1. Introduction to Python

- 1.1 Programming (general)
- 1.2 Programming using Python
- 1.3 Basic input and output
- 1.4 Errors
- 1.5 Development environment
- 1.6 Computers and programs (general)
- 1.7 Computer tour
- 1.8 Language history
- 1.9 Why whitespace matters
- 1.10 Python example: Salary calculation
- 1.11 Additional practice: Output art
- 1.12 zyLab training: Basics Lab
- 1.13 zyLab training: Interleaved input / output Lab
- 1.14 LAB: Formatted output: Hello World! Lab
- 1.15 LAB: Formatted output: No parking sign Lab
- 1.16 LAB: Input: Welcome message Lab
- 1.17 LAB: Input: Mad Lib Lab
- 1.18 zyLab training*: One large program Lab

2. Variables and Expressions

- 2.1 Variables and assignments
- 2.2 Identifiers
- 2.3 Objects
- 2.4 Numeric types: Floating-point
- 2.5 Arithmetic expressions
- 2.6 Python expressions
- 2.7 Division and modulo
- 2.8 Module basics
- 2.9 Math module
- 2.10 Random numbers
- 2.11 Representing text
- 2.12 Additional practice: Number games
- 2.13 LAB: Divide input integers Lab
- 2.14 LAB: Driving costs Lab
- 2.15 LAB: Expression for calories burned during workout Lab
- 2.16 LAB: Using math functions Lab
- 2.17 LAB: Musical note frequencies Lab
- 2.18 LAB: Convert to dollars Lab

3. Types

- 3.1 String basics
- 3.2 List basics
- 3.3 Tuple basics
- 3.4 Set basics
- 3.5 Dictionary basics
- 3.6 Common data types summary
- 3.7 Additional practice: Grade calculation
- 3.8 Type conversions
- 3.9 Binary numbers
- 3.10 String formatting
- 3.11 Additional practice: Health data
- 3.12 LAB: Input and formatted output: Right-facing arrow Lab
- 3.13 LAB: Phone number breakdown Lab
- 3.14 LAB: Input and formatted output: Caffeine levels Lab
- 3.15 LAB: Input and formatted output: House real estate summary Lab
- 3.16 LAB: Simple statistics Lab
- 3.17 LAB: List basics Lab
- 3.18 LAB: Set basics Lab

4. Branching

- 4.1 If-else branches (general)
- 4.2 Detecting equal values with branches
- 4.3 Detecting ranges with branches (general)
- 4.4 Detecting ranges with branches
- 4.5 Detecting ranges using logical operators
- 4.6 Detecting ranges with gaps
- 4.7 Detecting multiple features with branches
- 4.8 Comparing data types and common errors
- 4.9 Membership and identity operators
- 4.10 Order of evaluation
- 4.11 Code blocks and indentation
- 4.12 Conditional expressions
- 4.13 Additional practice: Tweet decoder
- 4.14 LAB: Remove gray from RGB Lab
- 4.15 LAB: Smallest number Lab
- 4.16 LAB: Interstate highway numbers Lab
- 4.17 LAB: Seasons Lab
- 4.18 LAB: Exact change Lab
- 4.19 LAB: Leap year Lab
- 4.20 LAB: Golf scores Lab

5. Loops

- 5.1 Loops (general)
- 5.2 While loops
- 5.3 More while examples
- 5.4 Counting
- 5.5 For loops
- 5.6 Counting using the range() function
- 5.7 While vs. for loops
- 5.8 Nested loops
- 5.9 Developing programs incrementally
- 5.10 Break and continue
- 5.11 Loop else
- 5.12 Getting both index and value when looping: enumerate()
- 5.13 Additional practice: Dice statistics
- 5.14 LAB: Convert to reverse binary Lab
- 5.15 LAB: Password modifier Lab
- 5.16 LAB: Output range with increment of 5 Lab
- 5.17 LAB: Print string in reverse Lab
- 5.18 LAB: Brute force equation solver Lab
- 5.19 LAB: Adjust values in a list by normalizing Lab

6. Functions

- 6.1 User-defined function basics
- 6.2 Print functions
- 6.3 Dynamic typing
- 6.4 Reasons for defining functions
- 6.5 Writing mathematical functions
- 6.6 Function stubs
- 6.7 Functions with branches/loops
- 6.8 Functions are objects
- 6.9 Functions: Common errors
- 6.10 Scope of variables and functions
- 6.11 Namespaces and scope resolution
- 6.12 Function arguments
- 6.13 Keyword arguments and default parameter values
- 6.14 Arbitrary argument lists
- 6.15 Multiple function outputs
- 6.16 Help! Using docstrings to document functions
- 6.17 Engineering examples
- 6.18 Lab training: Unit tests to evaluate your program Lab

