Software Development
Part-Time Online

18 - 34 weeks, 30 hours/week (Accelerated Program)
30 weeks, 10-15 hours/week (Flex Program)

Part-Time
class commitment

Career Path Focus
built into curriculum

Learn by Doing
real projects, real datasets

Join our 13,000+ global alumni and kickstart your career path in tech.
Your career path into software development begins on your first day of class. In 18 to 34 weeks, you'll study to become a self-sufficient, versatile developer who has the critical skills to pursue a career path in tech.

Anyone can learn to code, but the path to becoming a developer isn’t easy. You’ll start coding from day one. Dive into a fast, project-based learning environment that fosters collaboration, not competition.
Choose Between Two Options to Fit Your Schedule:

1. **Accelerated Program**
   Our accelerated program allows you to choose your own adventure! Choose 1, 2, or 3 full stacks at a part-time pace.
   - 18 - 34 Weeks
   - 30 Hrs/Week
   Includes complete web fundamentals, then choose from the following stacks:
   - Python
   - Javascript
   - Java

2. **Flex Program**
   Our flex program allows students to learn Python on a more accommodating schedule.
   - 30 Weeks
   - 10-15 Hrs/Week
   Includes complete web fundamentals, and Python (only Python is available through Flex at this time).
   - Python

Up Next: About the Accelerated Program
About the **Accelerated Program**

**Learn to build applications** in some of the top programming stacks of 2023. Pick between Python, JavaScript, or Java as your stack, or choose to extend the program and explore multiple languages.

**Week One to Two**
**Programming Basics**
To kickoff the program, you’ll study habits, computer basics, and fundamental programming concepts and skills necessary to be successful in your bootcamp!

**You’ll Focus On:**
- Basic computer literacy
- Algorithmic foundations
- Learning stamina

**Week Three to Six**
**Web Fundamentals**
You’ll start with Web Fundamentals—a four week course that starts with the basics to provide a good overview before jumping into specific languages.

**You’ll Focus On:**
- HTML
- CSS
- Javascript

**Weeks Seven to Fourteen**
**Stack 1 (Python, JavaScript or Java)**
You’ll get to decide which stack you’d like to focus on, either Python, JavaScript, or Java.

**You’ll Focus On One of the Following:**
- Python
- JavaScript
- Java

**Optional Extra Stack**
* + Add 8 Weeks
  
  Choose from either Python, JavaScript, or Java.

**Optional Extra Stack**
* + Add 8 Weeks
  
  Choose from either Python, JavaScript, or Java.

**Last Four Weeks**
The last four weeks of the course focuses on putting together everything you’ve learned to create unique projects, as well as preparing for potential interviews with more in-depth programming knowledge.

**What You’ll Focus On:**
- Projects
- Algorithms

**Up Next:** A Day in the Life
An Example Day’s Schedule in an Accelerated Program

24/7 Cohort Access
Your access to our LEARN Platform and Discord is available 24/7. Access your materials at whatever time you need them.

Self Study
Most students dedicate 30-35 hours a week to self-study, though you may need more or less depending on your learning style and experience.

Lectures
Live lectures are held three times per week for an hour from 6pm-7pm MST. Lecture days are Tuesday, Wednesday, and Thursday.

Optional Office Hours
Need more assistance understanding a concept? Optional office hours are held an hour prior to lecture times between 5pm-6pm MST.
About the **Flex Program**

Learn to build applications in the same Python curriculum, over a longer amount of time, so you can manage the rest of your commitments more easily.

**Weeks One to Two**  
Programming Basics  
To kick off the program, you’ll explore habits, computer basics, and fundamental programming concepts and skills necessary to be successful in your bootcamp!

You’ll Focus On:  
- Basic computer literacy  
- Algorithmic foundations  
- Learning stamina

**Weeks Three to Ten**  
Web Fundamentals  
You’ll start with Web Fundamentals—a four week course that starts with the basics to provide a good overview before jumping into specific languages.

