CODER FOUNDRY

COURSE SYLLABUS

MASTER PROGRAMMING CLASS

LOCATIONS & INSTRUCTORS

CHARLOTTE CAMPUS

8430 University Executive Park Dr., #650
Charlotte, NC 28262
(704) 226-0979

Andrew Jensen, Director of Education
andrewjensen.coderfoundry.com
ajensen@coderfoundry.com

Antonio Raynor
antonioraynor.coderfoundry.com
araynor@coderfoundry.com

TRIAD CAMPUS

1231 Shields Rd., Suite 5
Kernersville, NC 27284
(336) 231-8632

Thomas Parrish
thomasparrish.coderfoundry.com
tparrish@coderfoundry.com

Ria Manglani
riamanglani.coderfoundry.com
rmanglani@coderfoundry.com

KNOWLEDGE PREREQUISITES

Required for Course Admittance

- Experience building HTML web pages
- Experience coding in an object-oriented language (C#, Java, C++, Objective-C, Python, Ruby, etc.)
- Basic understanding of relational databases and SQL
- Basic understanding of software development principles and application life cycles

If the applicant has deficiencies in any of the above-mentioned areas, we require completion of the corresponding tutorials:

- Microsoft Virtual Academy
  - HTML5 & CSS3 Fundamentals (link)
  - Software Development Fundamentals (link)
  - Database Fundamentals (link)
  - JavaScript Fundamentals (link)
  - C# Fundamentals for Absolute Beginners (link)
- PluralSight
  - Becoming a .NET Developer (link)
  - C# Fundamentals (link)
  - Object Oriented Fundamentals in C# (link)

All applicants must pass a programming skills and logic assessment, as well as a personal interview with our Skills Assessment Team, prior to admittance.

HARDWARE AND SOFTWARE PREREQUISITES

A personal laptop that meets the hardware requirements listed below.

- 2.0 GHz or faster processor
- 4 GB RAM or better (8 GB is recommended)
- HDMI output
- Windows 7 or newer operating system (a MacBook running Windows in a virtual environment such as BootCamp is acceptable, provided the above requirements are met)

A laptop with these specs may be purchased at Walmart, Best Buy, Amazon.com, and other retailers for as little as $340.

Required software:

- Visual Studio 2013 Community with Update 4 or 5 (link)
- SQL Server Express 2014 with Tools (link)
- Google Chrome (we prefer this browser for its on-board debugging tools)
- A Git desktop client. We prefer either GitHub (link) for Windows or GitExtensions (link).

Other requirements:

- Professional e-mail account (*thorwisheshewasme@gmail.com*, *teddybearsrule@yahoo.com*, and *onehotnerd@hotmail.com* are **not** professional email accounts)
- Professional resume (at least a draft)
- Github account (link)
- Microsoft online account (link)

COURSE DESCRIPTION

This is a fast-paced, interactive learning experience that employs a learn-by-doing theory of education. Lectures are typically short in duration, between thirty and sixty minutes – during which essential skills related to the next development project are taught. Lecture sessions are followed by extensive and intense coding sessions focused on solving real-world problems. Upon completion you will have completed in excess of 600 hours of coding during which you will have built multiple real-world projects, all of which will be made available for viewing by prospective employers on your personal website.

You will be treated like an employee of Coder Foundry during your time in the course, with project specifications, weekly project deliverables, deadlines, and accountability interviews.

Mondays are reserved for Accountability Interviews, during which you will report on the work you have completed during the previous week, your status on the current deliverable and project, and any pitfalls or difficulties you may have encountered. You will also demo your software in its current deliverable state, and may be asked to discuss various aspects of the code you have written to that point. In addition, these interviews serve as an opportunity for us to coach you on valuable interviewing skills in an effort to better prepare you for the job interviews in which you will participate upon completion of the course.

COURSE PATHWAYS

Not all students enter Coder Foundry with the same level of development skill, or with the same preparation. To better accommodate all of our students, we have incorporated two distinct curriculum pathways within our Master Class.

Upon completion of the course prep work and personal interview, the Coder Foundry staff will assess each student's overall progress and ability, and assign that student to the pathway that bests suits his or her individual situation.

Once assigned to a specific pathway, a student need not remain in that pathway for the duration of the course. Each Monday interview during the first three weeks of the course is an opportunity for Coder Foundry staff to reassess each student's progress, and thus determine whether the student's current pathway is the best fit to ensure that student's continued success.

Every student who attends Coder Foundry should understand that our primary objective is to help him or her find and maintain, through the teaching and application of marketable skills, gainful employment as a software developer.

The job market is not like an academic institution, where an inadequate, C-minus grade is considered passing. No one gets or keeps a job with a C-minus performance as a developer. It is far better to be an outstanding developer in one area (like MVC) than merely adequate in multiple areas.

API PATHWAY

This is our standard curriculum plan. It assumes each student possesses a certain level of development experience sufficient for success in the course without supplementary instruction. It includes training in HTML5, Bootstrap, CSS, JavaScript, C#, ASP.NET MVC, ASP.NET Web API, and AngularJS.