7. Strings

- 7.1 String slicing
- 7.2 Advanced string formatting
- 7.3 String methods
- 7.4 Splitting and joining strings
- 7.5 LAB: Checker for integer string Lab
- 7.6 LAB: Name format Lab
- 7.7 LAB: Count characters Lab
- 7.8 LAB: Mad Lib - loops Lab
- 7.9 LAB: Palindrome Lab
- 7.10 LAB: Acronyms Lab
- 7.11 LAB: Remove all non-alpha characters Lab

8. Lists and Dictionaries

- 8.1 Lists
- 8.2 List methods
- 8.3 Iterating over a list
- 8.4 List games
- 8.5 List nesting
- 8.6 List slicing
- 8.7 Loops modifying lists
- 8.8 List comprehensions
- 8.9 Sorting lists
- 8.10 Command-line arguments
- 8.11 Additional practice: Engineering examples
- 8.12 Dictionaries
- 8.13 Dictionary methods
- 8.14 Iterating over a dictionary
- 8.15 Dictionary nesting
- 8.16 LAB: Varied amount of input data Lab
- 8.17 LAB: Filter and sort a list Lab
- 8.18 LAB: Elements in a range Lab
- 8.19 LAB: Contact list Lab
- 8.20 LAB: Car wash Lab
- 8.21 LAB: Word frequencies (dictionaries) Lab
- 8.22 LAB: Check if list is sorted Lab

9. Classes

- 9.1 Classes: Introduction
- 9.2 Classes: Grouping data
- 9.3 Instance methods
- 9.4 Class and instance object types
- 9.5 Class example: Seat reservation system
- 9.6 Class constructors
- 9.7 Class interfaces
- 9.8 Class customization
- 9.9 More operator overloading: Classes as numeric types
- 9.10 Memory allocation and garbage collection
- 9.11 LAB: Car value (classes) Lab
- 9.12 LAB: Winning team (classes) Lab
- 9.13 LAB: Nutritional information (classes/constructors) Hidden Lab
- 9.14 LAB: Artwork label (classes/constructors) Hidden Lab
- 9.15 LAB: Triangle area comparison (classes) Hidden Lab
- 9.16 LAB: Vending machine Hidden Lab

10. Inheritance

- 10.1 Derived classes
- 10.2 Accessing base class attributes
- 10.3 Overriding class methods
- 10.4 Is-a versus has-a relationships
- 10.5 Mixin classes and multiple inheritance
- 10.6 Testing your code: The unittest module
- 10.7 LAB: Pet information (derived classes) Lab
- 10.8 LAB: Instrument information (derived classes) Lab
- 10.9 LAB: Course information (derived classes) Lab
- 10.10 LAB: Book information (overriding member methods) Lab
- 10.11 Copy of LAB: Book information (overriding member methods) Hidden Lab
- 10.12 LAB: Plant information Hidden Lab

11. Final Project

- 11.1 Final Project Lab

32. Final Project

- 32.1 Final Project Lab

31. Python for Data Science

- 31.1 Introduction to data science
- 31.2 Data science lifecycle
- 31.3 Introduction to Python for data science
- 31.4 Introduction to Jupyter Notebooks

18. Additional Material

- 18.1 String formatting using % Optional
- 18.2 String formatting using format() Optional
- 18.3 String formatting using dictionaries Optional
- 18.4 Basic graphics Optional

12. Exceptions

- 12.1 Handling exceptions using try and except Optional
- 12.2 Multiple exception handlers Optional
- 12.3 Raising exceptions Optional
- 12.4 Exceptions with functions Optional
- 12.5 Using finally to clean up Optional
- 12.6 Custom exception types Optional
- 12.7 LAB: Exception handling to detect input string vs. integer Optional Lab
- 12.8 LAB: Exceptions with lists Optional Lab
- 12.9 LAB: Step counter - exceptions Optional Lab
- 12.10 LAB: Fat-burning heart rate Hidden Optional Lab
- 12.11 LAB: Simple integer division - multiple exception handlers Hidden Optional Lab
- 12.12 LAB: Student info not found - custom exception types Hidden Optional Lab

