Teaching Introduction to Python Programming

CodeHS Professional Development Course

Details

- Course delivery method: Online delivery
- Contact Hours: Approximately 30 contact hours
- Prerequisite(s): NA
- Recommended: Best for individuals planning to teach high school computer science within a year of completing the course

Description

This course equips participants with the skills to teach computer science to middle and high school students. Participants will demonstrate their mastery of basic Python programming concepts by creating coding programs that solve different puzzles and challenges. Additionally, they will explore best practices for engaging and accessible computer science instruction. They will learn strategies for facilitating collaboration with all levels of participants in a blended computer science classroom. By the end of this course, participants will be empowered with the skills, pedagogical knowledge, and confidence to deliver a successful middle or high school Python computer science class.

Course Goals

Given the level of dependence on computers for our social, economic, and daily needs, learning a fundamental level of computer science concepts has become a basic literacy for full civic participation. Becoming fluent in a computer programming language supports participants having access to a broader range of potential employment, increases their potential compensation for work, and adds a creative outlet to their lives. Participants will understand the importance of providing access and quality computer science instruction to diverse learners. They will be prepared to teach a successful and engaging computer science course, with a strong understanding of content and how to facilitate student learning.

Course Objectives

1. Participants will recognize issues in computer science education related to access and equity, and they will explain why it is essential that children are exposed to high-quality and inclusive computer science instruction in secondary school.

Note: The online course is self-paced, so times listed are estimates for each module. Students may complete each module in more or less time than is noted.

2. Participants will design lessons and activities that introduce foundational programming topics.
3. Participants will create simple programs and projects that utilize foundational programming skills, including functions, control structures, and top-down design.
4. Participants will identify common misunderstandings of novice coders and develop best practices for teaching these concepts in their classrooms.
5. Participants will demonstrate effective debugging strategies to guide their classes in identifying mistakes and correcting their work.
6. Participants will develop their own blended classroom vision and approaches to facilitate learning and collaboration in their environment.
7. Participants will generate procedures and discuss strategies to proactively manage challenges in a blended classroom.

Programming Environment

Teachers write and run programs in the browser using the CodeHS editor.

More Information

Browse the content of this course at https://codehs.com/course/teachpython/overview

Course Breakdown (30 hours)

Module	Description
Module 1: Welcome	This introductory module provides an overview of the CodeHS professional development experience. Educators will explore key goals for the course, examine strategies for planning and pacing their teaching, and reflect on access and equity in computer science education. This module also sets the foundation for goal-setting and long-term success in teaching computer science. Assessed on: ● Short Answer
Module 2: How CodeHS Works	In this module, educators will gain familiarity with the CodeHS platform. You'll learn how to navigate the teacher and student views, explore course structure, and effectively use tools for assigning content, grading student work, and monitoring progress. Assessed on: ● Quizzes ● Exercises ● Grading and Respond (to student exercises)

Note: The online course is self-paced, so times listed are estimates for each module. Students may complete each module in more or less time than is noted.

Module 3: Planning Your First Month of Introduction to Python Programming	This module walks educators through how to effectively prepare for the first month of teaching their CodeHS course. You'll develop a personalized pacing guide, identify essential early lessons, and explore best practices for setting classroom expectations and building student engagement from day one. Assessed on: ● Quizzes ● Short essays ● Coding exercises ● Creating Resources
Module 4: Spotlight Lessons for Introduction to Python Programming	Explore key lessons that are especially impactful or foundational in the course you'll be teaching. This module highlights strategies for teaching these lessons effectively and discusses how to adapt them for different learners. Assessed on: ● Quizzes ● Short essays ● Exercises ● Unit Test
Module 5: Computer Science Pedagogy	This module explores key teaching strategies for computer science, including fostering computational thinking, managing cognitive load, and using a variety of instructional approaches. Educators will also learn how to support all learners through growth mindset practices and Universal Design for Learning (UDL). Assessed on: ● Quizzes ● Short essays ● Coding exercises ● Creating Resources ● Teaching Strategies
Module 6: Effective Debugging Strategies	Debugging is an essential skill for both students and teachers. In this module, you'll learn how to model and teach effective debugging strategies, identify common coding errors, and support students in becoming independent problem-solvers. Assessed on: ● Quizzes ● Short essays ● Providing feedback to student programs, demonstrating how to coach students through effective debugging strategies ● Debugging difficult students programs

Note: The online course is self-paced, so times listed are estimates for each module. Students may complete each module in more or less time than is noted.

Module 7: Collaboration in Computer Science	Discover how to foster collaboration in your classroom. This module covers structured group work, pair programming, peer code reviews, and strategies for creating a collaborative learning environment while addressing potential challenges that may arise. Assessed on: ● Quizzes ● Short essays ● Coding exercises ● Creating Resources ● Teaching Strategies
Module 8: Assessing Student Mastery	In this module, educators will explore a variety of assessment methods used in computer science classrooms. You'll learn how to assess coding exercises, projects, and open-ended tasks, as well as how to provide actionable feedback that supports student growth. Assessed on: ● Quizzes ● Short essays ● Assessment exercises ● Creating Resources
Module 9: Course Wrap-Up	The Course Wrap-Up module includes an end-of-course survey and a course completion badge.

Assessment and Learning Resources

- To pass the course, participants must complete 85% of the course assignments and activities.
- Participants will receive formative feedback and are afforded re-doing assignments, as needed.
- Instructors will communicate to students and grade work through the CodeHS platform on a daily basis M-F.
- The CodeHS Learning Management System includes all assigned readings, articles, videos, quizzes, coding examples, coding exercises, and other assignments in each module's content area.

Note: The online course is self-paced, so times listed are estimates for each module. Students may complete each module in more or less time than is noted.