

# Introduction to Scripting Languages: Python

## Introduction to Scripting Languages: Python

ITSE-1359

**CREDIT Fall 2015**

08/24/2015 - 12/13/2015

## Course Information

### Section 002

*Distance Learning*

ONL DIL

Richard Baldwin

baldwin@austincc.edu

(512) 223.4758

### Section 003

*Distance Learning*

ONL DIL

Richard Baldwin

baldwin@austincc.edu

(512) 223.4758

### Section 004

*Distance Learning*

ONL DIL

Richard Baldwin

baldwin@austincc.edu

(512) 223.4758

## Office Hours

- Su

12:00 PM - 5:00 PM

NRG 4238 or nearby labs

All ACC students have an ACC GMail account. If you have a device that supports Gmail video chat, you can make video calls using GMail.

I will accept GMail video calls from ACC student GMail addresses whenever I am online and available, both inside and outside of scheduled office hours.

You can place a video call to me by hovering your mouse over dick.baldwin@gmail.com in your GMail chat list and clicking the camera icon below the image in the popup. If I don't answer, Email a name and GMail address and I will attempt to call you back.

A GMail video call supports screen sharing in both directions. Therefore, if you have multiple or complicated questions, a video call may be much more useful and efficient than Email and much quicker than an office visit. It may also be possible for you to initiate a video call from a cell phone or tablet.

If you don't have a webcam, you can still initiate an online voice chat with two-way screen sharing using the procedure described above.

## COURSE DESCRIPTION / RATIONALE

This syllabus applies to all sections of ITSE 1359 taught by Prof. Baldwin including:

- Regular classroom sections
- Regular Distance Learning sections
- Accelerated Programmer Training Program (APT) sections

### Important Notes

See the **Main Web Page** for this course [here](#) for additional information.

You do **NOT** need to purchase a textbook for this section of this course.

All students enrolled in this course are required to complete online orientation at the beginning of the course.

- Click [here](#) and follow the instructions to complete the orientation process.
- Then open your Blackboard course, read the Announcements, and select **Orientation Test00** in the menu on the left side of the page. You must successfully complete **Test00** with a score of at least 80 before you can view and begin working on your assignments. If you don't achieve a score of at least 80 on your first attempt, keep reading the orientation material and

- in the [Blackboard](#) menu on the left and the first three assignments should become visible. (Asg04 should be visible under Assignments-T.)
- Use the [Blackboard](#) "Send Email" feature within the first five days of the course to send an Email message to Prof. Baldwin confirming that you have completed online orientation. Make the subject of your message read "***Online orientation complete.***" If you fail to do this, you may suffer administrative penalties, which may include the loss of eligibility for financial aid.

#### Logical steps for starting and completing this course

1. Begin by reading this syllabus in its entirety paying particular attention to the section titled *Course Requirements*.
2. Access the *Assignment-XXX* and *TestXX* sections for the course on [Blackboard](#) to get more specific information regarding the *Assignments* and the *Tests*.
3. Access [Instructions for Downloading and Submitting Assignments](#) to get specific instructions for downloading and submitting assignments.
4. Access [Instructions for Accessing and Taking Blackboard Tests](#) to get specific instructions for accessing and taking Blackboard tests.
5. Go to <http://www.austincc.edu/baldwin/>. Follow the link to the *Main page* for your course, complete the online orientation, and confirm completion as described above.
6. Complete the orientation test identified as **Test00** with a score of at least 80-percent to cause the first four assignments to become visible in Blackboard.
7. Follow the links to the learning resources provided at <http://www.austincc.edu/baldwin/> in order to complete the assignments, take the tests, and successfully complete the course.

#### Course Description

Introduction to scripting languages including basic data types, control structures, regular expressions, input/output, and textual analysis.

#### Course Rationale

This course is an introduction to scripting languages and Python. The purpose of the course is to prepare students for building scripts that control a sequence of program steps such as those used in developing testing and deploying software. A modern scripting language, Python, is used as an example of a scripting language.

#### Course Structure

Students are expected complete an online orientation and signify completion of the orientation effort by successfully completing an orientation test with a grade of 80 or higher. Students are then expected to study the learning resources in the Ebook titled [ITSE 1359 Introduction to Scripting Languages: Python](#) from beginning to end. Along the way, students are expected to successfully write twelve Python programming assignments and to successfully complete four competency tests. Specifications and date deadlines for the assignments and tests are provided in Blackboard.

