

# Python Introduction

Day 1 & 2

### Lesson Overview:

This lesson introduces students to the basics of Python programming, focusing on foundational concepts such as variables, loops, conditionals, and functions. By the end of the lesson, students will be able to write a simple Python program. Additionally, students will install Python and Pygame, laying the groundwork for future projects.

### Lesson Objectives:

By the end of this lesson, students will be able to:

- Write a simple Python program using variables, loops, conditionals, and functions.
- Understand and explain basic programming concepts in Python.
- Install Python and Pygame on their computers for future programming activities.

### Vocabulary:



1. **Variable:** A named storage location in memory that holds a value, which can change as the program executes.
2. **Loop:** A programming construct that repeatedly executes a block of code while a specified condition is true (e.g., `for` and `while` loops).
3. **Conditional:** A statement that allows a program to execute certain code based on whether a condition is met (e.g., `if`, `elif`, `else` statements).
4. **Function:** A reusable block of code that performs a specific task. Functions can accept parameters and may return a value.

## Lesson Plan

### Day 1: Introduction to Python Basics

#### Materials:

- **PowerPoint Presentation** ("Day 1 - Python Basics.pptx")
- **Computers** (Able to install python and IDE)



<p>PYTHON BASICS PRESENTATION</p> <div><p><b>Learn   10 min</b></p></div>	<p><b>Objective:</b> Introduce basic programming concepts using the PowerPoint.</p> <p><b>Topics Covered:</b></p> <ul style="list-style-type: none"><li>• Variables: How to declare and assign values.</li><li>• Loops: Difference between <code>for</code> and <code>while</code> loops.</li><li>• Conditionals: Using <code>if</code>, <code>elif</code>, and <code>else</code> for decision making.</li><li>• Functions: Defining and calling functions, including the use of parameters and return values.</li></ul>
<p>PRACTICE</p> <div><p><b>Apply   35 min</b></p></div>	<p><b>Hands-On Practice:</b> After covering each topic, encourage students to write a small code snippet that uses each concept. For example:</p> <ul style="list-style-type: none"><li>• Create a variable that stores the user's name.</li><li>• Write a <code>for</code> loop that prints numbers 1 to 10.</li><li>• Write an <code>if</code> statement that checks if a number is positive or negative.</li></ul>

	<ul style="list-style-type: none"> <li>• Create a simple function that greets the user.</li> </ul>
--	--

## Day 2: Applying Python Concepts

### Materials:

- **Computers** (Able to install python and IDE)

<p>PYTHON BASICS PRESENTATION</p> <div>  <p><b>Learn   15 min</b></p> </div>	<p><b>Objective:</b> Review key concepts from Day 1 through interactive discussion and code examples.</p> <p><b>Activity:</b> Ask students to recall and explain variables, loops, conditionals, and functions. Have example code snippets on the board and walk through them.</p>
<p>HANDS ON PROJECT</p> <div>  <p><b>Apply   45 min</b></p> </div>	<p><b>Hands-On Coding Exercise (45 minutes):</b></p> <ul style="list-style-type: none"> <li>• <b>Objective:</b> Have students create a small Python program that uses all the basic concepts.</li> <li>• <b>Task:</b> Students will build a simple text-based program, such as a calculator, quiz, or basic guessing game.</li> </ul>

### Additional Resources:

- [Python for Kids - GeeksforGeeks](#)