



Philadelphia University
Faculty of Administrative & Financial Sciences
Department of Business Networking and Systems
Management
First Semester, 2014/2015

<u>Course Syllabus</u>	
Course Title: Advanced Programming (2)	Course code: 0371212
Course Level: Third Year	Course prerequisite(s) and/or co-requisite(s): Advanced Programming (1) 0371211
Lecture Time: 14:10-15:00 (Sun, Tue, Thu).	Credit hours: (3) hours

<u>Academic Staff Specifics</u>				
Name	Rank	Office Number/Location, and Office Phone Number	Office Hours	E-mail Address
Dr. Hussein H. Owaied Al-Shemery	Ph.D.	32422 / Second Building Ext: 2631	10:00-12:00 (Sun,Tue,Thu) 9:45-11:00 (Mon, Wed)	ashemery @philadelphia.edu.jo

Course Description:

Python is a real programming language that has a very simple and clean syntax that will help you to learn the basics of programming without the language "getting in the way". You will study a number of data structures and algorithms that are typically used in programming real applications, and learn to apply these in new situations. At the end of this course, you should be able to design and program solutions to problems of moderate complexity. Do not consider this a course in Python. This is a course in the fundamental aspects of programming and is designed to teach you the necessary programming and problem solving skills that you will be able to apply in the next part of the course which is about Networking Programming using Python.

This course discusses every aspect of client and server programming. And as Python begins to replace lots of favorite programming language, this course will benefit serious application developers who want a feature-rich, yet simple language, for deploying their products. Explanation to multitasking network servers using several models, including: forking, threading, and non-blocking sockets. Furthermore, the extensive lab examples demonstrate important concepts and practices, and provide a cadre of fully-functioning stand alone programs. Students may even use the provided examples as building blocks to create their own software.

Course Objectives:

- The student will demonstrate the ability to perform the following objectives using Python in an integrated development environment:
- Write well-designed programs using the Python language.
- Introduction to networking and Internet protocols programming: TCP/IP protocol architecture; user datagram protocol (UDP); standard Internet services.
- Sockets programming; client/server; peer-to-peer.
- Web server development: HTTP protocol and Web servers.
- Database programming; creating applications using MySQL database.
- System programming; create basic system applications.
- Apply Multi-threading and exception handling to their Network applications..

Course Components

- Support material (s).
- Study guide (s).
- Homework and laboratory guide (s) if (applicable).

• **Books**

Text book:

1. Fundamentals of Python: First Programs, 2nd Edition,
Authors: Kenneth Lambert,
Publisher: Course Technology.
Year of Publication: 2012
2. Foundations of Python Network Programming “The comprehensive guide to building network applications with Python”, 2nd Edition,
Authors: John Goerzen, Brandon Rhodes,
Publisher: Apress,
Year of Publication: 2010

In addition to the above, the students will be provided with handouts by the lecturer.

Teaching methods:

- Lectures.
- Discussion groups.
- Tutorials.
- Debates.
- Homework’s.
- Basic Research/Presentation.

Learning Outcomes:

• **Knowledge and understanding:**

After successfully passing this course, the student will be able to:

- Demonstrate the ability to create simple Python software.
- Demonstrate the ability to understand, use, and create functions.
- Demonstrate the ability to analyze, use, and create classes using Python.
- Demonstrate the ability to use and create simple Networking server/client software using TCP/IP sockets.
- Demonstrate the ability to understand, and create simple web server software.
- Demonstrate the ability to create applications that works with databases.
- Demonstrate the ability to understand, and create simple system software.
- Demonstrate the ability to apply Multi-threading and exception handling to their Network applications.

- **Cognitive skills (thinking and analysis).**
 - The lecturer will present the material in the text book in an interactive way that stimulates the thinking side of students.
 - Conducting the learning objectives for each module components in clear manner to insure the material is digested by the students.
- **Communication skills (personal and academic).**
 - For every lecture the last five minutes will be open for discussion. For further discussion, the students are welcome at the lecturer's office hour as appeared in first page.
 - Project Development: Groups of approximately two to three students develop projects, complete research, schedule meetings, write papers and reports, and deliver a 20-30 minute oral presentation using visual aids.
 - Group Management: Students work on group projects to practice interpersonal skills by communicating with group members, other groups, and peers outside the group.

Assessment instruments

- Short Reports and/or Presentations and/or Short Research Projects.
- Quizzes.
- Home works.
- Final examination.

<u>Allocation of Marks</u>		
Assessment Instruments	Mark	Exam Date and Day
First Examination	20	17/11/2013, Sunday
Second Examination	20	19/12/2013, Thursday
Final Examination	40	19/1/2013 – 28/1/2013
Attendance, Quizzes, Home works, and Reports and/or Research	20	
Total	100	

Documentation and academic honesty

This course is given from the textbook mentioned above. It is copyright protected. Students are encouraged to purchase this textbook from the university bookshop.

Definition of Plagiarism

Plagiarism is the unacknowledged borrowing of another writer's words or ideas.

How Can Students Avoid Plagiarism?

To avoid plagiarism, you must give credit whenever you use

- another person's idea, opinion, or theory;
- any facts, statistics, graphs, drawings—any pieces of information—that are not common knowledge;
- quotations of another person's actual spoken or written words; or
- Paraphrase of another person's spoken or written words.

If you are in doubt about whether what you are doing is inappropriate, consult your instructor. **A claim that “you didn’t know it was wrong” will not be accepted as an excuse.**

Penalty for Plagiarism

The minimum penalty for an act of plagiarism is a 0 on the assignment, homework, and project. Serious cases of plagiarism may result in failure in the course as a whole, or expulsion from the university.

Course Academic Calendar

Week	Basic and support material to be covered	Homework/reports and their due dates
(1)	Introduction to Python	Students drag and add period
(2)	Python Objects, and Numbers	
(3)	Sequences: Strings	
(4)	Sequences: Lists and Tuples	
(5)	Conditionals and Loops	
(6) First Examination	Quick review	First Examination
(7)	Files and Input/Output	
(8)	Errors and Exceptions	
(9)	Functions and Functional Programming	
(10)	Modules	Delivery of scientific research
(11) Second Examination	Quick review	Second Examination
(12)	Introduction to Socket Programming	
(13)		
(14)		
(15)	Creating Basic Python Applications	
(16) Final Examination	Comprehensive review for all the topics learned in the whole semester to resolve obstacles that may appear while studying.	Final Examination

Expected Workload:

On average students need to spend 2 hours of study and preparation for each 50-minute lecture/tutorial.

Attendance Policy:

Absence from lectures and/or tutorials shall not exceed 15%. Students who exceed the 15% limit without a medical or emergency excuse acceptable to and approved by the Dean of the relevant college/faculty shall not be allowed to take the final examination and shall receive a mark of zero for the course. If the excuse is approved by the Dean, the student shall be considered to have withdrawn from the course.