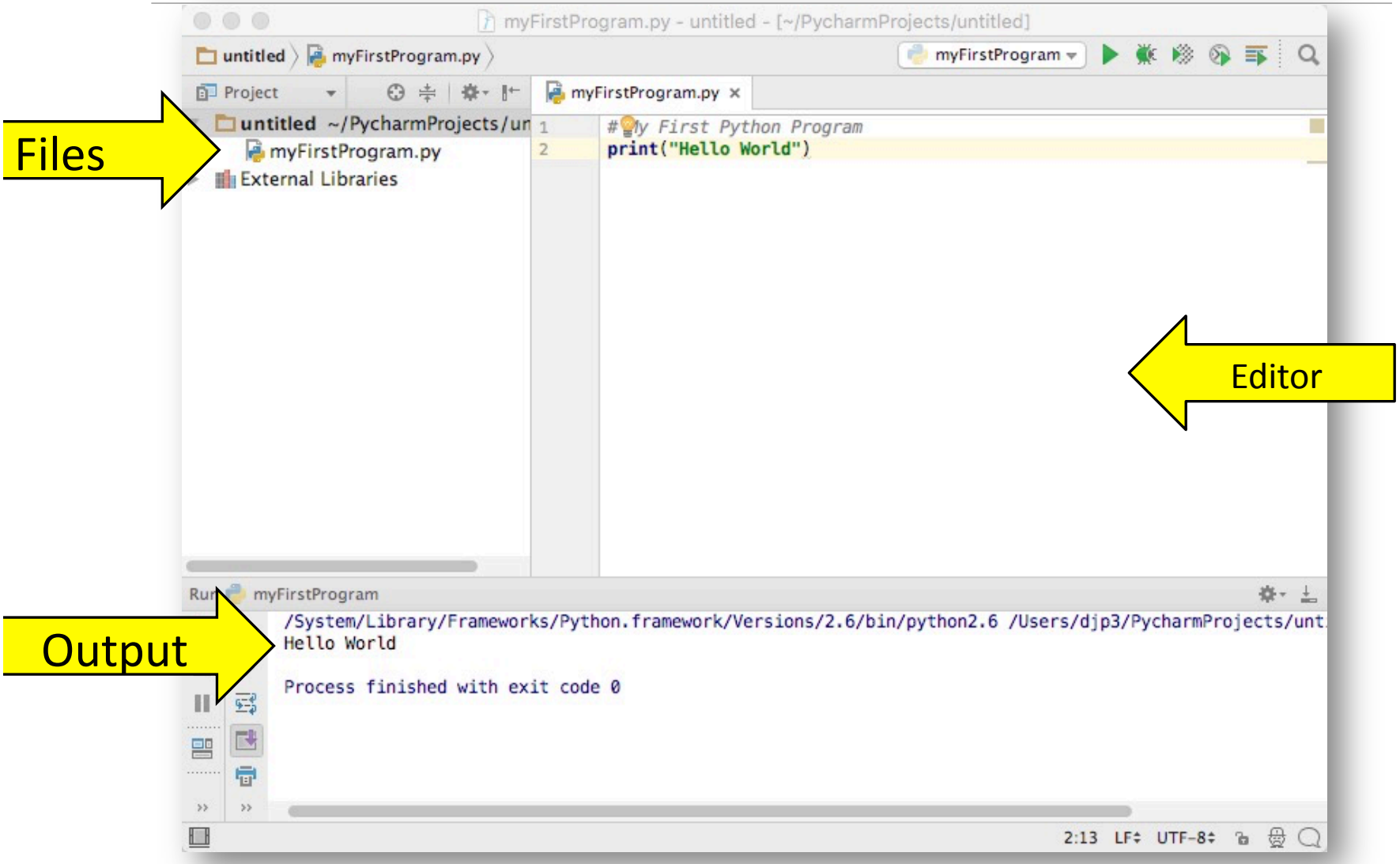
Programming Environments

- There are several ways of creating a computer program
  - Using an Integrated Development Environment (IDE)
  - Using a text editor
- You should use the method you are most comfortable with.
  - I'll PyCharm for all my in-class examples

IDE components

- The source code editor can help programming by:
  - Listing line numbers of code
  - Color lines of code (comments, text…)
  - Auto-indent source code
- Output window
- Debugger

The PyCharm IDE



Your first program

- Traditional ‘Hello World’ program in Python

```
1  # My first Python program.
2  print("Hello, World!")
3
```

- We will examine this program in the next section
  - Typing the program into your IDE would be good practice!
  - Be careful of spelling e.g., ‘print’ vs. ‘primt’
  - PyTHon iS CaSe SeNsItiVe.

