

CS 216 Software Development

Spring 2026

Instructor Information

Instructor: Dr. Jeff Lehman (Prof. Lehman or Dr. Lehman)

Office: Science Hall 182 (Dowden Science Hall, Room 182)

Office Phone: (260) 359-4209

Text (via Google Voice): (260) 200-3842

Email: jlehman@huntington.edu

Office Hours (Help and Questions):

I am available each week during the following office hours:

Monday, Wednesday, Friday: 10:00 am – 11:00 am (Indiana Time)

Tuesday, Thursday: 2:00 pm – 3:00 pm (Indiana Time)

No appointment is needed! You are also welcome to stop by the office outside of scheduled office hours. If you stop by and I am unavailable, we can easily arrange a time that works for you.

To ensure I am available at a specific time, please use the Microsoft Bookings Page to schedule a meeting.

You may also email, text, or call anytime you have questions. I typically respond within 24 hours (usually much sooner).

Meeting Time & Location

Monday, Wednesday, and Friday

9:00–9:50 am

Science Hall 175/177

Note: There are two doors to enter the classroom: 175 (left) and 177 (right).

Course Description

Review of introductory topics including variables, input/output, selection, and iteration. Advanced programming topics including functions, strings, lists, dictionaries, object-oriented programming including inheritance, recursion, basic data structures and algorithms, file processing, graphical user interfaces, and source code management.
Prerequisite: CS111.

Textbook

[Think Python 3rd Edition](#) By Allen B. Downey
Textbook is free with print copies for purchase.

Additional Reference and Tutorial websites [W3 Schools](#) - [Python Geeks for Geeks](#) - [Python](#)

Grading

Each course component is worth a specific number of points. The percentage of total points determines the final grade. The instructor reserves the right to adjust assignments and points while maintaining overall percentage weightings.

Component	Points	Percent
Course Engagement	100	10%
Problem Sets (x10)	300	30%
Project	100	10%
Exams (#1, #2, #3, Final)	500	50%
Total	1000	100%

Grading Scale

- **A:** 93.0+
- **A-:** 90.0–92.9
- **B+:** 88.0–89.9
- **B:** 83.0–87.9
- **B-:** 80.0–82.9
- **C+:** 78.0–79.9
- **C:** 73.0–77.9
- **C-:** 70.0–72.9
- **D+:** 68.0–69.9
- **D:** 63.0–67.9
- **D-:** 60.0–62.9
- **F:** 0–59.9

Course Engagement (10%)

Active Course Engagement is vital for success in this course. Weekly readings and textbook exercises present the course material and provide the background needed to complete problem sets and prepare for exams. Students are encouraged to take notes summarizing topics covered in class and while working through textbook exercises. Throughout the semester, students will be asked to complete assigned readings and short exercises as part of the Course Engagement grade. These activities are designed to help students prepare for class and actively engage with course material. These exercises may include brief written responses, short online submissions, or in-class activities. In some cases, students may be asked to demonstrate engagement by answering questions, participating in discussions, or applying concepts during class. Completion and thoughtful participation, rather than perfection, are the primary expectations for these activities. Students are encouraged to ask questions during class sessions, utilize office hours, and send questions via email and text.

Course Engagement grades are recorded after each exam: 25 points after Exam #1, 25 points after Exam #2, 30 points after Exam #3, and 20 points after Final Exam. Scores are based on attendance, in-class work, weekly assignments, textbook exercises, communication with the instructor (such as asking questions or sharing insights), a participation self-reflection, and the instructor's observations. **Course Engagement is not a single assignment but a cumulative assessment of participation and engagement over time.**

Attendance

Class sessions enable you to engage with the material, providing opportunities to ask questions, participate in discussions, and work through assignments with instructor support.

Attendance is recorded at the beginning of class. Students are expected to attend all class sessions, arrive on time, stay actively engaged, complete all in-class assignments, refrain from using phones/computers for non-course purposes, and remain in class for the entire session (except during breaks for emergencies). Late arrivals, early departures, and taking breaks during class may be treated as partial or full absences.

Huntington University policy requires students to attend at least two-thirds (2/3) of scheduled class sessions in a face-to-face course. **Students who miss one-third or more of class meetings will automatically fail the course.** This total includes excused absences for athletics, music, other university activities, or illness. Students are responsible for all material they miss.

If you arrive late or need to leave early, use the door opposite the instructor (SH 175) workstation and do so quietly. Attending part of a session is better than missing entirely.

Do not attend class if you are sick. Your health and the health of others come first.

If you miss a class, please contact Prof. Lehman. Students who miss multiple sessions may be referred to Student Services, the Dean's Office, or the Registrar for support.

Virtual class sessions may be scheduled occasionally when the instructor has a course-related conflict. These sessions are considered regular class meetings, and attendance and participation are required.

