01-1 Python Basics

CSI 500

Spring 2018

Note: course material adopted loosely from :
Downey, Allen B. *Python for software design: how to think like a computer scientist*. Cambridge University Press, 2009.
http://greenteapress.com/wp/think-python/

The Python programming language

- High-level computer programming language
  - Widely used
  - Easy to learn
- Powerful, lots of add-on packages extend base functionality
  - Free
  - Supports modern object-oriented design patterns
- Yes, in fact named after a British comedy troupe of the 1970s…

What is a program?

- Sequence of statements intended to be executed by a computer
- Kinds of computer languages
    - High level (Java, Python)
    - Low level (IBM 360 assembly)
- Two major modes of interaction
    - Interpreted - Python, Java, R, LISP
    - Compiled - C/C++, FORTRAN, Ada

What is debugging?

- Syntax errors
  - Statements do not conform to the formal computer language
  - Example from algebra: 3 + / 7 2 *
- Runtime errors
  - System detects an error at run time, can’t continue
  - Example: attempt to divide by 0
- Semantic errors
  - Statements are syntactically correct, but don’t do “the right thing”
- Experimental debugging
  - Introduce features into code to assist in debugging, such as print statements

Formal and Natural Languages

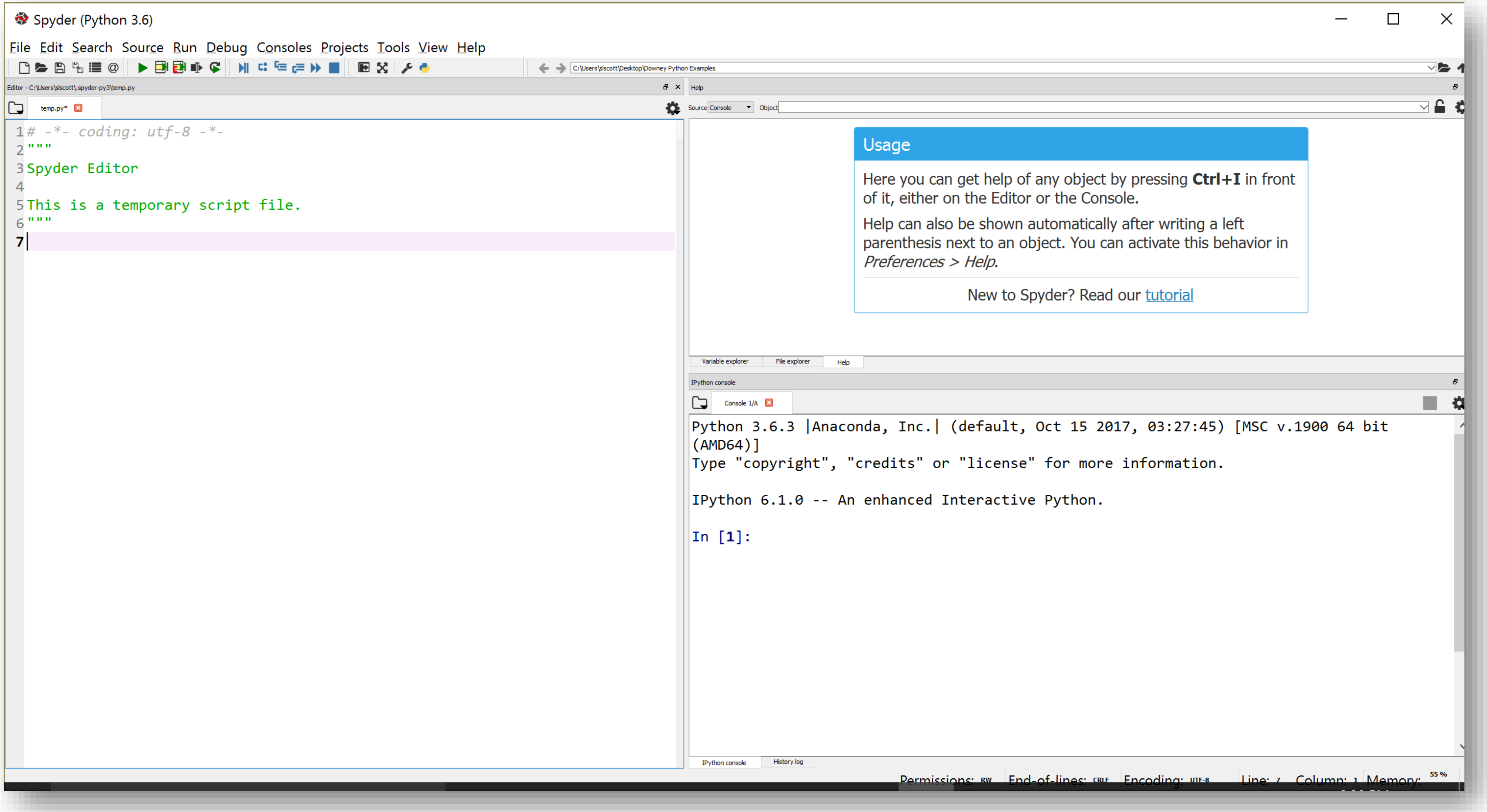
- Formal language
  - Language specifically designed for a problem-solving application
  - Examples: Java, Python, algebra, calculus
  - Syntax is strictly defined: tokens (words) and expressions (sentences) must be parsed (read) according to set of rules
- Natural language
  - Language developed by humans for communication
  - Examples: English, French, Spanish, Greek, Arabic, Farsi, Urdu, Chinese, …
  - Syntax is loosely defined: words and sentences generally follow a form, but ambiguity, redundancy, nuance allowed

