

Full-stack development is basically union of Frontend (client side) and Backend (server side) development. Clarusway Full Stack training will provide you most popular skills that is required in the market in the job listings.



Clarusway Full Stack

At the end of the course, having achieved all the skills of full-stack developer, you will be able to develop an app in agile development team.

You fill up your resume with in-demand Full Stack Developer skills:

**** Computational Thinking***

**** Git / GitHub,***

**** Linux,***

**** Agile/Jira,***

**** HTML,***

**** CSS,***

**** Bootstrap,***

**** SASS,***

**** JavaScript,***

**** TypeScript,***

**** jQuery,***

**** React,***

**** Redux,***

**** FireBase,***

**** Pyhton,***

**** Django Web Framework,***

**** Django REST,***

**** SQL and PostgreSQL,***

**** Deployment*** to popular hosting web sites,

**** Testing with Selenium,***

**** Capstone*** Projects

- In addition, other helpful support that a data analyst/scientist needs are also included in the course:

  ***** Group and 1on1 ***mentorships***,

  **** Career Management System*** (Resume, LınkedIn support),

  **** Informative Sessions*** (Experience Sharing),

  ***** Time Sensitive ***Real Interview Tasks***,

  ***** Capstone ***Project Presentations*** (all mentor group members),

  **** Project competitions*** between mentor groups,

  **** Mock Interviews***

  ***** etc.

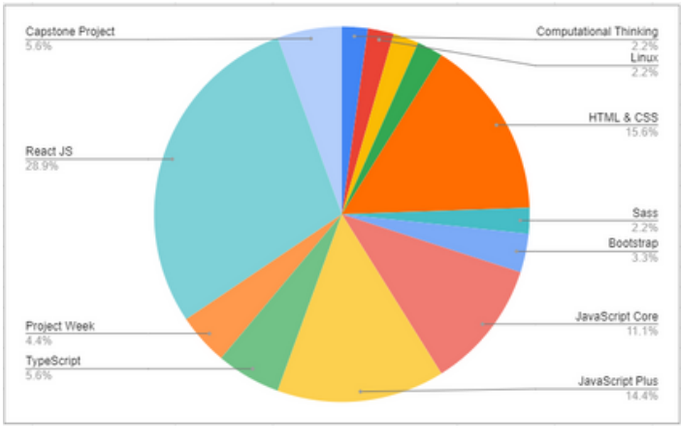
Lectures Breakdown

The Full Stack course consists of 2 Modules. Frontend module (FE) and Backend Module (BE). Frontend module takes 18 weeks and Backend Module takes 12 weeks.

Below is a table showing the distribution of Courses available in the FE and BE Modules their total hours and weight percentages.

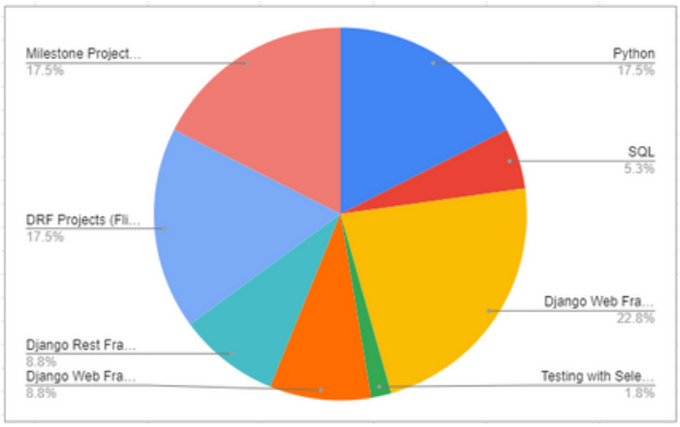
Frontend Module - Part Time

FRONTEND DEVELOPMENT MODULE	Total Hours
Computational Thinking	6
Linux	6
Git	6
Agile & Jira	6
HTML & CSS	42
Sass	6
Bootstrap	9
JavaScript Core	30
JavaScript Plus	39
TypeScript	15
HTML CSS JS Project	12
React JS	78
Capstone Project	15
Total	**270**

