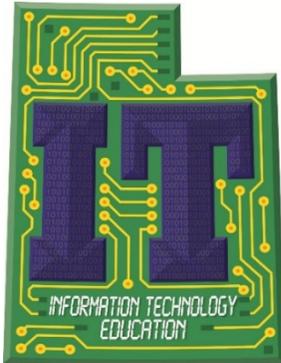


# STRANDS AND STANDARDS

## INTRODUCTION TO PYTHON 1



### Course Description

Python is a language with a simple syntax, and a powerful set of libraries. It is an interpreted language, with a rich programming environment, including a robust debugger and profiler. While it is easy for beginners to learn, it is widely used in many scientific areas for data exploration. This course is an introduction to the Python programming language for students without prior programming experience. [See Microsoft Imagine Academy for more details about the course and for added curriculum.](#)

<b>Intended Grade Level</b>	7-9
Units of Credit	0.5
Core Code	35.02.00.00.004
Concurrent Enrollment Core Code	NA
Prerequisite	Recommended: Creative Coding
Skill Certification Test Number	NA
Test Weight	NA
<b>License Type</b>	CTE and/or Secondary Education 6-12
<b>Required Endorsement(s)</b>	No Endorsement until 2020/2021
Endorsement 1	
Endorsement 2	
Endorsement 3	

## STRAND 1

### Python and Jupyter Basics

#### Standard 1

Students will be able to:

- Understand the history of programming languages.
- Understand the differences between high-level and low-level languages.
- Understand how to work with Jupyter notebooks.
- Write basic working code using the Python 3 programming language.

#### Standard 2

Students will be able to:

- Understand how variables work in Python.
- Understand a data type.
- Understand the type() built-in function.

#### Standard 3

Students will be able to:

- Write working code using type() and variables.

#### Standard 4

Students will be able to:

- Understand string/number addition and add variables.
- Understand the three main types of errors.
- Understand how to fix each type of error.

#### Standard 5

Students will be able to:

- Create on-screen art.

#### Standard 6

Students will be able to:

- Understand the input() function.
- Apply the input() and print() function in programs.

#### Standard 7

Students will be able to:

- Apply formatting to the print() function

#### Standard 8

Students will be able to:

- Understand what a Boolean value is.
- Understand string methods and how to use them.

### Standard 9

Students will be able to:

- Apply string formatting methods.
- Understand the in keyword.

### Performance Skills

- Use Python 3 in Jupyter notebooks.
- Write working code using print() and # comments.
- Write working code using type() and variables.
- Combine strings using string addition (+).
- Add numbers in code (+).

## STRAND 2

### Functions

#### Standard 1

Students will be able to:

- Understand built-in and user-defined functions.
- Create a user-defined function.
- Understand parameters and use them in functions.

#### Standard 2

Students will be able to:

- Create functions with a return value.
- Create functions with multiple parameters.

#### Standard 3

Students will be able to:

- Use knowledge of sequence in coding tasks.
- Use coding best practices.

### Performance Skills

- Create functions with a parameter.
- Create functions with a return value.
- Create functions with multiple parameters.

## STRAND 3

### Conditionals

#### Standard 1

Students will be able to:

- Control code flow with if... else conditional logic by using Boolean string methods (.isupper(), .isalpha(), startswith(...)).

**Standard 2**

Students will be able to:

- Control code flow with if...else conditional logic by using comparison operators (>, <, >=, <=, ==, !=).

**Standard 3**

Students will be able to:

- Control code flow with if...else conditional logic by using strings in comparisons.

**Standard 4**

Students will be able to:

- Code more than two choices by using elif.
- Gather numeric input by using type casting.

**Standard 5**

Students will be able to:

- Perform subtraction, multiplication, and division operations in code.

**Performance Skills**

- Control code flow with if...else conditional logic by using Boolean string methods (.isupper(), .isalpha(), .startswith(...)).
- Control code flow with if...else conditional logic by using comparison operators (>, <, >=, <=, ==, !=).
- Control code flow with if...else conditional logic by using strings in comparisons.

**STRAND 4****Nesting and Loops****Standard 1**

Students will be able to:

- Create nested conditional logic in code.

**Standard 2**

Students will be able to:

- Use escape sequences with print() statements.

**Standard 3**

Students will be able to:

- Create forever loops using while and break.
- Use incrementing variables in a while loop.

**Standard 4**

Students will be able to:

- Control while loops by using Boolean operators.