-end Description-

---

## STUDENT LEARNING OUTCOMES/LEARNING OBJECTIVES

**IMPORTANT:** See the link to the **Main Web Page** for this course on the **Course Description / Rationale** page for additional information about this course, including a requirement for online orientation.

#### Course Objectives/ Learning Outcomes

1. Demonstrate the basic techniques used to create scripts for automating system administrative tasks.
2. Design, code, and test Python applications using Python doctests.
3. Demonstrate the use of regular expressions in processing text.
4. Demonstrate the use of Python in developing applications using networking and databases.

#### Competencies

1. Language Fundamentals - Demonstrate the basic techniques used to create Python scripts for automating system administrative tasks.
2. Testing - Design, code, and test Python applications using Python doctests.
3. Text Processing - Demonstrate the use of regular expressions in processing text.
4. Networking and Databases - Demonstrate the use of Python in developing applications using networking and databases.

#### Scans Competencies

Use your favorite search engine to search for "Secretary's Commission on Achieving Necessary Skills" for an explanation of SCANS. The following list summarizes the SCANS competencies addressed in this particular course:

#### RESOURCES:

**INFORMATION:**

- 3.1 Acquires and Evaluates Information
- 3.2 Organizes and Maintains Information
- 3.3 Uses Computers to Process Information

**SYSTEMS**

- 4.1 Understands Systems

**TECHNOLOGY**

- 5.2 Applies Technology to Task

**BASIC SKILLS**

- 6.1 Reading

- 6.2 Listening

**THINKING SKILLS**

- 7.2 Decision Making

- 7.3 Problem Solving

- 7.4 Mental Visualization

- 7.5 Knowing How to Learn

- 7.6 Reasoning

**PERSONAL SKILLS**

- 8.1 Responsibility

- 8.2 Self-Esteem

- 8.4 Self-Management

-end of Outcomes-

---

## READINGS

**IMPORTANT:** See the link to the **Main Web Page** for this course on the **Course Description / Rationale** page for additional information about this course, including a requirement for online orientation.

**Textbook:** Students are not required to purchase a textbook for this course. The learning resources for this course are provided by Baldwin's Ebook titled [ITSE 1359 Introduction to Scripting Languages: Python](#) and other free online material referenced by that Ebook.

-end of Readings-

---

## COURSE REQUIREMENTS

**IMPORTANT:** See the link to the **Main Web Page** for this course on the **Course Description / Rationale** page for additional information about this course, including a requirement for online orientation.

**Assignment and Test Schedules**

During the Fall and Spring semesters, this course is offered during the 16-week, 12-week, and 8-week sessions. In the summer, the course is offered in the 9-week session. The information in this section of the syllabus is intended to apply to all four sessions.

You must complete and submit the following seventeen items:

- One take-home orientation test **before beginning work on the programming assignments**. This test carries the same grade value as a programming assignment. (See *Orientation Test00* in the left-side menu in [Blackboard](#).) This test is further described in the document titled [Instructions for Accessing and Taking Blackboard Tests](#) and in the **Course Description / Rationale** section of this syllabus.

*Assignments-X and Assignments-N in the left-side menu in [Blackboard](#)). Also see [Instructions for Downloading and Submitting Assignments](#).*

- **Four** Blackboard competency tests -- one for each of the four competencies. Each competency test carries the same grade value as a programming assignment. (See *Test01, Test02, Test03 and Test04 in the left-side menu in [Blackboard](#)*.) These competency tests are further described in the document titled [Instructions for Accessing and Taking Blackboard Tests](#) and in the **Course Description / Rationale** section of this syllabus.

### **Submitting assignments**

You may submit each assignment up to two times on or before the submission deadline explained below. Your highest score between the two scores for each assignment will be used to compute your final grade.

### **Taking the orientation test**

You may take *Test00* an unlimited number of times on or before the submission deadline. The submission deadline for *Test00* is the same as the submission deadline for *Test04* at the end of the semester. (*However, you need to achieve a score of at least 80 on Test00 very early in the semester to gain access to the assignments. You can continue taking it until the end of the semester in an attempt to improve your score.*) Your highest score among all the scores for *Test00* will be used to compute your final grade.

