

Syllabus: Python

Starting TBD

Instructor information

Instructor

Zafar Daud

Email

codingnrobotics@gmail.com

Office location & hours

Discover Center - Waters Edge
Class hours -Saturday's: 10-11AM
Office Hours: TBD

General information

Description

This course is designed to provide students with a comprehensive introduction to the Python programming language. The course includes topics such as computer programming basics, data types, variables, input and output, logical operators, conditional statements, lists, dictionaries, tuples, sets, string operations, loops, functions, classes and objects, and artificial intelligence. Throughout the course, students will engage in interactive activities, including coding quizzes, Google Slides activities, game-based learning, and hands-on projects such as creating a quiz and a guessing game. The course will also cover the exciting field of artificial intelligence, including hands-on activities such as "Make me Happy". By the end of the course, students will have a solid understanding of the Python programming language and its applications, as well as the ability to apply their knowledge to real-world problems.

Expectations and goals

1. **Respectful Behavior:** All students should treat each other, the instructor, and their work with respect and kindness.
2. **Effort and Persistence:** Put in their best effort, even when faced with challenges. It's okay to make mistakes and we learn more from our mistakes than our successes.
3. **Collaboration:** Students are encouraged to work together on group projects or to help each other troubleshoot coding issues.
4. **Creativity:** Be creative. There is no one right answer in programming, and your creativity is valued.
5. **Attendance and Punctuality:** I expect students to attend class regularly and be on time. If you miss a class, contact me to catch up on missed materials.
6. **Growth Mindset:** Develop a growth mindset, with effort and practice you can improve your coding skills.
7. **Enjoyment and Curiosity:** Lastly, have fun and be curious about programming. The goal is not just to learn but also to enjoy the process.

Required materials

Each student will require to bring a laptop to the class.

- Windows or Apple with Chrome Browser is required.

Course Breakdown

Lesson 1:

- We will discuss:
 - The benefits students can gain from learning computer programming.
 - What is Python? Who uses Python?
- Learn about the interface in Replit
- We will discuss what print statements and comments are
- Arithmetic operators in Python

Lesson 2:

- What are the data types in Python?
- Variables and how to create them
- Naming conventions for variables

Lesson 3:

- Input and Output
- How to get input from the user
- Casting to other data types

Lesson 4:

- Logical operators - and, or, not
- Guess the animal coding class activity
- Comparisons
 - == Equals
 - != Does not equal
 - >= Greater than or equal to
 - > Greater than
 - <= Less than or equal to
 - < Less than

Lesson 5:

- Conditional statements
 - If, if-else, if-elif

Lesson 6:

- If-statements in python

- Students will learn about the syntax for if and if-else conditional statements
- Conditional statements and input

Lesson 7:

- Students will learn about the syntax for if-elif conditional statements
- The modulo operation

Lesson 8:

- What are errors?
 - Syntax errors
 - Runtime errors
 - Logical errors
- How to fix errors?
- Detect the Python Code Errors: Syntax, Logic, and Runtime Challenge

Lesson 9:

- What are modules?
- Rock Paper Scissors activity

Quiz

Lesson 10:

- Introduction to lists:
 - Creating lists
 - Indexing
 - len() function

Lesson 11:

- Continuation of lists:
 - Using mathematical operations on lists such as addition and multiplication
 - Append
 - Remove
 - Insert

Lesson 12:

- Students will be introduced to dictionaries and discuss keys VS values
- Students will learn when to implement dictionaries and when to implement lists
- How to create a dictionary
- How to print values and keys in a dictionary
- Adding and removing values in a dictionary
- Changing existing values in a dictionary
- Length of dictionaries

Lesson 13:

- Students will be introduced to tuples
- Students will discuss the difference between tuples and lists
- How to create a tuple
- How to get the length of a tuple
- How to create a tuple with one item
- How to access items in tuples
- Nested tuples

Lesson 14:

- Students will be introduced to sets
- Students will discuss the difference between tuples, lists, and sets
- Creating a set
- Duplicate values in a set
- How to get the length of a set
- Adding and removing items in a set

Lesson 15:

- Introduction to string operations:
 - Indexing
 - index()
 - len()
 - count()
 - Slicing
 - Reversing a string
 - upper()
 - lower()
 - capitalize()
 - replace()

Quiz

Lesson 16:

- Student will be introduced to loops:
 - For loops VS while loops
- Syntax of while loops

Lesson 17:

- Students will learn about the syntax of for loops
- How to loop through a string and list
- Range function

Lesson 18:

- Guess the number Activity

Lesson 19:

- Alien game

Lesson 20:

- Introduction to functions
- How to create functions in Python
- What are parameters and arguments?

Lesson 21:

- Parking lot simulation independent coding assignment

Lesson 22:

- Continuation of functions
- Students will learn about return statements
- Useful built in python functions

Quiz

Lesson 23:

- Multiplication Coding Assignment

Lesson 24:

- Lambda functions
- Comparing regular functions to lambda functions
- Higher order functions

Lesson 25:

- map() function

Lesson 26:

- filter() function
- Comparing filter() to map()

Lesson 27:

- Students will be introduced to classes and objects - object oriented programming (OOP)

Lesson 28:

- OOP in Python
- What is `__init__()` ?

- What is the keyword “self”?
- How to create objects from classes
- How to create object methods
- Students will learn how to modify object properties

Lesson 29:

- Students will be introduced to File I/O
- They will learn:
 - The difference between 'r' and 'w' modes when opening a file
 - How to read the entire contents of a file using the read method
 - How to read the contents of a file line by line using a for loop.
 - How to write text to a file using the write method.
 - How to append text to an existing file using the open function with 'a' mode.

Quiz 5

Lesson 30:

- High Score FILE I/O Assignment

Lesson 31:

- Students will be introduced to artificial intelligence
- Guess the animal class activity
- How can we reduce the number of wrong predictions?
- What are some examples of AI that you have seen in real life?

Lesson 32:

- “Make me happy” artificial intelligence activity

Lesson 33:

- “Chatbot” artificial intelligence activity

Lesson 34:

- Introduction to pygame
- You will go through all the main steps of creating a simple pygame project including:
 - How to add colors
 - How to add shapes such as circles, triangles, rectangles, and lines
 - How to add text

Lesson 35:

- Now that students know the basics, students will be tasked to recreate a picture of a house. You will Guide students to complete this all-together step by step. Make sure to walk students through your thinking process.

Disclaimer:

While this course includes 35 lesson plans designed to provide a comprehensive introduction to Python programming, please note that the pace of our learning journey may vary. My primary focus is to ensure that students understand and internalize the core concepts effectively. As a result, there is a possibility that we may not cover all 35 lesson plans within the course's duration.

My priority is to foster a deep understanding of Python fundamentals and computational thinking, which may require spending more time on certain topics based on student needs and progress. I value quality learning over the quantity of content covered.