

**TOST Group FULL STACK
DEVELOPER COURSES
SYLLABUS DRAFT**

www.trainings.tostgroup.com



HTML

- 1.Introduction to HTML
- 2.Browsers and HTML
- 3.Recommended w3c structure for HTML page.
- 4.Heading fonts(H1-H6), creating a list, attaching links, creating form and tables, adding images.
- 5.Editor's Offline and Online
- 6.Tags, Attribute and Elements
- 7.Doctype Element
- 8.Comments
- 9.Headings, Paragraphs, and Formatting Text
- 10.Lists and Links
- 11.Images and Tables



CSS

- 1.Introduction CSS
- 2.Applying CSS to HTML
- 3.Selectors, Properties and Values
- 4.CSS Colors and Backgrounds
- 5.CSS Margins, Padding, and Borders
- 6.CSS Text and Font Properties
- 7.Selectors, how many types of CSS usage are there?
- 8.Basic Styling, Relative & Absolute.
- 9.Media Query, Responsive Design, Bootstrap.
- 10.Grid & Flex, Position, Box Model.
- 11.CSS General Topics



JAVASCRIPT

1. Introduction to JavaScript
2. Applying JavaScript (internal and external)
3. Understanding JS Syntax
4. Introduction to Document and Window Object
5. Variables and Operators
6. Data Types and Num Type Conversion
7. Math and String Manipulation
8. Objects and Arrays
9. Date and Time
10. Conditional Statements
11. Switch Case
12. Looping in JS
13. Functions
14. Map, Rest, forEach, reduce, set timeout, Async functions,
15. Callback >>> Promises >>> async-await,
16. Convert callback to promises.
17. Template string/literals



ReactJS

- 1.Introduction
- 2.Templating using JSX
- 3.Components, State and Props
- 4.Lifecycle of Components
- 5.Single Page Application
- 6.Rendering List and Portals
- 7.Multi-Page Application, Babel.
- 8.Error Handling
- 9.Routers
- 10.Redux and Redux Saga
- 11.Immutable.js
- 12.Service Side Rendering
- 13.Unit Testing
- 14.Webpack
- 15.Hooks



NodeJS

1. Node js Overview
2. Node js - Basics and Setup
3. Node js Console
4. Node js Command Utilities
5. Node js Modules
6. Node js Concepts
7. Node js Events
8. Node js with Express js
9. Node js Database Access
10. Simple Server
11. Response Types– HTML, JSON
12. Routing.
13. API Authentication
14. JWT token
15. Microservice Introduction



Mongo Database

1. What is Database?
2. Types of Database
3. SQL/No-SQL
4. What is Mongo, JSON/BSON
5. Mongo Structure, Architecture
6. Integration, Atlas, What is _id
7. CRUD with MongoDB Atlas from CMD
8. Operators Mongo Sharding
9. Mongo Master-Slave
10. ACID
11. Mongo Compass & NoSQL Booster Mongoose
12. Schema & Schema Types, Model
13. Indexes, Connection to MongoDB
14. Document, Sub Document
15. Middlewares
16. CRUD with Mongoose



Python

1. Python Installation & Configuration
2. Developing a Python Application
3. Connect MongoDB with Python



VCS

Learn version control with Git. This is a bankable skill for every web developer so that they can collaborate and work as a team while working individually. It is also imperative to use VCS so that you can simplify the work on identifying errors and also store every piece of coding that you've worked on.



GIT

1. Why the command line?
2. Installing Git (Windows Only)
3. Notes about Git for Windows
4. Installing Git (MacOS and Linux)
5. Getting started with GitHub
6. Configuring Git on your computer
7. Creating and adding an SSH Key
8. How to clone a repository?
9. How to create a new repository on GitHub?
10. How to push to your GitHub repository
11. Git status
12. Un-staging a file
13. Undeleting a file
14. Git origins and remotes
15. Git branching
16. Committing to a new branch
17. Merging a branch into master
18. Seeing your Git history
19. Downloading updates from GitHub
20. How to get updates from GitHub?
21. Checkout: code-time travel
22. README.md files



Understanding API: Restful APIs

- 1.The simplest way to think about an API (the metaphor)
- 2.A real life use case
- 3.What programming languages can we use?
- 4.Introduction to RESTful APIs
- 5.Introduction to JSON
- 6.Using SWAPI Notes
- 7.A real JavaScript API example
- 8.GET requests
- 9.POST requests
- 10.DELETE requests
- 11.PUT/PATCH requests
- 12.Consuming APIs (how to use them)
- 13.Requests and responses
- 14.Common status codes
- 15.API security



Part I.