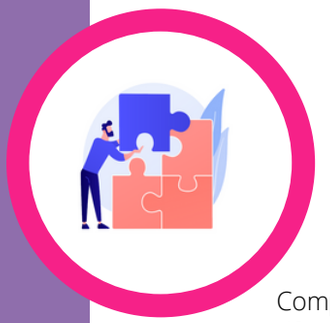


Backend Module - Part Time

BACKEND DEVELOPMENT MODULE	Total Hours
Python	30
SQL	9
Django Web Framework	39
Testing with Selenium	3
Django Web Framework Project (Blog App)	15
Django Rest Framework (DRF)	15
DRF Projects (Flight App, Quiz App)	30
Milestone Project Period	30
Total	**171**



Lectures and Learning Outcomes



Computational Thinking (CT)

COMPUTATIONAL THINKING DEALS WITH A SET OF PROBLEM-SOLVING METHODS THAT INVOLVE EXPRESSING PROBLEMS AND THEIR SOLUTIONS IN WAYS THAT A COMPUTER

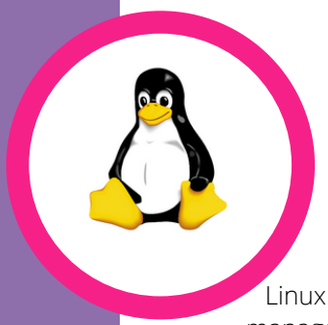
Computers can be used to help us solve problems. However, before a problem can be tackled, the problem itself and the ways in which it could be solved need to be understood. Computational thinking allows us to do this. Computational Thinking allows us to take a complex problem, understand what the problem is and develop possible solutions. We can then present these solutions in a way that a computer, a human or both can understand.

Computational Thinking involves taking that complex problem and breaking it down into a series of small, more manageable problems (decomposition). Each of these smaller problems can then be looked at individually, considering how similar problems have been solved previously (pattern recognition) and focusing only on the important details, while ignoring irrelevant information (abstraction). Next, simple steps or rules to solve each of the smaller problems can be designed (algorithms).

At the end of this Lecture, the students know the high-level principles, as well as the historical and theoretical backgrounds, for solving problems efficiently by using computational tools and information-processing agents. The students are able to understand and use the main data structures for organising information, to develop algorithms for addressing computational-related tasks, and to implement such algorithms in a specific programming language.

The learning objectives of Computational Thinking Lecture are:

- To define computational thinking.
- To define the terms decomposition, pattern recognition, abstraction and algorithmic thinking.
- To analyze the complex problems to draw conclusions.
- To recognise scenarios where each of these computational thinking techniques are applied.
- To apply decomposition, pattern recognition, abstraction and algorithmic thinking to help solve a problem.



Linux (OS)

LINUX IS AN FREE AND OPEN-SOURCE OPERATING SYSTEM LIKE OTHER OPERATING SYSTEMS SUCH AS MICROSOFT WINDOWS, APPLE MAC OS THAT ENABLES THE COMMUNICATION BETWEEN COMPUTER HARDWARE AND SOFTWARE

Linux operating system or a kernel distributed under an open-source license, is the software that directly manages a system's hardware and resources, like CPU, memory, and storage. The OS sits between applications and hardware and makes the connections between all of your software and the physical resources that do the work. Its functionality list is quite like UNIX. The kernel is a program at the heart of the Linux operating system that takes care of fundamental stuff, like letting hardware communicate with software.

The learning objectives of this lecture are:

- To describe the relationship between GNU and Linux.
- To describe the relationship between Linux and Unix.
- To discuss features which make Linux a viable and popular operating system.
- To explain the structure of the Linux operating system
- To explain the different Linux distributions and their usage.
- To understand and use Essential Linux commands to manage files and file systems
- To manipulate on permissions of Linux users
- To manipulate on some editors used for Linux commands.

GIT / GitHub

GIT IS SOFTWARE FOR TRACKING CHANGES IN ANY SET OF FILES, USUALLY USED FOR COORDINATING WORK AMONG PROGRAMMERS COLLABORATIVELY DEVELOPING SOURCE CODE DURING SOFTWARE DEVELOPMENT

Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency. Git is easy to learn and has a tiny footprint with lightning fast performance. Besides, Github is a web-based platform used for version control. Git simplifies the process of working with other people and makes it easy to collaborate on projects. Team members can work on files and easily merge their changes in with the master branch of the project. In this Lecture, you'll learn how to keep track of the different versions of your code and configuration files using a popular version control system (VCS) called Git. We'll also go through how to set up an account with a service called GitHub so that you can create your very own remote repositories to store your code and configuration.