### **Taking the competency tests**

You may take competency *Test01, Test02, Test03, and Test04* up to two times each on or before their respective submission deadlines. Your highest score between the two scores for each competency test will be used to compute your final grade.

### **Submission deadlines**

The submission deadlines for these seventeen items vary depending on the session in which you are enrolled. You can find the **submission deadline** for each item by opening your course in Blackboard, selecting **Tools** from the left-side menu, and selecting **My Grades**. You can also find the **submission deadline** for each item by opening your Blackboard calendar. **It will not be possible for you to submit an item after 11:59 PM on the date shown.**

*Note that submission deadlines may fall on holidays or on other days that the ACC campuses are closed such as spring break. If so, you need to anticipate that circumstance and make appropriate arrangements in advance to avoid missing a deadline.*

Because the deadline for *Test00* is the same as the deadline for *Test04*, there are 16 unique deadlines (twelve for assignments and four for tests).

For general planning purposes approximately five weeks are allowed to complete the assignments and the test for the first competency in the 16-week session. The deadlines for the assignments and the tests for the remaining three competencies occur approximately every six days thereafter until the end of the semester. The initial startup time and the time interval between deadlines is correspondingly shorter for the 12-week, 9-week, and 8-week sessions.

You are permitted and encouraged to submit your assignments and to take your competency tests early.

### **Academic Testing Centers**

This course may require you to complete one or more online Blackboard tests in an ACC Academic Testing Center. If so, it is your responsibility to make all necessary arrangements with the testing center to complete the tests, including accessibility, hours of operation, etc. It is also your responsibility to comply with the [Testing Center Guidelines](#). (*In the event that you find the above link broken, you can search for and access testing center requirements from the [main ACC web site](#).*)

### **Classroom testing**

Students enrolled in a classroom section must complete the online competency tests during a regularly scheduled class or lab period AND must request to take the competency test during the first ten minutes of the class or lab period. Students enrolled in a classroom section are also permitted and encouraged to submit their assignments and to take their competency tests early.

### **Prerequisites**

One semester of programming or departmental approval.

### **Grade Policy**

Your grade will be based both on concepts and practical application.

### **Grading Scale**

Letter grades will be assigned as follows:

90% - 100%	A
80% - 89%	B
70% - 79%	C
60% - 69%	D
0% - 59%	F

Each of the seventeen assignments and tests listed earlier will be weighted equally when computing your final grade. Depending on the final scores of all the students taking the course, it is possible that a curve may be applied to the final grades before they are submitted for recording.

-end of Requirements-

---

## COURSE / CLASS POLICIES

**IMPORTANT:** See the link to the **Main Web Page** for this course on the **Course Description / Rationale** page for additional information about this course, including a requirement for online orientation.

**Transfers:**

Although it is technically possible for a student to transfer from one section to another section of the same course, this process has caused many problems in the past, and is not allowed unless the reasons for the transfer are compelling. Students desiring to transfer between CIS/CSC courses must first obtain permission from an Assistant Dean for CIS/CSC who will initiate the paperwork. (*Note, however, that I will allow you to informally transfer between my in-class section and my distance-learning section of the same course at any time during the semester in those semesters where both are available.*)

**Incomplete:**

Here is the official information that I have received regarding Incomplete grades:

A student may receive a temporary grade of "I" (*Incomplete*) at the end of the semester only if **ALL** the following conditions are satisfied:

1. The student is unable to complete the course during the semester due to circumstances beyond their control.
2. The student must have earned at least half of the grade points needed for a "C" by the end of the semester.
3. The request for the grade must be made in person at the instructor's office and necessary documents completed.
4. To remove an "I", the student must complete the course by two weeks before the end of the following semester. Failure to do so will result in the grade automatically reverting to an "F".

To give you an idea of the gravity of the situation, I can recall having given a student a temporary grade of "I" only once during my entire teaching career at ACC.

**Freedom of Expression Policy:**

It is expected that faculty and students will respect the views of others when expressed in classroom discussions.

**Academic Integrity:**

A student is expected to complete his or her own projects and tests. Students are responsible for observing the policy on academic integrity described in the Current ACC Student Handbook.