The First Program

- By time honored convention, the first program written when learning a new language is “Hello World!”

- Once we get our Python environment set up, we’ll run the program twice
  - Interpreted mode: interact directly with the Python system at the prompt
  - File mode: write a program containing Python statements, and then have the Python system read our file and execute it

Open your Spyder Python IDE



```
# -*- coding: utf-8 -*-
"""
Spyder Editor

This is a temporary script file.
"""

```

Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

```
Python 3.6.3 |Anaconda, Inc.| (default, Oct 15 2017, 03:27:45) [MSC v.1900 64 bit
(AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 6.1.0 -- An enhanced Interactive Python.

In [1]:
```

Our first Python **interaction**

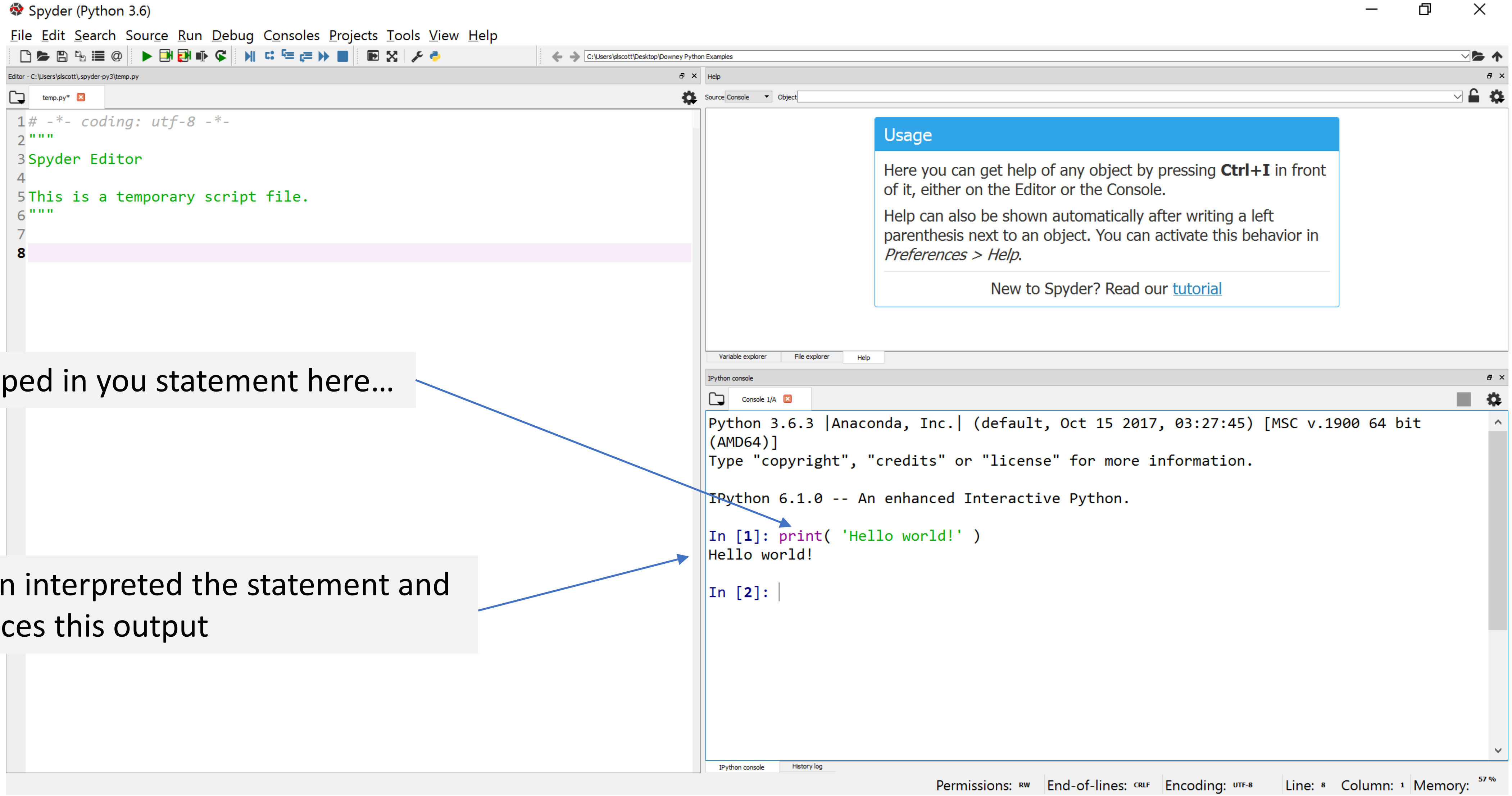
- At the **In [1]:** prompt, type in the following statement and press enter (you don't print the **In [1]:** that's the prompt)

```
In [1]: print ( 'Hello, world!' )
```

- Things to notice:
  - We used single quotes for the message string
  - By Pythonic convention, use single quotes for strings
  - Double quoted strings are allowed, but usually reserved for nested quoting such as "the university's parking lot is full"

Note: during lectures for this course, when you see the Python logo, it means you need to do something Pythonic

Here's what it should look like…



```
# -*- coding: utf-8 -*-
"""
Spyder Editor

This is a temporary script file.
"""

