Jeide, Matthew 4/24/2025 Period 2

W16B: DE [NOTES] Unit 4 Vocabulary and Videos

Warning Must review the YouTube video to understand the basics of Python: https://www.youtube.com/watch?v=fWjsdhR3z3c

Refer to the My.PLTW.ORG for more information on **Python's Reference Guide**. Review the Statements and the Description. Locate at least three Projects that use Python code.

Python® Reference. See my.pltw.org for more information on Python coding.

Activity 4.1.2

The `while` statement	Description
`while` *[condition]*:	Identifies a section of code to repeat while the condition evaluates to true. An always-true condition, as in `while True`, will run forever.

The `if/else` statement	Description

`if btn.is_pressed:`
`  led.on()`
`else:  led.off()`
	Tests for some conditions. When the condition evaluates to true, the code below the `if` and before the `else` executes; when the condition evaluates to false, the code in the `else` executes.

Jeide, Matthew 4/24/2025 Period 2

Activity 4.1.3

Comparison operators	Description
a < b	less than: evaluates to true if a is less than b
a > b	greater than: evaluates to true if a is greater than b
a <= b	less than or equal to: evaluates to true if a is less than or equal to b
a >= b	greater than or equal to: evaluates to true if a is greater than or equal to b
a == b	equal to: evaluates to true if a is equal to b
a != b	not equal to: evaluates to true if a is not equal to b

Activity 4.1.4

The `while not` statement	Description

`while not [condition]`	Use a `not` to repeat until the condition evaluates to false (not true). For example, the following will iterate until the user presses the pi-top cancel button.
	`while not screen.cancel_button.is_pressed:`
	`...`

Comments	Description
`# a helpful comment`	A comment begins with the # character and should be used to help readers of your code understand what it is doing. The program ignores comments when it runs.

Jeide, Matthew 4/24/2025 Period 2

The `if/elif/else` statement	Description

`if a < 0:`
`  print("neg")`
`elif a > 0:  print("pos")`
`else:`
`  print("0")`
	`elif` combines an `else` and an `if` so you can test additional conditions in an `if` statement. Use as many `elif`s inside an `if` statement as you want.

Mathematical operators	Description
`a + b`	Addition; adds two values.
`a - b`	Subtraction; subtracts two values.
`a * b`	Multiplication; multiplies two values.
`a / b`	Division; divides two values.
`a // b`	Floor division; divides two values and drops the decimal portion of the returned value.
`a**b`	Exponent; raises a value to the power of another value.
a % b	Modulus; returns the remainder of a division operation.

Continue reference on my.pltw.com.

Introduction

PLTW Unit Frameworks provide an overview of the levels of understanding that each build upon the higher level: Knowledge and Skills, Objectives, Domains, and Competencies. The most fundamental level of learning is defined by course Knowledge and Skills statements. Each Knowledge and Skills statement reflects specifically what students will know and be able to do after they've had the opportunity to learn the course content. Students apply

Jeide, Matthew 4/24/2025 Period 2

Knowledge and Skills to achieve learning Objectives, which are skills that directly relate to the workplace or applied academic settings. Objectives are organized by higher-level Domains.

1. Review the YouTube video. Explain why https://www.youtube.com/watch?v=fWjsdhR3z3c coding is a good skill to acquire.

Learning to code, especially in Python, is useful because it's simple to pick up and actually lets you build things fast. The video shows how you can go from nothing to working with variables, loops, and basic logic in under 10 minutes. Python's readable, easy to test, and doesn't need a bunch of setup—just install it and start typing. It's popular too, so there are tons of resources if you ever get stuck. Overall, it's low effort to start, but high reward.

2. List three videos of projects created using Python code. Make sure these projects are relevant to engineering.

The projects may not be PLTW Projects.

	YouTube Link	Describe Project
1	https://www.youtube.com/watch?v=cAkMcPfY_Ns	From scratch (and numpy), the developer creates a neural network in Python. Neural networks are the foundation for projects in AI.
2	https://www.youtube.com/watch?v=7tXsC8YlCq8	In 48 hours, the developer who traditionally develops in C++ learns python and pygame, while simple in nature, this project showcases the versatility of python.
3	https://www.youtube.com/watch?v=Ab8TOSFfNp4	This project expands even further on the versatility of python, creating a 3D voxel engine in python (similar to Minecraft).

3. What is the difference between Python and C++?

Python is easier to write and read, but it's slower and doesn't give as much control over how your computer runs things. C++ is way faster and lets you manage memory directly, but it's harder to learn and more complex. Python is good for quick scripts and general projects, while C++ is used when speed and performance really matter, like in games or systems work.