You’ll Focus On:  
- HTML  
- CSS  
- Javascript

**Weeks Eleven to Twenty-Six**  
Python  
You’ll dive into Python, the stack of the Flex Program, over the course of a 16 week program, at your pace. Unlike the Accelerated program, you do not have a choice of stack. You also do not have the option to add additional stacks.

You’ll Focus On:  
- Python  
- OOP  
- Flask  
- MySQL  
- Ajax*

**Weeks Twenty-Seven to Thirty**  
Projects & Algorithms  
The last four weeks of the course focuses on putting together everything you’ve learned to create unique projects, as well as preparing for potential interviews with more in-depth programming knowledge.

You’ll Focus On:  
- Projects  
- Algorithms

*Optional Topic
An Example Day’s Schedule in a **Flex Program**

**Morning**
Head to Work

**Evening**
Lecture, Office Hours & Self Study

**24/7 Cohort Access**
Your access to our LEARN Platform and Discord is available 24/7. Access your materials at whatever time you need them.

**Self Study**
Most students dedicate 10-15 hours a week to self-study, though you may need more or less depending on your learning style and experience.

**Lectures**
Live lectures are held **twice per week** for an hour from 6pm-7pm MST. Lecture days are **Monday/Wednesday** or **Tuesday/Thursday** depending on your cohort’s start date.

**Optional Office Hours**
Need more assistance understanding a concept? Optional office hours are held an hour prior to lecture times between 5pm-6pm MST.

**Up Next:** Let’s Dive Into the Stacks!
Let’s Dive Into the Stacks!

What does 3 stack mean?

A stack refers to a programming language, and when we refer to ‘full stack’, we mean you’ll study every facet of that programming language.

Stack One: Python
Python is one of the most popular languages in the industry\(^1\). Its diversity, adaptability, and easy-to-master basics makes it the perfect language to start with at bootcamp.

Stack Two: Javascript
JavaScript is ideal for building dynamic websites and applications. It runs on every application level making it an efficient, modern approach to web development.

Stack Three: Java
Java is a high-level language which revolutionized language development post-release. It’s adopted widely in the industry and going strong for 20+ years.

What Python is used for:
- Web Applications
- Web Development
- Machine Learning
- Data Science
- Cloud Infrastructure

What Javascript is used for:
- Web Applications
- Mobile Applications
- Game Development
- Web Servers
- Animation

What Java is used for:
- Web Applications
- Mobile Applications
- Game Development
- Web Servers

Programming Basics

To kick off the program, you’ll examine habits, computer basics, and fundamental programming concepts and skills necessary to be successful in your bootcamp! During this section, students study basic computer literacy skills, such as how to install and navigate basic programming tools. Students apply algorithmic thinking to make predictions of common programming skills, such as variables, arrays, conditionals, functions, and loops.

Additionally, students experience the rigor and intensity of the bootcamp, strengthening their cognitive processing stamina, resiliency, and other behavioral skills necessary for a bootcamp. By the end of the course, students should walk away with the basic computer literacy, algorithmic foundations, and learning stamina needed to find success in a bootcamp.

Up Next: Web Fundamentals
Web Fundamentals
Front-End Development & The Web

**HTML**
*Intro to HTML*
- Basic Nesting Practices, Indentation
- The Head & Body
- Body Tags (lists, tables, etc.)
- Building Forms & Declaring Input Values
- Containers, Elements, Attributes, & Classes

**CSS**
*Intro to CSS*
- CSS Selectors & Declarations
- Inspecting Element
- Inline, Block, Float, and Positioning
- Div Layout & Formatting
- Styling Text & How Fonts Work
- Using Properties & Backgrounds
- Replicating Complete User Interfaces
- Using CSS Reset & Boilerpoint

**Javascript**
- Functions & Debugging
- Event handling
- Parameters
- Implementing Dynamic Content
- Traversing DOM Elements

**jQuery**
- Essentials of the jQuery Library
- jQuery UI Library & More Libraries*

**Responsive Web Design**
*Intro to Responsive Web Design (RWD)*
- Breakpoints, Units, & Media Queries
- Basics to Typesetting & Scaling
- Cross-device RWD
- Grid System, Fluid Grids, & Adaptive Layouts

