Fairfax Collegiate
2026 Summer Program
Intro to Python Course Syllabus
Rising Grades 7-9

Course Description

Get started with Python programming.

Practice using variables, strings, boolean logic, branching, loops, lists, functions, and classes. Write your own 2D games using the Pygame library.

Students get a beginner-friendly introduction to computer programming using Python. The class uses step-by-step lessons, guided practice, and short lab assignments to help students move confidently from basic concepts to more advanced techniques. Lessons build gradually so students can experiment, ask questions, and understand how each new skill fits into real programs.

Students create small projects throughout the course, including Mad Libs generators, Rock-Paper-Scissors simulations, Hangman, and finally a custom Tic-Tac-Toe game. Exercises are fun, fast-moving, and designed to build strong problem-solving habits. Fairfax Collegiate provides laptops for all student work.

At the end of the course, families receive digital copies of all student projects. Students leave ready to explore Python further on their own and more confident taking on future programming courses or personal coding projects.

Learning Objectives

Course Goals	**Learn Basic Programming:** Students learn basic programming concepts such as variables, loops, functions, dictionaries, and classes using the intuitive and beginner-friendly language Python. **Develop Problem Solving Skills:** Students learn to use programming as a tool to solve problems that would be difficult otherwise. **Prepare for More Advanced Programming:** Students practice basic concepts of programming that they can go on to apply to other programming languages and environments, giving them a head start in future courses.
Course	**Introduction to Programming:** Students learn about how programming works and are introduced

Topics	to the Python programming language. Students learn about basic components of Python syntax, such as key words and indentation. **Hello World:** Students are introduced with a general explanation both for how computers process information as well as how students should model it. **Variables:** Students learn to use variables, the core building blocks of code, to their advantage. **If Statements:** Students learn how to model their code through if-statements, which provide the framework for the logic intrinsic in software. **Loops:** Students practice the use of both for loops and while loops, the slightly more involved cousin of if statements when it comes to logically laying out an idea. **Functions:** Students practice the use of functions, which are used to keep code neat and to organize a coder's thoughts. **Lists:** Students learn to write lists, which store values together in a logical and meaningful way, and to iterate through the contents of lists. **Dictionaries:** Students practice the use of dictionaries, which are similar to lists except dictionaries can use anything rather than just numbers to access an element. Students also learn how to iterate through dictionaries. **Classes:** Students practice the use of classes, a type of object with both variables and methods, as well as writing the methods within classes and subclasses.

Course Schedule

Class Meeting 1	**Icebreaker and Introduction Lesson: What Is Code:** Students complete an icebreaker to get to know each other and the instructor. **Intro to Programming, Hello World, and Variables:** Students begin with a basic introduction to programming. **Mad-Libs Activity:** Students complete a Mad-Libs activity to better understand the concept of variables. **Mad-Libs Activity:** Students complete a Mad-Libs activity to better understand the concept of variables. **Data Types Overview:** Students explore the different types of variables that exist in Python. **Variable Practice:** Students practice what they have learned about variables.
Class Meeting 2	**If Statements: If, Else, and Elif:** Students learn about the various if statements in Python. **Flow Blocks:** Students explore how flow blocks work with if statements using boolean logic. **If Statements: Rock, Paper, Scissors:** Students create a simulator of Rock, Paper, Scissors using if statements.

	Loops in Programming: Students are introduced to for loops and while loops.
Class Meeting 3	**For Loops:** Students strengthen their understanding of how for loops are used in programming. **Loop Practice:** Students practice what they have learned regarding loops. **Lists:** Students explore lists and arrays in Python. **Lists:** Students explore lists and arrays in Python. **Putting It Together:** Students combine everything they have learned to accomplish various tasks.
Class Meeting 4	**Putting It Together:** Students combine everything they have learned to accomplish various tasks. **String Manipulation and Concatenation:** Students learn how to use and modify string variables in Python. **String Manipulation and Concatenation:** Students learn how to use and modify string variables in Python. **Basic String Operations:** Students learn functions that help them utilize strings more effectively when programming.
Class Meeting 5	**Dictionaries:** Students understand what dictionaries in Python are and how they can be used. **Hangman:** Students use their knowledge of dictionaries to program a Hangman game. **Functions:** Students learn what functions are and how they are coded. **Making Simple Functions:** Students create functions that accomplish a variety of goals.
Class Meeting 6	**File I/O:** Students learn how to input and output files in Python. **Practicing File I/O:** Students use file I/O to sort through files.
Class Meeting 7	**Regular Expressions:** Students learn about expressions within Python. **Regex With File I/O:** Students apply Regex to the File input/output activity with the dictionary.
Class Meeting 8	**Classes in Programming:** Students understand what classes are and how they are used in programming. **Using Classes:** Students explore how classes work in Python specifically. **Shapes Class Activity:** Students practice what they have learned about classes in an activity about shape classifcation.
Class Meeting 9	**Modules:** Students discover basic modules in Python. **Revisiting Rock Paper Scissors:** Students go back to the Rock, Paper, Scissors game they made earlier and improve it with their new knowledge. **Revisiting Rock Paper Scissors:** Students go back to the Rock, Paper, Scissors game they made earlier and improve it with their new knowledge. **Intro to Tic-Tac-Toe:** Students make a two player Tic-Tac-Toe gamae as their final project.

Class Meeting 10	**Final Project:** Students create a Tic-Tac-Toe game as their final project for the course.
	Final Project: Students create a Tic-Tac-Toe game as their final project for the course.