Throughout this course, you'll learn about Git's core functionality so you can understand how and why it's used in organizations. We'll look into both basic and more advanced features, like branches and merging. We'll demonstrate how having a working knowledge of a VCS like Git can be a lifesaver in emergency situations or when debugging. And then we'll explore how to use a VCS to work with others through remote repositories, like the ones provided by GitHub. By the end of this Lecture and Cohort, you'll be able to store your code's history in Git and collaborate with others in GitHub, where you'll also start creating your own portfolio!

The learning objectives of GIT / GitHub Lecture are:

- To understand why version control is a fundamental tool for coding and collaboration.
- To install and run Git on your local machine.
- To explain the difference between Git and GitHub.
- To use the shell commands to do basic version control with Git.
- To use and interact with GitHub.
- To collaborate with others through remote repositories.

Agile

AGILE IS A SOFTWARE DEVELOPMENT METHODOLOGY TO BUILD A SOFTWARE INCREMENTALLY USING SHORT ITERATIONS OF 1 TO 4 WEEKS SO THAT THE DEVELOPMENT IS ALIGNED WITH THE CHANGING BUSINESS NEEDS

Agile is a methodology often applied toward software development that aids with project management. This project management methodology helps teams deliver value to their customers faster and with fewer issues. Some of the benefits of the Agile methodology are Improved quality, increased focus on business value, increased focus on users, stakeholder engagement, transparency, and the methodology allows for changes as needed.

The learning objectives of the Agile Lecture are:

- To learn the meaning of user stories, daily stand-ups, retrospectives and kanban boards.
- To learn the key concepts of Agile Development, Agile Project Delivery and Agile Project Management.
- To understand the differences between Agile and traditional project delivery, other methodologies.

Jira

JIRA SOFTWARE IS PART OF A FAMILY OF PRODUCTS DESIGNED TO HELP TEAMS OF ALL TYPES MANAGE WORK

Jira is a cloud and subscription-based issue tracking tool but also designed to handle team coordination in agile software development as well. It offers a comprehensive suite of bug tracking throughout the entire software development lifecycle. This software makes teams work towards a common goal and facilitates planning, tracking as well as the release of the software.

HTML5

You will learn the necessary tools to make the next generation web pages by working on more complex structures on top of what you have learned in the IT fundamentals part. Students are expected to code their own web applications from scratch using HTML5. Lectures;

Introduction to HTML
HTML Elements and Tags
HTML Text
HTML Formatting Elements
HTML Lists
HTML Tables
HTML Images
HTML Links
HTML Forms
HTML Input Types
HTML Form Elements

CSS3

All the fundamentals of CSS will be covered including the idea of selectors, methods of setting colors and backgrounds, way of formatting fonts and texts, styling UI elements such as hyperlinks, lists, tables, as well as the concept of the CSS box model, etc.

Once you're comfortable with the basics, you'll move on to the next topics that explain the way of setting dimension and alignment of elements, methods for positioning elements on a web page, using image sprites, as well as the concept of relative and absolute units, visual formatting model, display and visibility, layers, pseudo-classes and elements, media-dependent style sheets, and much more. Lectures;

Introduction to CSS
Use of CSS
Getting into CSS
CSS Colors
CSS Properties
Opacity/Transparency Properties
Units in CSS
CSS setting Height and Width
CSS Outline
CSS Combinators
CSS Flex-Box
CSS Media Query
CSS Grid

Sass (Syntactically Awesome Style Sheets)

Sass is the most mature, stable, and powerful professional-grade CSS extension language in the industry. It is CSS with superpowers. Lectures;

- Variables
- Nesting
- Partials and @importLesson
- Mixins and @include
- Inheritance and @extend
- Operators
- Control Directives and Expressions

Bootstrap5

Bootstrap is a powerful front-end framework for faster and easier web development. It includes HTML, CSS, and Javascript-based design templates for creating common user interface components like forms, buttons, navigations, dropdowns, alerts, modals, tabs, accordions, carousels, tooltips, and so on.

Bootstrap gives you the ability to create flexible and responsive web layouts with less effort. Bootstrap saves you from writing lots of CSS code, giving you more time to spend on designing webpages.

Bootstrap Basic Classes
Bootstrap Components
Bootstrap Layout

JavaScript

JavaScript is a programming language to make interactive websites. If you are willing to be a front end developer, JavaScript is a must. On the other hand, as a full-stack web developer, you can make your life with JavaScript easier using back-end development with Nodejs.

