Course Code
Python (Basic)
Module 4, 2025-2026

Course Information

Instructor: Yingjie Lan
Office: PHBS Building, Room 754
Phone: 86-755-2603-3191
Email: ylan@phbs.pku.edu.cn
Office Hour: Wednesday 3pm – 5pm

Teaching Assistant:
Phone:
Email:

Classes:
Lectures: Day, Time
Venue: PHBS Building, Room

Course Website:
Py.phbs.pku.edu.cn

1. Course Description

1.1 Context

Course overview:

This course teaches python programming language with a focus on the skills for data analysis and scientific computation. It provides a philosophical understanding of programming to allow a deeper appreciation of why and how the language system works with the built-in data types of the language, and to cultivate better understanding of the big picture of libraries and better ability to build models and classes to solve problems of their own. The course is roughly divided into two parts: the first part focuses on the basic of the language, including elementary and compound data types, flow control structures, error and exception handling blocks, and object-oriented programming. In the second part, the students will then be exposed to a number of useful libraries for data scraping, analysis, and visualization, as well as for optimization. Both the fundamental concepts and their implementation are introduced for the libraries covered. By the end of the course, a term project is required with a presentation, and all project files shall be submitted for grading. After taking this course, the students should have an operating understanding of both the Python programming language and the libraries useful for their future study and research projects.

Prerequisites: Basic understanding of a computer programming language.

1.2 Textbooks and Reading Materials

As the course will cover a wide range of topics, there is no textbook that covers everything. Instead, digital slides and materials will be distributed to the students before classes.

2. Learning Outcomes

2.1 Intended Learning Outcomes

Learning Goals	**Objectives**	**Assessment (YES with details or NO)**
1. Our graduates will be effective communicators.	1.1. Our students will produce quality business and research-oriented documents.	Yes, project presentation slides
	1.2. Students are able to professionally present their ideas and also logically explain and defend their argument.	Yes, project presentation
2. Our graduates will be skilled in team work and leadership.	2.1. Students will be able to lead and participate in group for projects, discussion, and presentation.	Yes, team project
	2.2. Students will be able to apply leadership theories and related skills.	Yes, in team project
3. Our graduates will be trained in ethics.	3.1. In a case setting, students will use appropriate techniques to analyze business problems and identify the ethical aspects, provide a solution and defend it.	No
	3.2. Our students will practice ethics in the duration of the program.	No
4. Our graduates will have a global perspective.	4.1. Students will have an international exposure.	Yes, project team may involve international students
5. Our graduates will be skilled in problem-solving and critical thinking.	5.1. Our students will have a good understanding of fundamental theories in their fields.	Yes, fundamental ideas will be discussed
	5.2. Our students will be prepared to face problems in various business settings and find solutions.	No
	5.3. Our students will demonstrate competency in critical thinking.	Yes, the team project requires critical thinking

2.2 Course specific objectives

General objectives include:
1. skills for data analysis and scientific computation.
2. philosophical understanding of programming
3. working knowledge of the built-in data types of Python
4. to do object-oriented programming in Python
5. to use libraries for data scraping, analysis, and visualization, as well as for optimization

The objectives for each lecture are listed as follows:

Lecture 1: Introduction to Python
Objective: Equip students with the basic tools and understanding necessary to begin programming in Python.

Lecture 2: Building Blocks of Python
Objective: Teach students how to write reusable code blocks and perform repetitive tasks efficiently.

Lecture 3: Compound Data Types
Objective: Develop skills in handling complex data types and performing operations on them.

Lecture 4: Advanced Functions and Error Handling
Objective: Enable students to write robust programs with custom functions and proper error handling.

Lecture 5: Object-Oriented Programming and Modules
Objective: Introduce the concepts of classes and modules to promote code reuse and organization.

Lecture 6: File Handling and Data Manipulation
Objective: Teach students how to interact with various file formats for data storage and retrieval.

Lecture 7: Advanced Data Handling and Web Scraping
Objective: Expand the students' ability to automate tasks and integrate external services into their applications.

