* Getting Started
  + Setting Up your programming environment
  + Python on different operating systems
  + Troubleshooting installation issues
  + Running Python programs from a terminal
* Variables and simple data types
  + Writing our first programme in python
    - Print Hello World
  + What really happens when you run hello\_world.py
  + Data types in python
    - Integer
    - Float
    - String
    - Boolean
    - Complex
    - Sequenced data
      * List
      * Tuple
      * Set
    - Mapped Data
      * Dictionary
  + Variables
    - Variable naming rules
    - Avoiding name errors when using variables
    - Try it by yourself
    - Operators
      * +, -, \*, /
      * \*\*, //, %
      * ^
* User input
  + How the input() function works
    - Writing Clear Prompts
    - Using int() to Accept Numerical Input
    - Using eval()
* Strings
  + Defining String
  + Overcoming Errors
  + Combining or Concatenating strings
  + Adding whitespace to strings with tabs or newlines
  + “\n”, “\t”
  + length of string
  + String slicing
  + String methods
    - Changing Case
      * upper
      * lower
      * capitalize
      * title
    - Striping Whitespace or any value
      * rstrip(), lstrip(), strip()
    - Modify
      * replace
      * center
    - Check values
      * count
      * ends with
      * starts with
      * find
      * index
      * is alphanumeric
      * is alphabet
      * is digit
    - Checking cases
      * is lower
      * is upper
      * is title
* Numbers
  + Integer Operations
  + Float Operations
  + Avoiding Type errors with str() function
* Something more
  + fstring
  + Comments
  + Type casting
  + Print
    - sep
    - end
  + The Zen of python
  + Mini Project
    - Temperature
    - BMI
    - Quadric
    - Currency
* If Statements
  + A simple example
  + Conditional Tests
    - Checking for Equality
    - Ignoring Case When Checking for Equality
    - Checking for Inequality
    - Numerical Comparisons
    - Checking Multiple Conditions
      * Using and to Check Multiple Conditions
      * Using or to Check Multiple Conditions
    - Checking Whether a Value Is in a List
    - Checking Whether a Value Is Not in a List
    - Boolean Expressions
  + If statements
    - Simple if Statements
    - if-else Statements
    - The if-elif-else Chain
    - Using Multiple elif Blocks
    - Omitting the else Block
    - Testing Multiple Conditions
  + Styling your if statements
  + Mini Project
    - Coin Flip (random module)
    - Grade
    - Calculator
    - Wisher (time module)
* Match Cases
  + Simple example
  + Default case
  + Default case with if
  + Mini Project
    - Calculator
* Introducing Lists
  + What is a list?
    - Accessing elements in a list
    - Index position
    - Using Individual values from a list
  + Changing adding, and removing elements
    - Modifying elements in a list
    - Adding elements to a list
      * appending Elements to the End of a List (append)
      * Inserting Elements into a List (insert)
    - Extending list
    - Removing elements from a list
      * removing an Item Using the del Statement
      * removing an Item Using the pop() Method
      * Popping Items from any Position in a List
      * removing an Item by Value
  + Check for values
    - index
    - count
  + Organizing a list
    - Sorting a List Permanently with the sort() Method
    - Sorting a List Temporarily with the sorted() Function
    - Printing a List in Reverse Order
    - Finding the Length of a List
    - Avoiding Index Errors When Working with Lists
* Working with Lists
  + Working with part of a List
    - Slicing a List
    - Copying a List
    - String as a list
  + Using if statements with lists
    - Checking for Special Items
    - Checking That a List Is Not Empty
      * If (list)
    - Using Multiple Lists
  + Styling your code
    - The Style Guide
    - Indentation
    - Line Length
    - Blank Lines
    - Other Style Guidelines
    - Try by yourself (<https://python.org/dev/peps/pep-0008/>)
* Tuples
  + Defining a Tuple
  + Looping Through All Values in a Tuple
  + Writing over a Tuple
  + Operations on tuple
* For loop
  + Simple for loop
  + For loop with string
  + For loop with list
    - Looping through the entire list
      * A Closer Look at Looping
      * Doing More Work Within a for Loop
    - Avoiding Indentation Errors
      * Forgetting to Indent
      * Forgetting to Indent Additional Lines
      * Indenting Unnecessarily
      * Indenting Unnecessarily After the Loop
      * Forgetting the Colon