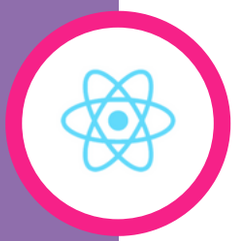
JavaScript is the most popular language in the world, a head-to-head race with Python. According to the Devskiller IT Skills and Hiring Report 2020 72% of companies are seeking to hire JavaScript developers. Lectures;

- Introduction to JavaScript
- Variables
- Data Types
- Operators
- Conditionals
- Loops
- Functions
- Strings
- Arrays
- Iterators
- Objects
- The Document Object Model (DOM)
- DOM Events
- Callback Functions
- Asynchronous Programming
- Object Oriented Programming (OOP)

TypeScript

TypeScript is a strongly typed programming language that builds on JavaScript, giving you better tooling at any scale. It is a strict syntactical superset of JavaScript and adds optional static typing to the language. It is designed for the development of large applications and transpiles to JavaScript.

Lectures; Configuration, Types and Variables, React with TypeScript



React.js

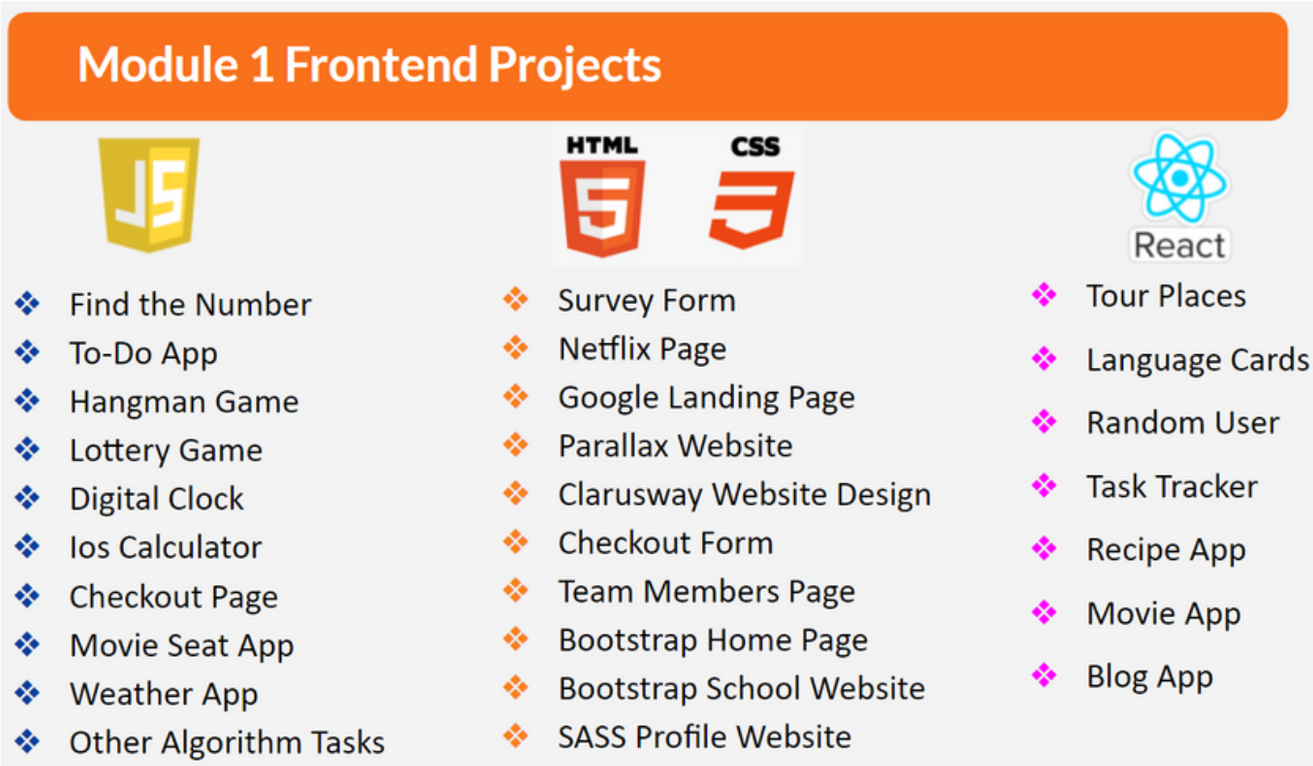
React is remarkably flexible. Once you have learned it, you can use it on a vast variety of platforms to build quality user interfaces. React is a library, NOT a framework. Its library approach has allowed React to evolve into such a remarkable tool. And the most demanded FrontEnd skill is experience with ReactJs. You will make many mid and high level projects with these trend technologies. Lectures;

- Introduction
- JSX & Components
- Styling in React
- The component Lifecycle
- Events in React
- Hooks
- React Router
- Context
- React Redux & Redux Thunk

Hands-on Projects during Front End

A DIFFERENT NUMBER OF ASSIGNMENTS AND PROJECTS FOR EACH COURSE DURING THE MODULE IS A NECESSITY IN TERMS OF REINFORCING THE LEARNED SUBJECT.