*"Acts prohibited by the college for which discipline may be administered include scholastic dishonesty, including but not limited to cheating on an exam or quiz, plagiarizing, and unauthorized collaboration with another in preparing outside work. Academic work submitted by students shall be the result of their own thought, research or self-expression. Academic work is defined as, but not limited to tests, quizzes, whether taken electronically or on paper; projects, either individual or group; classroom presentations, and homework".*

The penalty accessed for violations will be in accordance with the current ACC Student Handbook policy. See <http://www.austincc.edu/> for more information.

**Attendance Policy:**

The college policy states that students are expected to attend classes and will be held responsible for all material covered in class. Regular attendance helps ensure satisfactory progress towards completion of the course.

*(Students enrolled in Open Campus classes are not expected to attend class. Prof. Baldwin does not call the roll and does not maintain an official record of attendance.)*

**Withdrawal Policy:**

It is the student's responsibility to complete a Withdrawal Form in the Admissions Office if they wish to withdraw from this class. The last date to withdraw for this semester is provided in the ACC Academic calendar for the semester in which the student is enrolled.

It is not the responsibility of the instructor to withdraw students from the class even though the instructor has the prerogative to do so under various circumstances. The instructor may elect to withdraw students from the course if he notices at some point that the student has failed to successfully complete at least 50-percent of the work to date and that there is insufficient work remaining for the student to earn a final grade of at least 70-percent in the course.

A grade of "W" will be automatically assigned if the student initiates a withdrawal through the Admissions and Records office, in accordance with the requirements of that office. If the student fails to complete the work and also fails to properly withdraw, a grade of A, B, C, D, or F will be assigned in accordance with the work that was completed.

**State law regarding withdrawals:**

many colleges you attend. Apparently, students who entered college before Fall 2007 are not affected. Ask a counselor for the official ACC interpretation.

**Students with Disabilities Policy:**

"Each ACC campus offers support services for students with documented physical or psychological disabilities. Students with disabilities must request reasonable accommodations through the Office for Students with Disabilities on the campus where they expect to take the majority of their classes. Students are encouraged to make this request three weeks before the start of the semester. (Refer to the Current ACC Student Handbook)"

**Testing Center Policy (*Open Campus Sections Only*):**

Visit the ACC web site at <http://www.austincc.edu/>. Select Search, and then search for the keywords testing center.

-end Policies-

---

## COURSE SUBJECTS

**IMPORTANT:** See the link to the **Main Web Page** for this course on the **Course Description / Rationale** page for additional information about this course, including a requirement for online orientation.

**Schedule of topics for the course**

The topics that are covered in the course are shown under **Contents** at [ITSE 1359 Introduction to Scripting Languages: Python](#) and are repeated below for convenience:

- Competency - Language Fundamentals
  - The Basics
  - Control Flow
  - Classes and Objects
  - File Input-Output
  - GUI Programming
- Competency - Testing
  - Doctest
- Competency - Text Processing
  - String Operations
  - Regular Expressions
- Competency - Networking and Databases
  - Networking with HTTP
  - Dbm and Shelve Databases
  - SQLite Database

As explained in the **Requirements** section of this syllabus, the schedule of topics depends on the session in which the student is enrolled: 16-week, 12-week, 9-week, and 8-week. The **Requirements** section also shows the approximate amount of time allowed for completion of each competency.

-end of Subjects-

[Sign Up](#) (//legacy.cnx.org/new\_account) | [Login](#) ([\[Legacy Site\]\(#\) \(//legacy.cnx.org/content/m51307/1.13/?collection=col11713/1.31&legacy=true\) | \[Donate\]\(#\) \(/donate\)](//legacy.cnx.org/login_form</a>) |</div><div data-bbox=)[Find Content](#) [Search](#)or [Choose a Subject](#)

# ITSE 1359 Introduction to Scripting Languages: Python

Book by: Richard Baldwin

[!\[\]\(642aa997563f9a325b310230bb5078b7\_img.jpg\) Contents +](#) [!\[\]\(9bef82f5a53106f2ad06a2de7acf5bcf\_img.jpg\) Metadata +](#)[Back](#)[Next](#)

## Itse1359-1000-Preface

Page by: Richard Baldwin

[▶ Summary](#)

---

# Table of contents

- [Welcome](#)
- [Information about the course](#)
  - [Course description](#)
  - [Prerequisites](#)
  - [Course rationale](#)
- [Learning resources](#)
  - [General](#)
  - [Primary learning resources](#)
- [How to use this instructional material](#)
- [An editorial comment](#)
- [Reinforcing what you are learning](#)
- [Miscellaneous](#)

# Welcome

Python at Austin Community College in Austin, TX.