13. Files

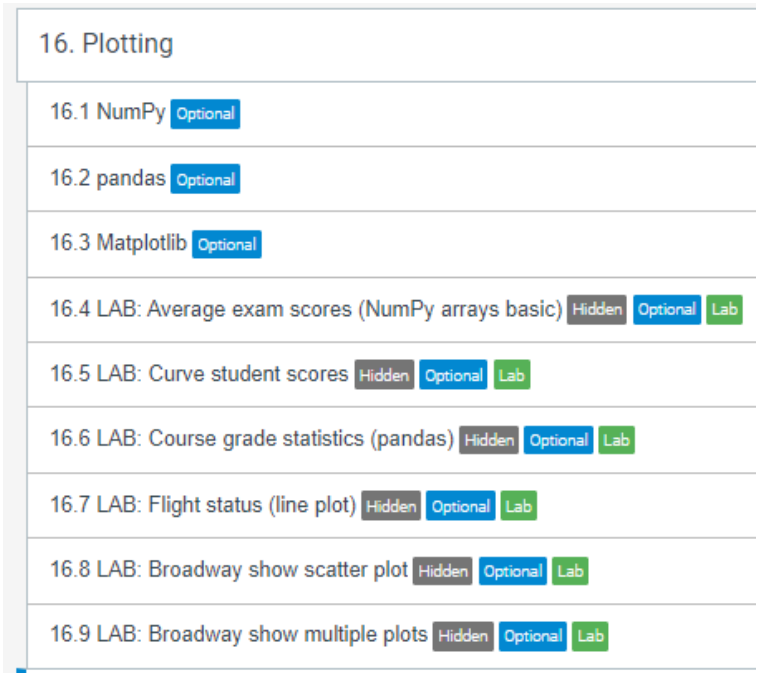
- 13.1 Reading files Optional
- 13.2 Writing files Optional
- 13.3 Interacting with file systems Optional
- 13.4 Binary data Optional
- 13.5 Command-line arguments and files Optional
- 13.6 The 'with' statement Optional
- 13.7 Comma-separated values files Optional
- 13.8 LAB: Words in a range (lists) Optional Lab
- 13.9 LAB: Word frequencies (lists) Optional Lab
- 13.10 LAB: Course Grade Optional Lab
- 13.11 LAB: Sorting TV Shows (dictionaries and lists) Hidden Optional Lab
- 13.12 LAB: File name change Hidden Optional Lab
- 13.13 LAB: Thesaurus Hidden Optional Lab
- 13.14 zyBooks built-in programming window Optional

14. Recursion

- 14.1 Recursive functions Optional
- 14.2 Recursive algorithm: Search Optional
- 14.3 Adding output statements for debugging Optional
- 14.4 Creating a recursive function Optional
- 14.5 Recursive math functions Optional
- 14.6 Recursive exploration of all possibilities Optional
- 14.7 LAB: All permutations of names Optional Lab
- 14.8 LAB: Number pattern Optional Lab
- 14.9 LAB: Fibonacci sequence (recursion) Optional Lab
- 14.10 LAB: Count the digits Hidden Optional Lab
- 14.11 LAB: Drawing a right side up triangle Hidden Optional Lab
- 14.12 LAB: Output a linked list Hidden Optional Lab

15. Modules

- 15.1 Modules Optional
- 15.2 Finding modules Optional
- 15.3 Importing specific names from a module Optional
- 15.4 Executing modules as scripts Optional
- 15.5 Reloading modules Optional
- 15.6 Packages Optional
- 15.7 Standard library Optional
- 15.8 LAB: Artwork label (modules) Hidden Optional Lab
- 15.9 LAB: Guess the random number Hidden Optional Lab
- 15.10 LAB: Quadratic formula Hidden Optional Lab
- 15.11 LAB: Unique random numbers (random module) Hidden Optional Lab
- 15.12 LAB: Dates Hidden Optional Lab
- 15.13 LAB: Radioactive decay Hidden Optional Lab


16. Plotting
16.1 NumPy Optional
16.2 pandas Optional
16.3 Matplotlib Optional
16.4 LAB: Average exam scores (NumPy arrays basic) Hidden Optional Lab
16.5 LAB: Curve student scores Hidden Optional Lab
16.6 LAB: Course grade statistics (pandas) Hidden Optional Lab
16.7 LAB: Flight status (line plot) Hidden Optional Lab
16.8 LAB: Broadway show scatter plot Hidden Optional Lab
16.9 LAB: Broadway show multiple plots Hidden Optional Lab


17. Searching and Sorting Algorithms
17.1 Searching and algorithms Optional
17.2 Binary search Optional
17.3 O notation Optional
17.4 Algorithm analysis Optional
17.5 Sorting: Introduction Optional
17.6 Selection sort Optional
17.7 Insertion sort Optional
17.8 Quicksort Optional
17.9 Merge sort Optional
17.10 LAB: Descending selection sort with output during execution Hidden Optional Lab
17.11 LAB: Sorting user IDs Hidden Optional Lab
17.12 LAB: Insertion sort Hidden Optional Lab
17.13 LAB: Merge sort Hidden Optional Lab
17.14 LAB: Binary Search Hidden Optional Lab