In the Front End Module, in addition to many assignments and mini-projects, a capstone project covering all development skills will be done.



Backend Module

Overview

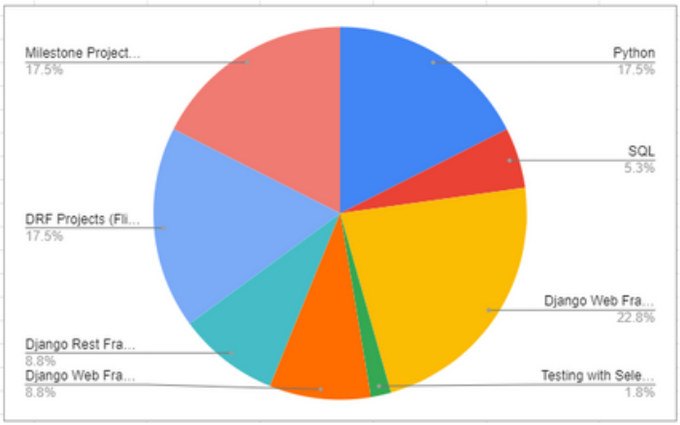
Simply the Backend Developer is the cooker who gives functionality behind the scenes.
The backend developer is responsible for all the coding that runs on the server side.
This is the code that makes the internet services running behind the scenes, such as connection to a database, web applications, integration of frontend products, Application Programming Interfaces (APIs). Backend developers work closely with the frontend team to deliver the final product.
The technology on the backend is a combination of servers, applications, and databases.
Backend developers produce the web services and APIs which are used by frontend developers and mobile application developers.
Backend developers make the magic behind-the-scene activities that happen when a user performs any action on the website, such as posting a message, ordering food, making hotel reservations, buying a plane ticket, etc.
As a summary, backend development is simply called server-side development, which focuses on databases, programming, and website architecture.

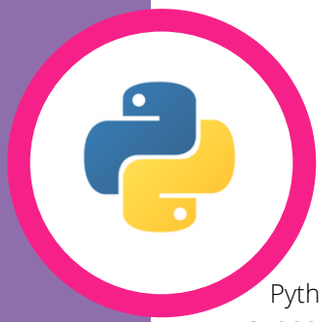
Lectures Breakdown

Backend Module takes 12 weeks. There is no prerequisite for this module.
Below is a table showing the distribution of Courses available in Back-End Module, their total hours and weight percentages.

Backend Module - Part Time

BACKEND DEVELOPMENT MODULE	Total Hours
Python	30
SQL	9
Django Web Framework	39
Testing with Selenium	3
Django Web Framework Project (Blog App)	15
Django Rest Framework (DRF)	15
DRF Projects (Flight App, Quiz App)	30
Milestone Project Period	30
Total	**171**





PYTHON Programming Languages

PYTHON IS AN INTERPRETED HIGH-LEVEL GENERAL-PURPOSE PROGRAMMING LANGUAGE THAT LETS YOU WORK MORE QUICKLY AND INTEGRATE YOUR SYSTEMS MORE EFFECTIVELY

Python Programming is intended for software engineers, systems analysts, program managers and user support personnel who wish to learn the Python programming language. Python's design philosophy emphasizes code readability with its notable use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python is used in virtually every industry and scientific field that can imagine, including; Data Science; Machine Learning; Web Development; Computer Science Education;Computer Vision and Image Processing; Game Development; Medicine and Pharmacology; Biology and Bioinformatics; Neuroscience and Psychology; Astronomy and other areas such as robotics, autonomous vehicles, business, meteorology, and graphical user interface (GUI) development.

The learning objectives of Python Programming Language Lecture are:

- To understand why Python is a useful scripting language for developers.
- To learn how to design and program Python applications.
- To learn how to use lists, tuples, and dictionaries in Python programs.
- To learn how to identify Python object types.
- To learn how to use indexing and slicing to access data in Python programs.
- To define the structure and components of a Python program.
- To learn how to write loops and decision statements in Python.
- To learn how to write functions and pass arguments in Python.