**Wireframing**
- Wireframing Fundamentals

**Git/Github**
*Git & Version Control*
- Using Terminal Commands*
- How to Create & Utilize a Repository
- Git Workflow Overview & States*

*Github*
- How to Use a Github Repository

*Optional Topics*
Python

MySQL
Intro to MySQL
- Database Design & Relationships
- Entity Relationship Diagrams (ERD)
- Database Normalization
- MySQL Workbench & Querying
- Conventions & Common Data Types
- How to Use ERDs
- Using a Database with Your UI Recreating ERDs

Python
Intro to Python
- Variables, Data Types & Best Practices
- Using Strings & Built-in String Functions
- List Creation & Manipulation
- Using Tuples & Built-in Tuple Functions
- How to Use Dictionaries in Python
- Conditionals, Operators, & Nested Loops
- Constructing Functions in Python

Python OOP
Intro to Object Oriented Programming
- Creating Objects & Classes
- Adding Properties/Attributes to Classes
- Constructing & Adding Methods to Classes
- Chaining Methods & Using Magic Methods
- How to Use Modules & Packages in Python
- Creating Multiple Objects
- Updating Methods with ‘Super’
- Overriding Inheritance & Polymorphism

Python Test Driven Development (TDD)*
- Unit Testing in Python & Outcomes
- How to Use Assertions Using
- TDD Methods: setUp & tearDown

Advanced Python*
- How to Use Multiple Arguments
- Ternary Operators in Python
- Using Lambda*
- Using Composition Over Inheritance*

Flask
Intro to Flask
- Routing in Flask Applications
- Building & Using Forms
- Rendering Templates & Views
- Delivering Static Content
- The Different HTTP Methods
- Implementing Cookies & Sessions
- Hidden Inputs & Form Validation

Flask w/ SQL
- Import, Export, & Connect Your Database
- Connecting & Running Python Across Files
- Database Communication & Validation
- Encryption & Data Security Basics

MVC
Intro to Model View Controller (MVC)
- Views, Session Classes & Session Data
- How to Use Models with Controllers
- Data Validation
- Using Bcrypt with MVC
- How to Use Multiple Controllers & Models

Deployment
- Amazon Web Services (EC2)
- Linux

*Optional Topics

Up Next: Javascript
JavaScript

JavaScript Fundamentals
- Declaring & Referencing
- Variables Variable Hoisting in JavaScript
- Conditionals, Operators, & Nested Loops
- Using Arrays & Loops in JavaScript
- Objects, Functions, & Function Scoping
- Variable Hoisting with Scoping
- Return Statements in JavaScript
- Function Hoisting

JavaScript OOP
- How to Use Object Constructors
- Common Constructors: ‘This’ & ‘New’
- Private Methods & Variables
- Creating Prototype Objects in JavaScript
- Best Practices for JavaScript OOP

Advanced JavaScript
- How to Use Callbacks
- Delegating Functionality & Event Handling

Node.js
Intro to Node
- How to Use Package Managers (NPM/Bower)
- Making a Full Web Server
- How to Work with Node Modules
- Common & Useful Node Modules
- Node.js

Modularization
- Using Require & Module.exports
- How to Modularize Existing Projects

Express.js
- HTTP Methods: Forms, Data Transfers, & Routing
- RESTful Routing

Socket.io
- Applications with Real-time Communication

MongoDB
MongoDB & Mongoose
- MongoDB Overview, CRUD Ops
- Intro to Mongoose
- Dependencies in Mongoose
- Mongoose Communication with MongoDB
- Mongoose Methods
- Data Validation with Mongoose
- Create Associations Between Mongo Objects

React
- Create React App
- Class Based Components
- Props, Children, Synthetic Events
- State, Lifecycle Methods
- Functional Components
- useState, useEffect, useReducer
- context API
- Manage application state using hooks: useState, useEffect
- useReducer, useContext

