**Python for Beginners**

1. How Python Runs Programs
   1. Introducing the Python Interpreter
   2. Program Execution
   3. The Programmer’s View
   4. Python’s View
   5. Execution Model Variations
2. How You Run Programs
   1. The Interactive Prompt
   2. Starting an Interactive Session
   3. The System Path
   4. Where to Run: Code Directories
   5. What Not to Type: Prompts and Comments
   6. Running Code Interactively
   7. Why the Interactive Prompt?
   8. Usage Notes: The Interactive Prompt
   9. System Command Lines and Files
   10. A First Script
   11. Running Files with Command Lines
   12. Command-Line Usage Variations
   13. Usage Notes: Command Lines and Files
   14. Unix-Style Executable Scripts: #!
   15. Unix Script Basics
   16. Module Imports and Reloads
   17. Import and Reload Basics
   18. Usage Notes: import and reload
   19. Using exec to Run Module Files
   20. The IDLE User Interface
   21. IDLE Startup Details
   22. IDLE Basic Usage
   23. Advanced IDLE Tools
   24. Embedding Calls
   25. Frozen Binary Executables
   26. Text Editor Launch Options
3. Introducing Python Object Types
   1. Python’s Core Data Types
   2. Numbers
   3. Strings
   4. Sequence Operations
   5. Immutability
   6. Type-Specific Methods
   7. Other Ways to Code Strings
   8. Unicode Strings
   9. Pattern Matching
   10. Lists
   11. Sequence Operations
   12. Type-Specific Operations
   13. Bounds Checking
   14. Nesting
   15. Comprehensions
   16. Dictionaries
   17. Mapping Operations
   18. Sorting Keys: for Loops
   19. Iteration and Optimization
   20. Tuples
   21. Files
   22. Binary Bytes Files
   23. Unicode Text Files
   24. Other File-Like Tools
   25. Other Core Types
   26. How to Break Your Code’s Flexibility
4. Numeric Types
   1. Numeric Type Basics
   2. Numeric Literals
   3. Built-in Numeric Tools
   4. Python Expression Operators
   5. Variables and Basic Expressions
   6. Numeric Display Formats
   7. Comparisons: Normal and Chained
   8. Division: Classic, Floor, and True
   9. Integer Precision
   10. Complex Numbers
   11. Hex, Octal, Binary: Literals and Conversions
   12. Bitwise Operations
   13. Decimal Type
   14. Fraction Type
   15. Sets
   16. Booleans
   17. Numeric Extensions
5. String Fundamentals
   1. Unicode
   2. String Basics
   3. String Literals
   4. Single- and Double-Quoted Strings Are the Same
   5. Escape Sequences Represent Special Characters
   6. Raw Strings Suppress Escapes
   7. Triple Quotes Code Multiline Block Strings
   8. Indexing and Slicing
   9. String Conversion Tools
   10. Changing Strings
   11. String Methods
   12. Method Call Syntax
   13. Methods of Strings
   14. String Method Examples: Changing Strings II and Parsing Text
   15. The Original string Module’s Functions (not available in 3.X)
   16. String Formatting Expressions
   17. Advanced Formatting Expression Syntax
   18. Dictionary-Based Formatting Expressions
   19. String Formatting Method Calls
   20. Formatting Method Basics
   21. Adding Keys, Attributes, and Offsets
   22. Advanced Formatting Method Syntax
   23. Comparison to the % Formatting Expression
   24. General Type Categories
   25. Types Share Operation Sets by Categories
   26. Mutable Types Can Be Changed in Place
6. Lists and Dictionaries
   1. Lists
   2. Basic List Operations
   3. List Iteration and Comprehensions
   4. Indexing, Slicing, and Matrixes
   5. Changing Lists in Place
   6. Dictionaries
   7. Basic Dictionary Operations
   8. Changing Dictionaries in Place
   9. More Dictionary Methods
   10. Example: Movie Database
   11. Dictionary Usage Notes
   12. Other Ways to Make Dictionaries
   13. Dictionary Changes in Python 3.X and 2.7
7. . Tuples, Files, and Everything Else
   1. Tuples
   2. Why Lists and Tuples?
   3. Records Revisited: Named Tuples
   4. Files
   5. Opening Files
   6. Using Files
   7. Text and Binary Files: The Short Story
   8. Storing Python Objects in Files: Conversions
   9. Storing Native Python Objects: pickle
   10. Storing Python Objects in JSON Format
   11. Storing Packed Binary Data: struct
   12. File Context Managers
   13. Other File Tools
   14. Core Types Review and Summary
   15. Object Flexibility
   16. References Versus Copies
   17. Comparisons, Equality, and Truth
   18. The Meaning of True and False in Python
   19. Python’s Type Hierarchies
   20. Type Objects
   21. Other Types in Python
   22. Built-in Type Gotchas
   23. Assignment Creates References, Not Copies
   24. Repetition Adds One Level Deep
   25. Beware of Cyclic Data Structures
   26. Immutable Types Can’t Be Changed in Place