Introduction to PHP

1. What Is PHP?
2. Installing PHP
3. Syntax & Hello World
4. How to write comments in PHP?
5. Introduction to PHP Variables
6. Strings & Concatenation
7. Replace areas of a string
8. Manipulate other areas of a string
9. Displaying HTML entities
10. PHP Lists: Arrays
11. Creating and destroying arrays: Explode & Implode
12. Operators
13. If...Else Statements
14. Else if Statements
15. Switch Statements
16. For Loops and Foreach Loop
17. While Loops
18. Break from Loops
19. continue, is array (), and quotes
20. Including Files (Code reusability)
21. Functions
22. Constants. The unchangeable variable.

Part II.

PHP Object Oriented Programming

1. PHP OOP
2. What is php oop
3. Classes
4. Constructor
5. Destructor
6. Access Modifiers
7. Inheritance
8. Class Constants
9. Abstract Classes and Methods
10. PHP OOP - Interfaces
11. PHP - What are Traits?
12. PHP OOP - Static Methods
13. PHP OOP - Static Properties
14. PHP Namespaces
15. PHP - What is an Iterable?



MySQL Database

1. What is SQL?
2. Setting up a Database
3. Introducing phpMyAdmin
4. MySQL Syntax
5. Tables, Columns and Rows
6. Connecting to a Database
7. Creating a Table: 5 Columns and 3 Rows.
8. Inserting Information
9. Escaping Information
10. Get information that's stored in a database
11. Selecting specific information: The WHERE clause
12. Selecting specific information: The And/Or clauses
13. Limiting results and ordering data
14. Updating the database
15. Deleting stored information

Mini Project

The mini project is the endgame of your full stack development course. This is a medium for you to apply and exhibit what you have learnt in this course. You get to take your theoretical learning and apply it in a real-world setting. This is a very important aspect of the full stack development course syllabus.

The prerequisite for a mini project is that you have completed the learning modules in the full stack development course, taken up on practicing code at lab sessions and also completed all your mini-projects, which will give you an idea how to get working on your mini project. You can build web apps that are fully functional.

What are the learning takeaways from the mini project?

Research skills- You have to research a valid real time problem and make copious notes on it before getting to the solution part.

Documentation- This is a very important part of the work for every developer. Keep precise notes and documentation of what you have created so that it is easy for you as well as others to understand and follow.

Coding- You learn to code and work with tools extensively for the project.

Experience - You get practical experience. When coding, not everything goes smoothly as in theory. You stumble into all sorts of errors while testing and writing codes. So this project will give you a taste of how the job is actually going to be.

Version control- You will save your work every time in the Git repository and handle versions. This will also help you collaborate with teams and work towards a common goal.

Portfolio - You get a jump start on building your portfolio. You can use your mini project and present it to a recruiter to show how you work and what an asset you'll be to their team.

Here are some mini-project ideas that students can work on at TOST GROUP!

- Question and answer sites
- School management systems
- Project management tools
- Social media sites
- Ticket booking sites

Mini Project

Minimum 5+ mini projects will be assigned to the learners. They may be module(s) based. You need to work them out and finish within the given time period. These will help you understand coding and how to work on full stack tools for a specific purpose. These projects include topics like creating a personal portfolio page or a landing page for a product and shopping sites.

Lab Sessions

These are hands-on practice sessions organized to help you try coding. There will be 16 sessions that will be marked labs.

In this time, you get a chance to work on your coding expertise. Your Trainers and Mentors will help you with the learning process. You get to try out text resizers using DOM manipulation, logic puzzles using onde.js, random quote generator, CRUD ops for node and mongo, google search pages and many more.

Requirements

1. Must be willing to learn.
2. High school or university diploma
3. Must be able to put in 100 minutes per day (that's all it takes!)
4. Must have a computer, editor and browser (We will help you get setup with those)

Contact us:



(+250)788844934



info@tostgroup.com



2th Floor, Prince House, RN3, RN15 St /Kigali-Rwanda



@tostgroup