*(Note to blind and visually impaired students: most of the material for this course is presented in plain text format and should be accessible using an audio screen reader or a braille display. While there are a few bitmapped images that aren't accessible, that material will not be needed for assignments or included on tests.)*

# Information about the course

## Course description

As of September 2014, the description for this course reads as follows:

Introduction to scripting languages including basic data types, control structures, regular expressions, input/output, and textual analysis.

## Prerequisites

One semester of programming or departmental approval.

## Course rationale

This course is an introduction to scripting languages and Python. The purpose of the course is to prepare students for building scripts that control a sequence of program steps such as those used in developing testing and deploying software. A modern scripting language, Python, is used as an example of a scripting language.

## Learning resources

### General

I have long been a proponent of free and open educational resources for college students. This version of this course does not require students to purchase printed textbooks. Instead the learning resources for the course consist of a large number of online resources arranged in the **following major sections** :

1. Language Fundamentals - Demonstrate the basic techniques used to create scripts for automating system administrative tasks.
2. Testing - Design, code, and test Python applications using Python doctests and unit testing tools.
3. Text processing - Demonstrate the use of string operations and regular expressions in processing text.
4. Networking and Databases - Demonstrate the use of Python in developing applications using networking and databases.

## Primary learning resources

You are probably reading this document online at [cnx.org](http://cnx.org/) (<http://cnx.org/>) , otherwise known as **openstax cnx** . As of October 2014, two views of the document are available: an *openstax* view and a *legacy* view. The material is the same in both views. Only the format and the top-level organization is different.

In the *openstax* view, the group of documents is referred to as a **book** and each document is referred to as a **page** .

In the *legacy* view, the group of documents is referred to as a **collection** and each document is referred to as a **module** .

You should not think of this book or this collection as a comprehensive tutorial on Python programming. Instead, you should think of it as an *annotated guidebook* to free online resources that constitute the primary learning resources for the course.

In some cases, I will explain the material in detail and illustrate the material with working programs. In other cases, I will partially explain the material and refer you to one or more online resources for additional explanations. In some cases, I will simply refer you to one or more online resources for the complete explanation. In all cases, I will either attempt to explain the material or refer you to one or more online resources for a full or partial explanation.

For many of the modules, I will also provide a *review module* containing questions and answers to help you retain the important information provided by the module.

References to free online resources will include but will not be limited to the following websites listed in no particular order:

- [How To Think Like a Computer Scientist - Interactive Version](http://interactivepython.org/courselib/static/thinkcspy/toc.html#t-o-c)  
(<http://interactivepython.org/courselib/static/thinkcspy/toc.html#t-o-c>)
- [How to Think Like a Computer Scientist](http://openbookproject.net/thinkcs/python/english3e/)  
(<http://openbookproject.net/thinkcs/python/english3e/>)
- [Skulpt](http://www.skulpt.org/) (<http://www.skulpt.org/>)
- [Online code visualizer](http://pythontutor.com/) (<http://pythontutor.com/>)
- [learnpython.org](http://www.learnpython.org/) (<http://www.learnpython.org/>)
- [Python at Codecademy](http://www.codecademy.com/en/tracks/python) (<http://www.codecademy.com/en/tracks/python>)
- [Program Arcade Games With Python And Pygame](http://programarcadegames.com/)  
(<http://programarcadegames.com/>)