WEEK	TOPIC(S)	PROJECT(S)
1	Git, Visual Studio, Bootstrap, JavaScript/jQuery review, MVC (introduction), Microsoft Azure	Bootstrap Exercises, JavaScript Exercises, Personal Website
2	C#, MVC (authentication and authorization, view models vs. data models, Code First database development, LINQ)	MVC Project #1
3	MVC (partial views, paging, search and filtering, More LINQ)	MVC Project #1 (continued)
4	MVC (Scaffolding, More LINQ), Project Development	MVC Project #2
5	Project Development	MVC Project #2
6	Project Development	MVC Project #2
7	SQL and Stored Procedures, Web API (controllers, routing, token-based authentication with EF)	Web API/AngularJS Project #1
8	AngularJS (Modules, Controllers and Services)	Web API/AngularJS Project #1
9	AngularJS	Final Project (Web API/AngularJS Project #2)
10	Project Development	Final Project (Web API/AngularJS Project #2)
11	Project Development	Final Project (Web API/AngularJS Project #2)
12	Project Development	Final Project (Web API/AngularJS Project #2)

MVC PATHWAY

This pathway is designed for those students who lack formal computer science training or significant coding experience. This course returns to the fundamentals of computer programming theory and constructs. Student learning is focused on HTML5, Bootstrap CSS, JavaScript, C#, and ASP.NET MVC. This pathway dispenses with the Web API and AngularJS portions of the course.

WEEK	TOPIC(S)	PROJECT(S)
1	Visual Studio, Git, Programming fundamentals with C#	C# Exercises
2	HTML, Bootstrap/CSS, JavaScript (Intro)	Bootstrap Exercises, Personal Website
3	JavaScript/jQuery, Microsoft Azure	MVC Project #1
4	MVC (introduction), MVC (authentication and authorization, view models vs. data models, Code-First database development, LINQ)	MVC Project #1
5	MVC (paging, search and filtering, more LINQ)	MVC Project #1
6	MVC (scaffolding, more LINQ)	MVC Project #1
7	MVC (Review: Code-First, authentication and authorization, models, scaffolding)	MVC Project #2
8	MVC (user management, role assignment)	MVC Project #2
9	MVC (partial views)	MVC Project #2
10	Project Development	MVC Project #3 (optional)
11	Project Development	MVC Project #3 (optional)
12	Project Development	MVC Project #3 (optional)

MVC PATHWAY SUPPLEMENTARY MATERIAL

Items in this section may be assigned to individual students during the first three weeks of the course if the instructor feels the student needs additional instruction in web programming fundamentals below the level at which the curriculum is designed to operate. Current recommendations are these free online courses provided by Codecademy™:

COURSE TITLE	URL
HTML & CSS	https://www.codecademy.com/en/tracks/web
JavaScript	https://www.codecademy.com/en/tracks/javascript
jQuery	https://www.codecademy.com/en/tracks/jquery

Students may also be asked to complete or repeat the course prep work, found at the beginning of this syllabus, if their preparation is deemed insufficient.

GRADING

We follow a grading pattern very similar to that of an employer, as opposed to the types of grading used in academic settings. You will be assigned a Pass/Fail grade based upon your performance on the course projects, according to project specifications, deliverables, and deadlines. Simply put, students who complete the projects on time and as required receive a passing grade.

We actively work to help students who receive passing grades obtain employment by seeking out suitable job opportunities on their behalf, setting up job interviews, and providing interview coaching.

We cannot adequately market non-passing students to potential employers. You must have a suitable body of work that we can show potential employers before we can schedule interviews. A suitable body of work results in a passing grade for the course.

TEXTBOOKS

We pride ourselves in our ability to sit at the leading edge of software development technologies, and do not make use of printed (read: quickly outdated) textbook materials. We do make use of a variety of online reference sources, which are updated from one course to the next as required.

We also provide our students the quick reference guides, *C# 5.0 Pocket Reference* and *LINQ Pocket Reference*, both by Joseph Albahari and Ben Albahari, as their content is not as quick to change as typical language resources.

CLASS POLICIES

All Coder Foundry students are expected to conduct themselves in a professional and respectful manner at all times. As a professional technology organization, we strongly adhere to the ACM/IEEE Code of Ethics and Professional Practice, and we expect the same conduct of our students. Students who knowingly and consistently act in a manner that violates or opposes this code of conduct will be dismissed from the course. In addition, the following policies are specific to Coder Foundry classroom operations.

ATTENDANCE

Attendance is mandatory, just as it is on a job. Students who accrue more than four (4) absences for any reason may be dropped from the course. Absences, regardless of the reason, have no effect on project deadlines. A project that is incomplete or late is a failed project, regardless of the reason.

HOMEWORK

Specific homework tasks, in addition to regular project work, may be assigned during the course. Unless otherwise specified, you will be expected to arrive the following class day with the task completed. Failure to do so may result in you being considered absent for the day.

CLASSROOM BEHAVIOR

Disruptive behavior is unacceptable in any classroom and in any workplace. Students will not engage in non-class-related behavior, such as other employment activities, telephone conversations, video conferencing, online shopping, social networking, online videos or movies, video gaming or gambling, grooming, napping, or other behavior that is inappropriate to a work environment while in class. You may be asked to leave for the day and considered absent should such behavior occur. Persistent behavior of this type will result in you being dropped from the course. Respectful teasing of instructors is permitted in moderation.

PERSONAL INTEGRITY

Academic and professional integrity are of the utmost importance. Your work must be your own. While we encourage our students to assist and learn from one another, just as they would in a work environment, ultimately you are responsible for your own work. Submitting a project that is not your own work is absolutely unacceptable and will result in immediate dismissal from the course.

SPECIAL ACCOMMODATIONS

If you have specific needs as the result of any disability, inform a member of the Coder Foundry staff before beginning the course so that necessary accommodations can be made.