Lecture 8: Data Analysis Fundamentals
Objective: Equip students with tools for basic data analysis and manipulation to solve numerical problems.

Lecture 9: Visualization and Data Pipelines
Objective: Introduce students to data visualization and foundational machine learning techniques to prepare them for advanced studies.

2.3 Assessment/Grading Details

Assessment Components	**Weights**
Quizzes(2-3 times)	30%
Term Project	40%
Homework (1-2 times)	20%
Participation	10%
Total	**100%**

2.4 Academic Honesty and Plagiarism

It is important for a student's effort and credit to be recognized through class assessment. Credits earned for a student work due to efforts done by others are clearly unfair. Deliberate dishonesty is considered academic misconducts, which include plagiarism; cheating on assignments or examinations; engaging in unauthorized collaboration on academic work; taking, acquiring, or using test materials without faculty permission; submitting false or incomplete records of academic achievement; acting alone or in cooperation with another to falsify records or to obtain dishonestly grades, honors, awards, or professional endorsement; or altering, forging, or misusing a University academic record; or fabricating or falsifying of data, research procedures, or data analysis.

All assessments are subject to academic misconduct check. Misconduct check may include reproducing the assessment, providing a copy to another member of faculty, and/or communicate a copy of this assignment to the PHBS Discipline Committee. A suspected plagiarized document/assignment submitted to a plagiarism checking service may be kept in its database for future reference purpose.

Where violation is suspected, penalties will be implemented. The penalties for academic misconduct may include: deduction of honour points, a mark of zero on the assessment, a fail grade for the whole course, and reference of the matter to the Peking University Registrar.

AI tools requirements：
Using AI tools to complete assignments or assessments without the approval of the course instructor will be regarded as an act of academic dishonesty. Depending on the severity of the situation, penalties will be implemented in accordance with the provisions of the Peking University Graduate Student Handbook.

For more information of plagiarism, please refer to *PHBS Student Handbook*.

3. Topics, Teaching and Assessment Schedule

Lecture 1: Introduction to Python
Topics Covered: Installation of Python, understanding internal models, external views, action logic, variables, immutability, and exploring different data types.
Objective: Equip students with the basic tools and understanding necessary to begin programming in Python.

Lecture 2: Building Blocks of Python
Topics Covered: Functions, basic operators, and branching/looping constructs.
Objective: Teach students how to write reusable code blocks and perform repetitive tasks efficiently.

Lecture 3: Compound Data Types
Topics Covered: Lists, tuples, dictionaries, and introduction to lambda functions.
Objective: Develop skills in handling complex data types and performing operations on them.

Lecture 4: Advanced Functions and Error Handling
Topics Covered: Advanced usage of functions (variadic parameters), and error management with try-except blocks.
Objective: Enable students to write robust programs with custom functions and proper error handling.

Lecture 5: Object-Oriented Programming and Modules
Topics Covered: Classes and object-oriented programming, using and creating modules.

Objective: Introduce the concepts of classes and modules to promote code reuse and organization.

Lecture 6: File Handling and Data Manipulation
Topics Covered: Reading and writing to files, general file handling techniques.
Objective: Teach students how to interact with various file formats for data storage and retrieval.

Lecture 7: Advanced Data Handling and Web Scraping
Topics Covered: Automating file downloads, making web scrapers, extracting batches of data.
Objective: Expand the students' ability to automate tasks and integrate external services into their applications.

Lecture 8: Data Analysis Fundamentals
Topics Covered: Introduction to NumPy for numerical operations, regression analysis basics, and Pandas for data manipulation.
Objective: Equip students with tools for basic data analysis and manipulation to solve numerical problems.

Lecture 9: Visualization and Data Pipelines
Topics Covered: Data visualization with plot libraries, organizing data processing into pipelines.
Objective: Introduce students to data visualization and foundational machine learning techniques to prepare them for advanced studies.

4. Miscellaneous