Deployment
- Amazon Web Services (EC2)
- Linux
- Production Environments

*Optional Topics
Java

Java Fundamentals
Intro to Java
- Java Development Kit Installation
- Executing Java Programs
- Variables, Data Types, & Type Casting
- Control Structures & Exceptions

Java OOP
Intro to Object Oriented Programming
- Creating Objects & Classes
- Methods, Member Variables & Constructors
- Overloading & this
- Inheritance & Packages

Advanced Java OOP
- Use of Static
- Interfaces & Abstract Classes
- Annotations
- Java Beans

Data Structures*
- Doubly Linked Lists
- Tries

Java Spring
Spring Intro
- Routing
- Java Server Pages
- Session
- Form Submission
- GET vs POST
- Dependency Injection

Spring MVC
- Model, View, and Controller (MVC) Design Pattern
- Java Persistence API (JPA)
- MySQL Connections
- Persistent Model Annotations
- Relationships
- Advanced Queries

Spring Security
- Spring Security Overview
- Authentication & Authorization
- Servlet API Integration
- Spring MVC Integration

Deployment
- Amazon Web Services (EC2)

*Optional Topics

Up Next: #CNet (optional 3 stack instead of Java)
Career Services

**Lifetime career services support.** Our experienced Career Services team provides guidance, strategy, and prep to help you in your job search whether it’s post-graduation or later down the road.

1. **Professional Profile & Portfolio Building**

   From day one, gain access to your Career Services Manager who will begin to guide you into creating your digital footprint, learning skills companies seek, and building a profile that communicates those points to recruiters. Milestones:
   - ✓ Linkedin profile creation and optimization
   - ✓ Github Portfolio Production
   - ✓ Resume Development & Curation

2. **Job Prospecting & Application Guidance**

   All while learning the most popular programs in tech, you’ll be working on your job search for when graduation approaches. Your Career Service Manager will work with you on potential job titles to seek, explain different role descriptions, and guide you on how a first job post-bootcamp can help you work toward your long-term career goals. Milestones:
   - ✓ Real Job Search
   - ✓ Sample Applications
   - ✓ Hiring Manager Communication
   - ✓ Job Title Refinement

3. **Interview Prep & Negotiation**

   One of the largest complaints by tech recruiters is it’s easy to find people who can code, perform data analysis, and can set up a Cybersecurity framework, but most of these people can’t communicate or work in teams. Whether you’re an introvert or a natural leader, our Career Services team will help you to show up as your best self in essential interviews and your day-to-day work. Milestones:
   - ✓ Mock Job Interviews
   - ✓ Technical Job Skills Tests
   - ✓ Target Compensation Management
   - ✓ Contract Negotiation

Coding Dojo cannot guarantee employment, salary or career advancement.

Up Next: Industry Trends
Industry Trends

Projected Employment Growth for Software Developers*
Between 2021-31

25%

$120,730
Median Annual Wage
for Software Developers*

How to Enroll

**Do Your Research**
- Explore our programs on our website and view other program overviews.
- Schedule a call with one of our Admissions Advisors who will talk through your future career goals and what program may best suit you.
- Attend an Open House to meet directly with our Instruction and Career Service Managers.

**Submit Application**
- Submit your application! The application process takes less than 5 minutes and does not include a technical assessment.
- Complete a quick 30-minute interview with our Admissions team.
- Receive your decision within 2-3 business days.

**Explore Financing Options**
- Our Admissions Advisors will help you explore our financing options.
- Coding Dojo offers a variety of payment options, financing partners, and partial-scholarships for those who qualify.

**Finalize Your Enrollment**
- Submit your deposit, confirm your financing, and sign your Enrollment Agreement to reserve your seat in class!
- Your Admissions Advisor will introduce you to your Student Experience Manager who will help you get ready to start bootcamp.

*Up Next: Financing Options*
Financing Options

**Installments**
Spread tuition payments out over your course with customizable installment plans.

**Third Party Financing**
Finance your bootcamp with a third party loan from a variety of vendors or source your own.

**Pay in Full**
Pay your tuition in full and get started.

Schedule a call with an Admissions Advisor to discuss which payment or financing option is right for you.

[Chat with Admissions]