Problem Sets (30%)

Weekly problem sets are assigned throughout the semester. These assignments are designed to help you apply and integrate the concepts discussed in the text, online resources, and class sessions. The material covered will assist students in preparing for course examinations and final project. Please ensure your work is neat, organized, and clearly labeled with your first and last name. Add your name to program comments for all code submitted. Most coding assignments will be hosted on GitHub with links submitted via Moodle.

Project (10%)

Each student will complete a final project that integrates many of the key concepts covered throughout the course, including strings, regular expressions, functions, classes, files, graphical user interfaces (GUIs), API's, and graphics. This project is an opportunity to demonstrate your understanding of these concepts by building a functional and interactive program. You are encouraged to choose a project that interests you, ensuring it incorporates the required elements while reflecting your creativity and problem-solving skills. Projects will be hosted on GitHub and may be used as part of a professional portfolio to demonstrate your skills to potential employers. Detailed project requirements, milestones, and due dates will be posted on Moodle.

Late Work Policy

Students should complete class assignments before the assigned due date and time. Unless otherwise noted, late work is accepted up to 48 hours late with a 10% penalty. Students should notify the instructor for cases of extreme hardship for which it may be possible to arrange alternate due dates. The instructor will post grades on Moodle. Please review the graded work and report any discrepancies. Students are encouraged to save returned assignments to document earned points.

Programming Tools

- Python 3 **will be used for coding assignments.**
 - **Thonny IDE** - local IDE
 - **Visual Studio Code** - local IDE
 - **GitHub** - hosting code, online IDE via code spaces
 - **Colab** - online notebook IDE
-

Academic Honesty and Plagiarism

Integrity is essential. Plagiarism includes submitting work that is not your own, copying answers, failing to cite sources, using AI when prohibited, or treating individual assignments as group work. Consequences may include a failing assignment grade, failure of the course, and referral to the Academic Dean.

Plagiarism is using another's ideas, information, or wording without proper acknowledgment. All quoted, paraphrased, or summarized material must be cited. Common knowledge is the only exception.

AI Statement

This course permits the use of AI tools to assist with learning and code development. AI tools include applications like ChatGPT, Google Gemini, Microsoft Copilot, Grammarly, and others. Students are encouraged to use these tools to understand concepts, receive feedback on writing, debug code, check for plagiarism, generate practice problems, and enhance productivity. While AI can be beneficial, it is not a substitute for mastering the material. Students must fully understand and be able to explain any code or answers they submit. Students must verify the accuracy of all AI-generated code and responses to ensure their reliability. AI solutions must adhere to assignment requirements to be accepted, and some assignments may require citations for AI assistance.

Assignments will include AI usage guidelines:

- **Red:** No AI use
- **Yellow:** Limited/cautious AI use
- **Green:** AI freely allowed

AI tools are not allowed on exams. Students who use AI in ways that contradict the instructional goals outlined in the course syllabus or rely on AI to complete assignments they are expected to do independently will be in violation of the Huntington University code of conduct. Such actions will be treated as cheating and/or plagiarism. Use these tools wisely and ethically.

Disability and Accessibility Policy

In compliance with Section 504 and the ADA, Huntington University provides reasonable accommodations.

Contact the **Academic Center for Excellence (ACE)**:

- Phone: 260-359-4290

- Email: ace@huntington.edu

Faculty will work with ACE to support eligible students.

Exams (50%)

Exams provide an opportunity to demonstrate your understanding of Python coding and programming concepts. Each exam is equally weighted. The final exam is comprehensive. Review topics and sample questions are provided.

- **Exam #1: Friday, February 6**
- **Exam #2: Wednesday, March 11**
- **Exam #3: Friday, April 10**
- **Final Exam: Wednesday, April 29 - 8:00 am**

The schedule may change slightly. Check the course website for updates.

Weekly Schedule

Week	M	W	F	Topic	Chapters
1	1/12	1/14	1/16	Basics	1, 2
2	MLK	1/21	1/23	Functions, Recursion	3, 4
3	1/26	1/28	1/30	Conditionals	5, 6
4	2/2	2/4	2/6	Exam #1	
5	2/9	2/11	2/13	Iteration	7
6	2/16	2/18	2/20	Strings, Reg. Expressions	8
7	2/23	2/25	2/27	Lists	9
8	Break	Break	Break		
9	3/9	3/11	3/13	Exam #2 , Dictionaries, Tuples	10, 11
10	3/16	3/18	3/20	Text Analysis	12
11	3/23	3/25	3/27	Files	13
12	3/30	4/1	Easter		
13	Easter	4/8	4/10	Exam #3	
14	4/13	4/15	4/17	Classes	14, 15
15	4/20	4/22	4/24		16, 17
16		4/29		Final Exam	

Last modified: Thursday, January 11, 2026
