

Byte Academy Tech Syllabus Timeline

Full-Stack Python



Common topics throughout course

- Git workflow
- Unix/Linux usage
- Debugging
- Group weekend projects

Phase 1-1: Python Basics

- Programming Fundamentals: Classes, Iteration, Control flow, Class Organization
- Utilize Git and Github
- Learning Bash commands and navigating your terminal
- Best Practices: Keeping it simple, DRY code, naming conventions, comments and documentation
- Python mini project: Well- documented Python module; (those are one to two-hour mini project which could be finished in class so that the students can discuss with peers and get feedback from the teacher/TA.)
- Python Standard Library (standard libraries typically include definitions for commonly used algorithms, data structures, and mechanisms for input and output.)

Phase 1-2: Computer Science, Beyond the basics

- Introduction to Computer Science
- Big O Notation, Data Structures, Sorts and Searches
- MVC - Model Views Controller
- SQL Introduction
- (Weekend) - Building a terminal application utilizing the MVC Design pattern and a SQL Database for persistent data

Phase 1-3: Databases

- SQL Relationships
- SQL Joins
- Introduction to APIs
- CRUD and HTTP Verbs
- (Weekend) - Building a terminal application with the MVC Design pattern, persisting data in SQL, and utilizing APIs to grab data in JSON format

Phase 2 - 1: Advanced Python

- Intro to Pandas
- Intro to Matplotlib
- Review phase 1
- Phase 1 Assessment
- (Weekend) - Introduction to HTML, CSS, and JavaScript

Phase 2 - 2: Front end

- Review intro to HTML, CSS, and JavaScript
- Higher Order Functions. Callbacks, Closure
- JavaScript Scope
- Document Object Model
- Event Listeners
- jQuery
- CSS Specificity
- CSS Pseudo Classes
- CSS Positioning
- CSS Media Queries
- CSS Grid Systems
- CSS Responsive Design
- Chrome Dev Tools
- (Weekend) - Building Tic Tac Toe, Blackjack, Connect Four

Phase 2 - 3: Backend

- AJAX Introduction
- What happens when you type google.com in the browser
- Introduction to Flask
- Request Response Cycle
- What is the Request Object
- Setting up virtual environments
- HTML Forms
- Jinja Templating
- HTTP Verbs
- Sending a response in different formats. (JSON, HTML Templates)
- SQLAlchemy Introduction
- (Weekend) - Building a full stack web application that will make requests to HTTP API's, and persist information in a SQL database. Utilize HTML, CSS, JavaScript, AJAX, Flask, and SQL

Phase 2 - 4: Going deeper with the backend

- Password Hashing
- User Sessions
- REST
- Single Page Applications
- (Weeklong / Weekend Part 1) - 2-3 days building a RESTful API in Flask that will talk to a SQL Database, accept requests, and return JSON Responses
- (Weeklong / Weekend Part 2) - 2-3 days building a Single Page Application that will consume the RESTful API just built using HTML, CSS, JS, AJAX, and Flask

Phase 2 - 5: Advanced web technologies

- Introduction to Django
- Django Directory Structure
- Django Models
- Django Forms / Templating
- Building more full stack applications
- Review for Phase 2 Assessment

Phase 3 -1: Final Project Phase, Deep Dive Lectures, and Mock Interviews

- Students will do three projects
- Each project done in 2 week sprints
- There will be 1 group project, 1 solo project, and the third project is optional between solo or group
- Lectures during this phase will adapt towards student interests and their projects
- Lectures include
 - SASS/SCSS
 - Flexbox
 - D3.js / C3.js
 - ES6
 - React
 - PostgreSQL
 - MySQL
- Deploy Final Projects on Digital Ocean