- [tutorialspoint -- Python Tutorial \(<http://www.tutorialspoint.com/python/>\)](http://www.tutorialspoint.com/python/)
- [The Python Tutorial \(<https://docs.python.org/3/tutorial/index.html>\)](https://docs.python.org/3/tutorial/index.html)
- [A Beginner's Python Tutorial \(<http://www.sthurlow.com/python/>\)](http://www.sthurlow.com/python/)
- [Another Beginner's Python Tutorial  
\(\[http://en.wikibooks.org/wiki/A\\\_Beginner%27s\\\_Python\\\_Tutorial\]\(http://en.wikibooks.org/wiki/A\_Beginner%27s\_Python\_Tutorial\)\)](http://en.wikibooks.org/wiki/A_Beginner%27s_Python_Tutorial)
- [Diving Into Python \(<http://www.diveintopython.net/toc/index.html>\)](http://www.diveintopython.net/toc/index.html)
- [Python/C API Reference Manual \(<https://docs.python.org/3/c-api/index.html#c-api-index>\)](https://docs.python.org/3/c-api/index.html#c-api-index)
- [The Python Language Reference \(<https://docs.python.org/3/reference/index.html>\)](https://docs.python.org/3/reference/index.html)
- [The Python Standard Library \(<https://docs.python.org/3/library/index.html#library-index>\)](https://docs.python.org/3/library/index.html#library-index)
- [Hands-on Python Tutorial \(<http://anh.cs.luc.edu/python/hands-on/3.1/handsonHtml/>\)](http://anh.cs.luc.edu/python/hands-on/3.1/handsonHtml/)
- [learnpython.org \(<http://www.learnpython.org/>\)](http://www.learnpython.org/)
- [Non-Programmer's Tutorial for Python 3 \(\[http://en.wikibooks.org/wiki/Non-Programmer%27s\\\_Tutorial\\\_for\\\_Python\\\_3\]\(http://en.wikibooks.org/wiki/Non-Programmer%27s\_Tutorial\_for\_Python\_3\)\)](http://en.wikibooks.org/wiki/Non-Programmer%27s_Tutorial_for_Python_3)
- [Python3 Tutorial \(\[http://www.python-course.eu/python3\\\_course.php\]\(http://www.python-course.eu/python3\_course.php\)\)](http://www.python-course.eu/python3_course.php)
- [Testing Your Code -- The Hitchhiker's Guide to Python \(<http://docs.python-guide.org/en/latest/writing/tests/>\)](http://docs.python-guide.org/en/latest/writing/tests/)
- [TkDocs \(<http://www.tkdocs.com/index.html>\)](http://www.tkdocs.com/index.html)
- [Hello Tkinter \(<http://effbot.org/ tkinterbook/ tkinter-hello-tkinter.htm>\)](http://effbot.org/ tkinterbook/ tkinter-hello-tkinter.htm)
- [The Tkinter Grid Geometry Manager \(<http://effbot.org/ tkinterbook/ grid.htm>\)](http://effbot.org/ tkinterbook/ grid.htm)
- [Python Course \(<http://www.python-course.eu/index.php>\)](http://www.python-course.eu/index.php)
- [Tkinter 8.5 reference: a GUI for Python  
\(<http://infohost.nmt.edu/tcc/help/pubs/ tkinter/ web/ index.html>\)](http://infohost.nmt.edu/tcc/help/pubs/ tkinter/ web/ index.html)
- [The Invent with Python Bookshelf \(<http://inventwithpython.com/bookshelf/>\) \(\*many online books, some free, some not free\*\)](http://inventwithpython.com/bookshelf/)
- [Python Scientific Lecture Notes \(<http://scipy-lectures.github.io/index.html>\)](http://scipy-lectures.github.io/index.html)

# How to use this instructional material

Everyone is welcome to use this material in any manner that is consistent with the [Creative Commons](http://creativecommons.org/) (<http://creativecommons.org/>) license under which it is published. However, if you are enrolled in Prof. Baldwin's section of this course at Austin Community College in Austin, TX, there are some other things that you need to know.

Although the URL may change in the future, for the Spring 2015 semester, you will find the online syllabus for the course at <http://www6.austincc.edu/directory/info.php?id=baldwin> (<http://www6.austincc.edu/directory/info.php?id=baldwin>) .

Also, although the URL may change in the future, for the Spring 2015 semester, you will find the entry to the college website for the course at <http://www.austincc.edu/baldwin/> (<http://www.austincc.edu/baldwin/>) .

You will also have access to a [Blackboard](http://www.blackboard.com/) (<http://www.blackboard.com/>) site that is specifically tailored to the course. You will find additional learning resources on that site including programming assignments and tests, assignment and test schedules, etc. The order of the programming assignments and tests mirrors the order of the topics covered in this Ebook. Therefore, you should study the material in the Ebook from start to finish in parallel with your efforts to complete the assignments and the tests.

## An editorial comment

In the opinion of this author, the Python programming language in and of itself is not a particularly interesting programming language. In fact, to a programmer accustomed to compiled, strongly-typed programming languages such as C, C++, and Java, the Python programming language seems to be a little "sloppy" and fraught with pitfalls. However, in the grand scheme of things, as of 2015, Python is an **extremely important** programming language.