Text editor programming

- You can also use a simple text editor to write your source code
- Once saved as Hello.py, you can use a console window to:
  - Compile the program
  - Run the program

Organize your work

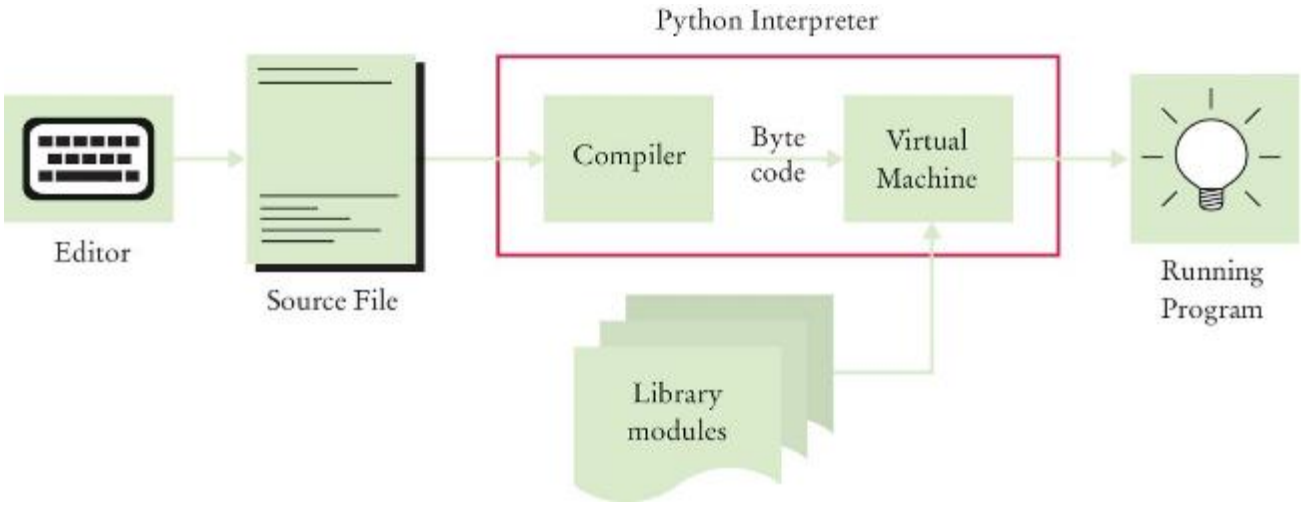
- Your ‘source code’ is stored in .py files
- Create a folder for this course
- Create one folder per program inside the course folder
  - A program can consist of several .py files
- Be sure you know where your IDE stores your files
  - You need to be able to find you files
- Backup your files:
  - To a USB flash drive
  - To a network drive

Python interactive mode

- Like other languages you can write/save a complete Python program in a file and let the interpreter execute the instructions all at once.
- Alternatively you can run instructions one at a time using interactive mode.
  - It allows quick ‘test programs’ to be written.
  - Interactive mode allows you to write python statements directly in the console window

Source Code to a Running Program

- The compiler reads your program and generates byte code instructions (simple instructions for the Python Virtual machine)
  - The Python Virtual machine is a program that is similar to the CPU of your computer
  - Any necessary libraries (e.g. for drawing graphics) are automatically located and included by the virtual machine



Let's Get Started!

- Open the PyCharm on you lab computer
- We are going to start simple, and as we learn more about Python, we'll use additional features in PyCharm

“Hello World”

- Type the following into the Editor:

```
# My first Python program
print(“Hello world!”)
```

- Save your file as “hello.py”
- This is “Step Two Write a simple program” from page 7 in your text.
- Remember – Python is **case sensitive**
  - You have to enter the upper and lower case letters exactly as this appear above

Analyzing Your First Program

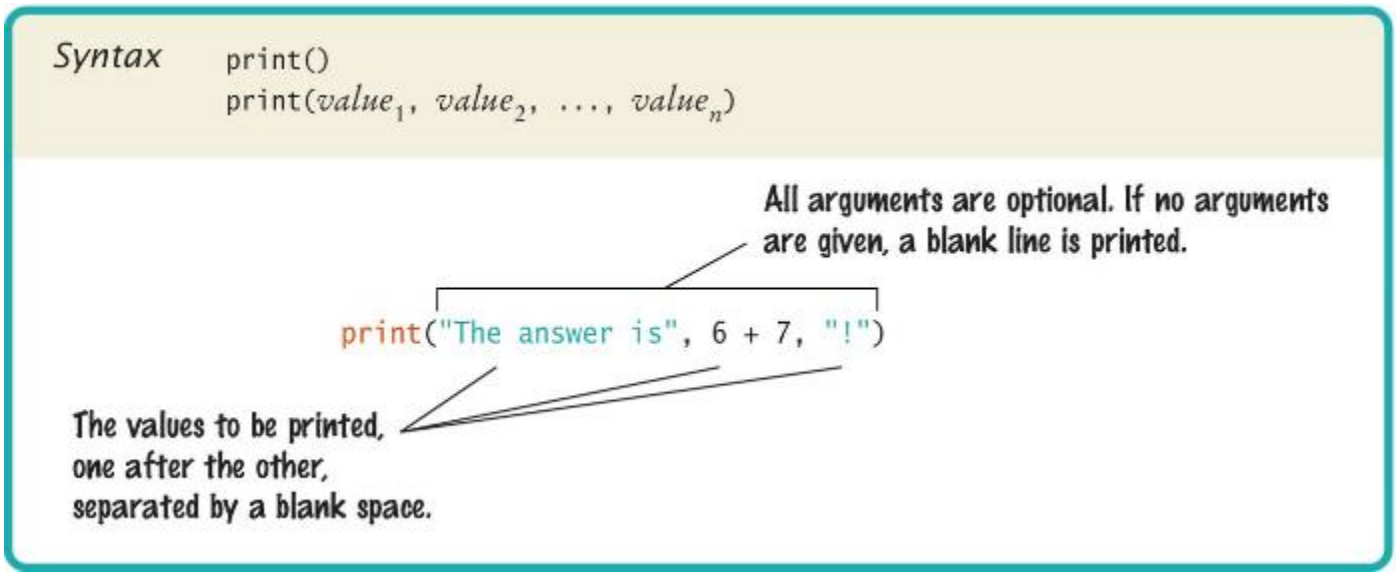
- A Python program contains one or more lines of instructions (statements) that will be translated and executed by the interpreter

```
# My first Python program
print("Hello World!")
```