Django

Python is one of the most popular programming languages. In the Module-2, you will take your python knowledge much further that you learned in the IT fundamentals section. With the javascript you learned in the frontend and the python you learned in this section, you will master both popular programming languages.

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so you can focus on writing your app without needing to reinvent the wheel. Once you learn Django, we'll teach you Django Rest, and you can build api endpoints very easily with Django.

Back end developers build the “under the hood” parts of websites that users don’t interact with directly. This involves using “server side” programming languages to request data from a site’s webserver, which then appears as HTML output on a user’s screen.

SQL

SQL IS THE STANDARD QUERY LANGUAGE FOR MANAGING RELATIONAL DATABASES. IT ENABLES DEVELOPERS TO CREATE EFFICIENT SOFTWARE QUICKLY BY LEVERAGING WELL-STRUCTURED DATA

A database management system (DBMS) is a software used to store and manage data. It guarantees the quality, durability, and confidentiality of information. The most popular type of DBMS are Relational Database Management Systems, or RDBMSs. Here, the database consists of a structured set of tables and each row of a table is a record. On the other hand Structured Query Language (SQL) is the standard language for data manipulation in a DBMS. In simple words it's used to talk to the data in a DBMS. A Query is a set of instruction given to the database management system. It tells any database what information you would like to get from the database.

Through the lecture SQLite is being used. It is an open-source, embedded, relational database management system. It is a lightweight database, with zero configuration, no requirements of a server or installation. SQLite is not a client-server database management system. It is an in-memory library that you can call and use directly. No installation and no configuration required. Despite its simplicity, it is laden with popular features of database management systems.

The learning objectives of SQL Lecture are:

- To learn Structured Query Language (SQL) to an intermediate/advanced level.
- To be able to write data retrieval queries and evaluate the result set.
- To be able to write SQL statements that edit existing data.
- To be able to write SQL statements that create database objects.
- To understand the structure and design of relational databases.
- To understand the importance and major issues of database security and the maintenance of data integrity.
- To retrieve data from single or multiple tables.
- To process data with row and aggregate functions.
- To manipulate data with correlated and non correlated subqueries.

Hands-on Projects

- Simple CRUD App
- ToDo App
- Pizza App
- Flight App
- Quiz App
- Stock App
- Blog App

Teamwork

Teams are formed from students. Every week a student leads their team's teamwork activity and a mentor from Clarusway attends these meetings as an observer. In these meetings, students work together on the teamwork document. In this weekly document, students solve the questions they have learned on a daily basis. They are trying to do the code challenge given to them. They learn by discussing the interview questions in this document. Teams can work together not only during teamwork activities, but also during projects and at any time they want. We match students as code buddies with their friends in the team and encourage and follow them to do pair coding. We change code buddies among themselves at certain periods. Thus, they see different approaches to a project among themselves. Teams use Jira and GitHub tools while managing their projects as a team.

Weekly Schedule

According to our current schedule, live lessons and other activities are planned and implemented in Turkish according to the CET time zone and in English according to the EST time zone. Our purpose in doing so is to provide students with an opportunity to participate in different sessions flexibly in a wider time frame, day and night. Students participating in the Turkish sessions can also participate in the English period if they wish. Our lessons which we call "in-class" are planned every Monday, Wednesday, Thursday, and Saturday.
Different sessions like Warm-ups, Workshops, Labs, Teamworks and Career Coaching's are planned as both individual and group activities before and after the classes. As an addition to live lecture sessions on all days including weekends, we have separate slots where students can ask questions and get instant answers using slack channels.

In addition, we have Projects, Assignments, Homeworks, etc. that allow our students to practice as post-class and extra curricular activities done by students first, and then solution sessions will be held by instructors if needed.

Typical weekly **part time** schedule

Time Zone	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
EDT	07-10 pm	07-10 pm	07-10 pm	07-10 pm	07-10 pm	10 am - 05 pm	-------
	Online Live Class	Online Live Class	Online Live Class	Online Live Class	Workshop Teamwork CMS	Workshop Lab Online Live Class	Freeday

www.clarusway.com

contact@clarusway.com

+1 (571) 360 66 77

contact@clarusway.com

1775 Tysons Blvd. 5th Floor.
Tysons, VA 22102