The importance of Python derives not from the language itself, but from the hundreds of independent open source Python libraries that have been developed by others in such areas as image manipulation, networking, plotting and graphics, engineering and scientific programming, web development, gaming, cryptography, database, geographic information systems (GIS) , audio and music, presentation, XML processing, etc. In fact, it is hard to come up with a programming application area where someone has not already supplemented the basic Python programming language with an open-source library designed for use in that application area. Many of those libraries are Python wrappers for compiled C code providing speed and efficiency not normally associated with interpreted languages such as Python. A long list of such library modules is provided at [UsefulModules](https://wiki.python.org/moin/UsefulModules) (<https://wiki.python.org/moin/UsefulModules>) . Many of the links on that page point to other pages that also contain lists of links. An even longer list is provided on the [SciPy Topical Software](http://www.scipy.org/topical-software.html) (<http://www.scipy.org/topical-software.html>) page.

## Reinforcing what you are learning

Although not required by the course, I highly recommend that you also study the material in the section of this Ebook titled [Putting Python to Work](#) (<http://cnx.org/contents/228ac3a7-ed79-4a10-8d52-890dfa1fd20d>) in addition to the required material described [earlier](#) . Seeing how Python is actually used in practice may help you to develop and retain the knowledge required to succeed in the course.

Similarly, I also recommend that you study the material on the following three websites in parallel with the material on this website to reinforce what you **are learning** :

[Python at Codecademy](http://www.codecademy.com/en/tracks/python) (<http://www.codecademy.com/en/tracks/python>)

### Program Arcade Games With Python And Pygame

(<http://programarcadegames.com/>)

### How To Think Like a Computer Scientist - Interactive Version

(<http://interactivepython.org/courselib/static/thinkcspy/toc.html#t-o-c>)

The first website listed above provides an interactive online Python tutorial that will help you learn the fundamentals of Python programming and test your progress along the way.

The second website listed above will not only help you learn about Python programming, but may also provide you with some enjoyment along the way. This website provides both text and video instruction, along with quizzes and programming projects to teach you how to use the Python language to write interactive arcade games involving sound, graphics, etc. While much of that material is beyond the scope of this course, you may find that learning that material will make the course more enjoyable.

The third website listed above is an excellent free online interactive Python textbook designed specifically for use in courses commonly referred by computer science academics as CS-1 courses. The non-interactive version of the book is located here (<http://openbookproject.net/thinkcs/python/english3e/>).

# Miscellaneous

This section contains a variety of miscellaneous information.

---

### **Housekeeping material**

- Module name: Itse1359-1000-Preface
- File: Itse1359-1000.htm
- Published: 10/13/2014
- Revised: 03/26/15

---

### **Disclaimers:**

**Financial** : Although the Connexions site makes it possible for you to download a PDF file for this module at no charge, and also makes it possible for you to purchase a pre-printed version of the PDF file, you should be aware that some of the HTML elements in this module may not translate well into PDF.

I also want you to know that, I receive no financial compensation from the Connexions website even if you purchase the PDF version of the module.

In the past, unknown individuals have copied my modules from cnx.org, converted them to Kindle books, and placed them for sale on Amazon.com showing me as the author. I neither receive compensation for those sales nor do I know who does receive compensation. If you purchase such a book, please be aware that it is a copy of a module that is freely available on cnx.org and that it was made and published without my prior knowledge.

**Affiliation :** I am a professor of Computer Information Technology at Austin Community College in Austin, TX.

-end-

---

[Downloads +](#)    [History +](#)    [Attribution +](#)

[Back](#)

▲ [Back to Top](#) ▲

[Next](#)

[Licensing](#) (//legacy.cnx.org/help/ip) | [Terms of Use](#) ([News](/tos</a>) | <a href=) ([Contact](//openstaxcollege.org/news</a>) |<br/><a href=) ([Supported by William & Flora Hewlett Foundation, Bill & Melinda Gates Foundation, 20 Million Minds Foundation, Maxfield Foundation, Open Society Foundations, and Rice University. Powered by OpenStax CNX.](//legacy.cnx.org/aboutus/contact</a>)</p></div><div data-bbox=)

(<http://creativecommons.org>) © 1999-2015, Rice University. Except where otherwise noted, content created on this site is licensed under a Creative Commons Attribution 4.0 License.