- The first line is a comment (a statement that provides descriptive information about the program to programmers).
- The second line contains a statement that prints a line of text onscreen "Hello, World!"

Basic Python Syntax: *Print*

- Using the Python ‘print()’ function.
  - A function is a collection of programming instructions that carry out a particular task (in this case to print a value onscreen).
  - It’s code that somebody else wrote for you!



Syntax for Python Functions

- To use, or call, a function in Python you need to specify:
  - The name of the function that you want to use (in the previous example the name was print)
  - Any values (arguments) needed by the function to carry out its task (in this case, “Hello World!”).
  - Arguments are enclosed in parentheses and multiple arguments are separated with commas.
  - A sequence of characters enclosed in quotations marks are called a string

More Examples of the print Function

- Printing numerical values
  - print(3 + 4)
  - Evaluates the expression 3 + 4 and displays 7
- Passing multiple values to the function
  - print("the answer is", 6 * 7)
  - Displays The answer is 42
  - Each value passed to the function is displayed, one after another, with a blank space after each value
- By default the print function starts a new line after its arguments are printed
  - print("Hello")
  - print("World!")
  - Prints two lines of text
  - Hello
  - World!

Our Second Program (Page 12, printtest.py)

```
##
#  Sample Program that demonstrates the print function
#
#  Prints 7

print(3 + 4)

#  Print Hello World! on two lines
print("Hello")
print("World!")

#  Print multiple values with a single print function call
print("My favorite number are", 3 + 4, "and" 3 + 10)

#  Print Hello World! on two lines
print("Goodbye")
print()
print("Hope to see you again")
```

Errors

- There are two Categories of Errors:
  - Compile-time Errors
    - aka Syntax Errors
      - Spelling, capitalization, punctuation
      - Ordering of statements, matching of parenthesis, quotes…
    - No executable program is created by the compiler
    - Correct first error listed, then compile again.
      - Repeat until all errors are fixed
  - Run-time Errors
    - aka Logic Errors
    - The program runs, but produces unintended results
    - The program may ‘crash’

Syntax Errors

- Syntax error are caught by the compiler
- What happens if you
  - Miss-capitalize a word: Print("Hello World!")
  - Leave out quotes print(Hello World!)
  - Mismatch quotes print("Hello World!')
  - Don't match brackets print('Hello'
- Type each example above in the Wing **Python Shell** window
  - What error messages are generated?

Logic Errors

- What happens if you
  - Divide by zero print(1/0)
  - Misspell output print("Hello, Word!")
  - Forget to output Remove line 2
- Programs will compile and run
  - The output may not be as expected
- Type each example above in the PyCharm **Python Shell** window
  - What error messages are generated?

Summary: Computer Basics

- Computers rapidly execute very simple instructions
- A *Program* is a sequence of instructions and decisions
- *Programming* is the art (and science) of designing, implementing, and testing computer programs
- The Central Processing Unit (CPU) performs program control and data processing
- Storage devices include memory and secondary storage (e.g., a USB Flash Drive)

Summary: Python

- Python was designed in a way that makes it easier to learn than other programming languages such as Java, C and C++.
- The designers goal was to give Python simpler and cleaner syntax.
- Set aside some time to become familiar with the programming environment that you will use for your class work.
  - It is important to practice with the tool so you can focus on learning Python
- An editor is a program for entering and modifying text, such as a Python program.

Summary: Python

- Python is case sensitive.
  - You must be careful about distinguishing between upper and lowercase letters.
- The Python compiler translates source code into byte code instructions that are executed by the Virtual machine.
- A function is called by specifying the function's name and its parameters.
- A string is a sequence of characters enclosed in quotation marks.

Summary: Errors and pseudo code

- A compile-time error is a violation of the programming language rules that is detected by the compiler.
- A run-time error causes a program to take an action that the programmer did not intend.
- Pseudo code is an informal description of a sequence of steps for solving a problem.
- An algorithm for solving a problem is a sequence of steps that is unambiguous, executable, and terminating.

Poll Everywhere

- PollEv.com/donpatterson223