CS1010: Introduction to Programming with Python **Spring 2025**

Lecture 2: Your First Python Program

Instructor: Madhavi Vaidya TA: Syeda Faaiza Afreen

Disclaimer: *These notes aggregate content from several texts and have not been subjected to the usual scrutiny deserved by formal publications. If you find errors, please bring to the notice of the Instructor.*

2.1 Introduction to Python Programming

Python is a widely used high-level programming language, designed to be easy to read and simple to implement. Its syntax closely resembles English, making it one of the most beginner-friendly programming languages.

2.1.1 Writing Your First Python Program

To write a Python program, you need to follow these steps:

1. Open your Python IDE (e.g., IDLE, PyCharm, or any other text editor).
2. Write the following Python code:

```
# This is a comment
# The following line prints "Hello, World!" to the console
print("Hello, World!")
```

3. Save the file with a '.py' extension (e.g., 'hello˙world.py').
4. Run the program to see the output.

2.1.2 Understanding the Code

- '': This is a comment. Anything following this symbol is not executed by Python but is intended for humans reading the code. - 'print("Hello, World!")': This line is a function call to Python's built-in 'print()' function, which outputs the string '"Hello, World!"' to the console.

2.1.3 Python Interpreter and Kernel

2.1.3.1 What is a Python Interpreter?

The Python interpreter is responsible for reading and executing your Python code line by line. It translates the code into machine instructions that the computer can understand.

2.1.3.2 What is a Python Kernel?

The Python kernel is the brain behind Jupyter Notebook and similar environments. It runs your code, remembers variables, and sends back results. It keeps running in the background while you're using the notebook.

2.2 Python Programming Basics

2.2.1 Keywords

Keywords are reserved words in Python that have a predefined meaning. They cannot be used as identifiers (variable names, function names, etc.). Examples of Python keywords are:

- ‘if‘
- ‘else‘
- ‘while‘
- ‘for‘
- ‘import‘
- ‘def‘

Example:

```
if x > 0:
    print("Positive number")
else:
    print("Non-positive number")
```

2.2.2 Comments

Comments are used to explain the code and make it easier for humans to understand. Python supports two types of comments:

2.2.2.1 Single-line Comment

A single-line comment begins with the ‘‘ symbol.

Example:

```
# This is a single-line comment
x = 5  # This is an inline comment
```

2.2.2.2 Multi-line Comment

Multi-line comments are enclosed between triple quotes ‘”’‘ or ‘”””‘.

Example:

```
'''
This is a multi-line comment
It can span multiple lines
'''
```

2.2.3 Identifiers

An identifier is a name used to identify a variable, function, class, or module. It can consist of letters (a-z, A-Z), digits (0-9), and underscores (‘_‘), but cannot begin with a digit.

Example:

```
variable_name = 10  # Valid identifier
2variable_name = 10  # Invalid identifier (cannot start with a digit)
```

2.2.4 System-defined Functions

Python has several system-defined functions that are built into the language. These functions are ready to use without needing to define them. Some commonly used system-defined functions include:

- ‘print()‘: Outputs to the console.
- ‘len()‘: Returns the length of an object (like a list, string, etc.).
- ‘input()‘: Takes input from the user.
- ‘type()‘: Returns the type of an object.

Example:

```
# Using len() function
name = "Python"
print(len(name))  # Outputs: 6
```

2.2.5 Data Types

Python supports various data types, which determine the kind of value a variable can hold. Some of the most commonly used data types are:

2.2.5.1 Integer (int)

Used to represent whole numbers.

Example:

```
x = 5  # Integer
```

2.2.5.2 Float

Used to represent decimal numbers.

Example:

```
x = 3.14  # Float
```

2.2.5.3 String (str)

Used to represent text.

Example:

```
name = "Python"  # String
```

2.2.5.4 Boolean (bool)

Used to represent truth values: ‘True‘ or ‘False‘.

Example:

```
is_active = True  # Boolean
```

2.2.5.5 List

A list is an ordered collection of items.

Example:

```
fruits = ["apple", "banana", "cherry"]  # List
```

2.2.5.6 Tuple

A tuple is similar to a list but is immutable (cannot be changed).

Example:

```
coordinates = (10, 20)  # Tuple
```

2.2.5.7 Dictionary (dict)

A dictionary stores data in key-value pairs.

Example:

```
person = {"name": "John", "age": 30}  # Dictionary
```

2.2.6 Example Python Program

Here's an example Python program that demonstrates the use of keywords, comments, identifiers, system-defined functions, and data types:

```
# This is a Python program demonstrating various features

# Keywords (if, else, and print)
x = 10  # Identifier
if x > 5:
    print("x is greater than 5")  # System-defined function: print()
else:
    print("x is 5 or less")

# Data types: int, float, string, boolean
y = 3.14  # Float
name = "Python"  # String
is_active = True  # Boolean

# Using system-defined functions: len() and type()
print("Length of name:", len(name))  # Output: 6
print("Type of y:", type(y))  # Output: <class 'float'>
```