```

Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

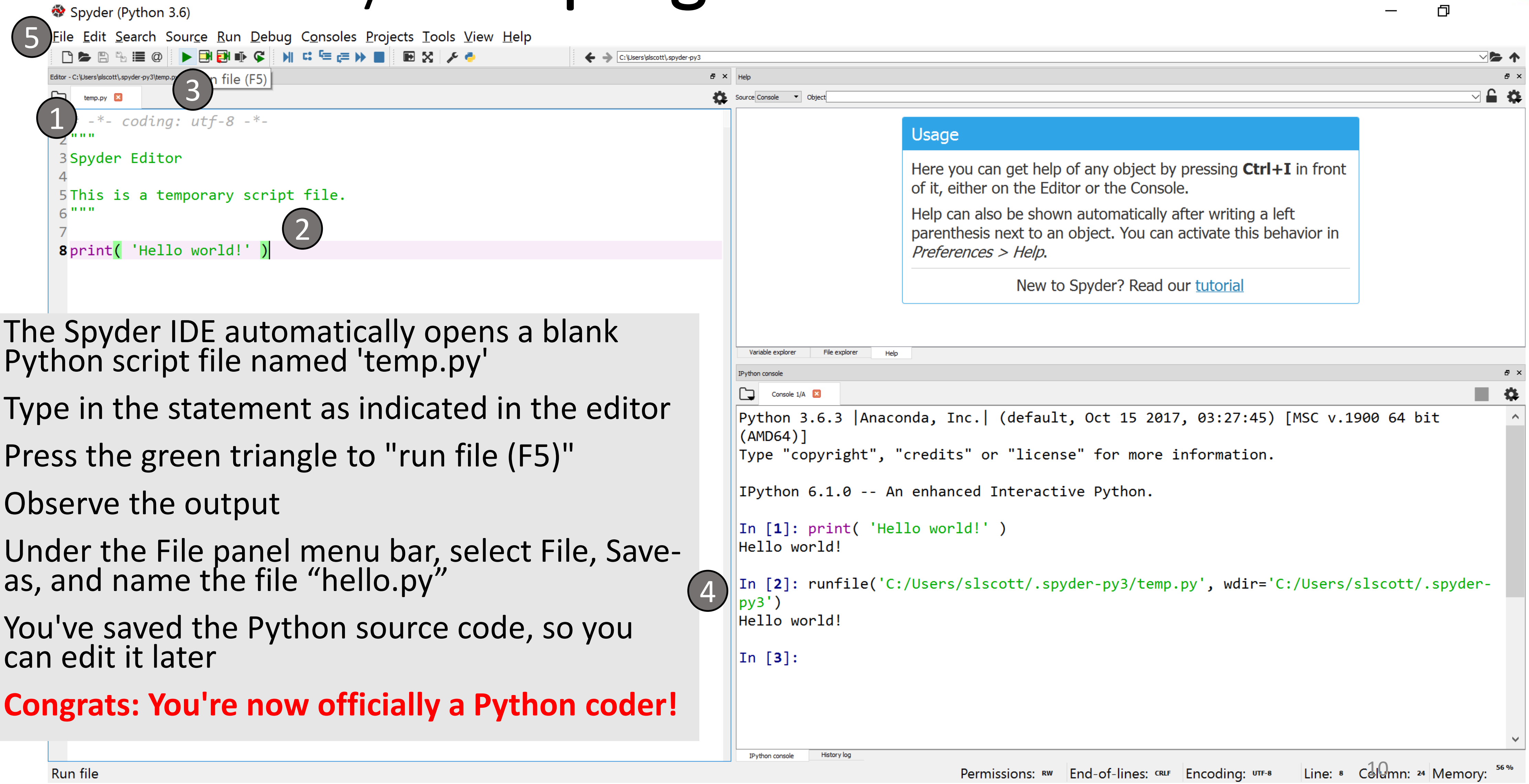
New to Spyder? Read our tutorial

```
Python 3.6.3 |Anaconda, Inc.| (default, Oct 15 2017, 03:27:45) [MSC v.1900 64 bit
(AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 6.1.0 -- An enhanced Interactive Python.

In [1]: print( 'Hello world!' )
Hello world!

In [2]:
```

You typed in you statement here…

Python interpreted the statement and produces this output

Our first Python **program**



```
# -*- coding: utf-8 -*-
"""
Spyder Editor

This is a temporary script file.
"""

print( 'Hello world!' )
```

Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

```
Python 3.6.3 |Anaconda, Inc.| (default, Oct 15 2017, 03:27:45) [MSC v.1900 64 bit
(AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 6.1.0 -- An enhanced Interactive Python.

In [1]: print( 'Hello world!' )
Hello world!

In [2]: runfile('C:/Users/slscott/.spyder-py3/temp.py', wdir='C:/Users/slscott/.spyder-
py3')
Hello world!

In [3]:
```

1. The Spyder IDE automatically opens a blank Python script file named 'temp.py'
2. Type in the statement as indicated in the editor
3. Press the green triangle to "run file (F5)"
4. Observe the output
5. Under the File panel menu bar, select File, Save-as, and name the file “hello.py”
6. You've saved the Python source code, so you can edit it later
7. **Congrats: You're now officially a Python coder!**

Python Basics

- Python is a widely used, object-oriented programming language
  - Freely available, includes wide variety of add-on packages for mathematical, scientific and research applications
- Most often used via an Integrated Development Environment (IDE)
  - manages running scripts, file system interactions
  - provides tools to assist developers