

*Programming Language:*  
Python

*Supported Devices:*  
Mac  
Windows  
Chromebook

*Instructional Models:*  
Direct Instruction  
Instructional Scaffolding  
Use of Learning Objectives  
Relevant Vocabulary  
Bloom's Taxonomy of  
Questions  
Project-Based Instruction  
Independent Study

*Supported Learning Models:*  
Classroom  
Flipped Classroom  
Blended  
Hybrid  
Synchronous  
Asynchronous

*Aligned With National  
CS Standards*

*Reinforces:*  
Logical thinking  
Creativity  
Persistence  
Resilience  
Communication skills  
Structural thinking  
Problem-solving

### INTRODUCTION TO PROGRAMMING WITH PYTHON

#### COURSE DESCRIPTION

Essentials Python Course features 96 lessons in coding immersion designed with instructional scaffolding to promote academic equity and success across all learning levels. In this course, students will begin with the very basic building blocks of coding (variables, functions, and decisions) and will use these skills to reach more advanced topics throughout the year.

The course is divided into three types of lessons; teaching, project, and challenge, followed by quarterly cumulative capstone projects.

- **TEACHING LESSONS** focus on introducing students to new concepts and allowing them to practice with sufficient assistance.
- **LESSON PROJECTS** allow students to apply learned skills.
- **CHALLENGE LESSONS** help develop students as programmers, allowing them to formulate solutions to problems independently.
- **CAPSTONE PROJECTS** at the end of each quarter act as a milestone for students to apply everything that they have learned in an in-depth project.

Through guided instruction, Essentials teaches students how to independently debug their code, a critical skill used by programmers. Developing this learning is made easier by the Essentials methodology of integrating these concepts gradually. By the end of the course, students will have the necessary skill sets to be comfortable coding in Python and creating projects.

### LEARNING OBJECTIVES

CodeWizardsHQ developed Essentials to support students in real-world, comprehensive learning. Students not only learn how to code but to think like programmers. Each lesson is designed to build computational thinking while learning how to code using Python.

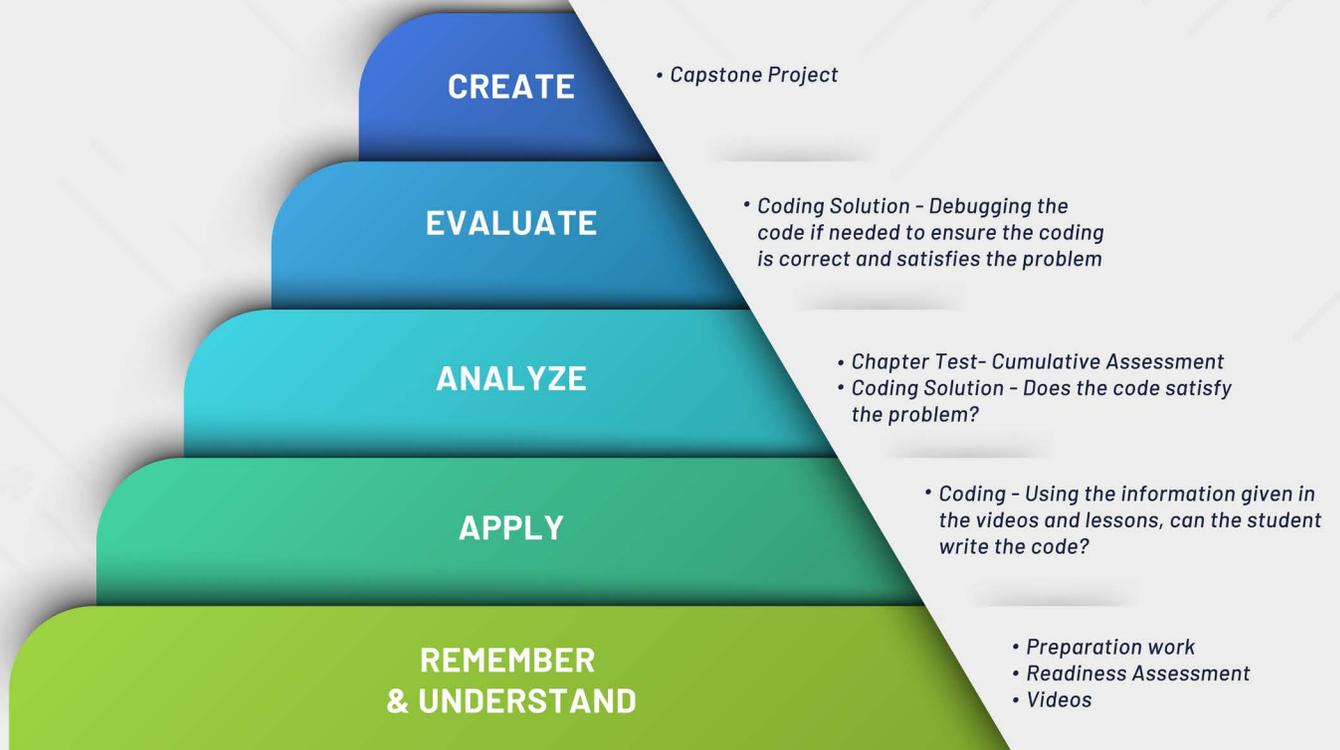
### BY THE END OF THE COURSE STUDENTS WILL BE ABLE TO:

- Differentiate between different data types in their programs
- Understand how to perform and display results of mathematical operations in code
- Understand, create, and update variables
- Be familiar with using different types of strings
- Understand conditional, elif, and greater than and less than statements
- Confidently use decisions, loops, and functions to create various projects
- Make use of the break and continue keywords
- Understand and create functions
- Understand variable scope and what is stored in a variable
- Learn to incorporate and work with parameters and learn about arguments
- Apply good user experience guidelines to create accessible projects
- Be able to read and write to an external file using Python
- Use various Python libraries

### RESOURCES INCLUDED:

- Learning Management System
- Cloud-based platform
- Administrative dashboard
- Integrated Development Environment
- Learning targets and planning tasks
- Pre-lesson activities
- Teacher/student engagement actions
- Assessing and advancing questions
- Engaging concept videos
- Proprietary slide decks
- Comprehensive assessment tools
- Auto-graded quizzes and tests
- Common errors and their solutions
- Cumulative capstone projects

## ESSENTIALS CURRICULUM IN APPLICATION OF BLOOM'S TAXONOMY



**BEGIN YOUR ESSENTIALS CODING JOURNEY TODAY!**