    - Making Numerical Lists
      * Using the range() Function
      * Using range() to Make a List of Numbers
      * Simple Statistics with a List of Numbers
      * List Comprehensions
    - Enumerate
  + Mini project
    - Magic Balls
* While loops
  + Introducing while loops
    - The while Loop in Action
    - Letting the User Choose When to Quit
    - Using a Flag
    - Using break to Exit a Loop
    - Using continue in a Loop
    - Avoiding Infinite Loops
  + Using a while
    - Moving Items from One List to Another
  + Mini Project
    - Rock, Paper, Scissor
* Dictionaries
  + A simple dictionary
  + Working with Dictionaries
    - Accessing Values in a Dictionary
    - Adding New Key-Value Pairs
    - Starting with an Empty Dictionary
    - Modifying Values in a Dictionary
    - Removing Key-Value Pairs
    - A Dictionary of Similar Objects
  + Looping through a Dictionary
    - Looping Through All Key-Value Pairs
    - Looping Through All the Keys in a Dictionary
    - Looping Through a Dictionary’s Keys in Order
    - Looping Through All Values in a Dictionary
  + Nesting
    - A List of Dictionaries
    - A List in a Dictionary
    - A List in a Dictionary
  + Using a while
    - Filling a Dictionary with User Input
  + Mini project
    - KBC
* Functions
  + Defining a function
    - Passing Information to a Function
    - Arguments and Parameters
  + Passing arguments
    - Positional Arguments
      * Multiple Function Calls
      * Order Matters in Positional arguments
    - Keyword Arguments
    - Default Values
    - Equivalent Function Calls
    - Avoiding Argument Errors
  + Return values
    - Returning a Simple Value
    - Returning a Simple Value
    - Returning a Dictionary
    - Using a Function with a while Loop
  + Passing a List
    - Modifying a List in a Function
    - Preventing a Function from Modifying a List
  + Passing an Arbitrary number with arguments
    - Mixing Positional and Arbitrary Arguments
    - Mixing Positional and Arbitrary Arguments
  + Storing Your functions in modules
    - Importing an Entire Module
    - Importing Specific Functions
    - Using as to Give a Function an Alias
    - Using as to Give a Module an Alias
    - Using as to Give a Module an Alias
  + Doc string
  + Styling functions
* Classes
  + Creating and using a class
    - Creating the Dog Class
      * The \_\_init\_\_() Method
    - Making an Instance from a Class
      * Accessing attributes
      * Calling Methods
      * Creating Multiple Instances
  + Working with classes and instances
    - The Car Class
    - Setting a Default Value for an Attribute
    - Modifying Attribute Values
      * Modifying an attribute’s Value Directly
      * Modifying an attribute’s Value through a Method
      * Incrementing an attribute’s Value through a Method
  + Inheritance
    - The **init**() Method for a Child Class
    - Defining Attributes and Methods for the Child Class
    - Overriding Methods from the Parent Class
    - Instances as Attributes
    - Modelling Real-World Objects
  + Importing Classes
    - Importing a Single Class
    - Storing Multiple Classes in a Module
    - Importing Multiple Classes from a Module
    - Importing an Entire Module
    - Importing All Classes from a Module
    - Importing a Module into a Module
    - Finding Your Own Workflow
  + The Python Standard Library
  + Styling Classes
* Files and Exceptions
  + Reading from a file
    - Reading an Entire File
    - File Paths
    - Reading Line by Line
    - Making a List of Lines from a File
    - Working with a File’s Contents
    - Large Files: One Million Digits
    - Is Your Birthday Contained in Pi?
  + Writing to a file
    - Writing to an Empty File
    - Writing Multiple Lines
    - Appending to a File
  + Exceptions
    - Handling the ZeroDivisionError Exception
    - Using try-except Blocks
    - Using Exceptions to Prevent Crashes
    - The else Block
    - Handling the FileNotFoundError Exception
    - Analyzing Text
    - Working with Multiple Files
    - Failing Silently
    - Deciding Which Errors to Report
  + Storing Data
    - Using json.dump() and json.load()
    - Saving and Reading User-Generated Data
    - Refactoring
* Testing Your Code
  + Testing a Function
    - Unit Tests and Test Cases
    - A Passing Test
    - A Failing Test
    - Responding to a Failed Test
    - Adding New Tests
  + Testing a class
    - A Variety of Assert Methods
    - A Class to Test
    - Testing the AnonymousSurvey Class
    - The setUp() Method
* Jarvis- The AI
* Flappy Bird Game