

Harvard CS50 AP

Second Semester

Python Fundamentals

Standards Mastery Framework/Flex Timing Prototype Semester

12 Weeks

Executive Summary

With the exception of the final Master Project, any signoff can be completed *at any time*. It is, however, strongly recommended that they be accomplished in order.

Students will create a 1" Blue binder with cover sleeve which will contain all of their papers and code printouts, organized in order. This will be submitted for an EXAM grade. *Details provided in class.*

All **S** (small) assignments are considered **normal** coursework

All **M** (medium) assignments are **Quiz** grades*

All **L** (large) assignments are **EXAM** grades*

Grades will be assigned for Quizzes and Exams Individually

All Items marked with a **T** are development team items.

All assignments will be given a rating of 1-4 for mastery as follows:

1: Student only understands most basic definitions/concepts. Cannot demonstrate mastery

2: Student can show limited understanding of the definitions/concepts with assistance

3: Student has mastered the definitions/concepts and can perform assigned tasks to standard(s)

4: Student has mastered and exceeded the standard by performing and demonstrating skills beyond what has been explicitly taught and/or required

Rankings will be assigned on the software tracker as follows:

Blank/0: No signoff/Not Attempted/No Credit

Red: 1

Blue: 2

Silver: 3

Gold: 4

For purposes of this initial program, half points will not be awarded

***Any irregularities on assessments are considered honor violations**

Running list of assignments

Size	#	Task
S	1	Code: Working POCO class, minimum 2 fields, full getters and setters
S	2	Code: 2 Working POCO classes for which one has a field (Attribute) of the other
S	3	Code: Working Python program that draws a graphical picture on the Canvas
M	4	Code: Python Complete: Project showing some stretch item not covered before in class
S	5	Paper: 1-page overview of history of Java.
S	6	Code: Hello World in Java / Java Install / Eclipse Install
S	7	Code: First method in main – Any quality PROCEDURE
S	8	Code: Second method in main – Any quality FUNCTION
M	9	Code: Show knowledge of casting from various data types to others
S	10	Code: Show knowledge of fixed number iteration (Some fixed # of loops with output)
S	11	Paper: 1-page paper on casting data types to each other and its effects on accuracy/program correctness
M	12	Code: Program that uses at least 5 String library functions, with complex output sequence
S	13	Code: Show use of all three types of Iteration
S	14	Code: Show use of conditionals (IF/Else & Switch)
S	15	Code: Write a program that adds up an array[25] if int
L	16	Code: Program that contains two functions: One that calls the other for a value it needs
S	17	Paper: 1-page paper that discusses getters/setters and ENCAPSULATION
S	18	Code: Write a POCO class in Java
S	19	Code: Exercise POCO class from 19
M	20	Code: Rewrite main() from 20 to include an Array of your POCO class objects
S	21	Code: Demonstrate use of JAVADOCS on one method
S	22	Code: Demonstrate use of Try-Except statements in your code
S	23	Code: Create a text file, and read it into a program
M	24	Code: Create a program that reads a text file, changes the text, then writes the new text back to the text file
S	25	Code: Create a basic Lambda function
S	26	Code: Demonstrate knowledge of 4 new built-in Java functions
S	27	Code: Demonstrate knowledge of 4 additional built-in Java functions
M	28	Code: Write a meaningful program that uses at least 1 POCO class totally new to you. user
S	29 T	Write a program that performs a simple function on a Raspberry Pi
S	30 T	Turn on an LED on a Raspberry Pi
S	31 T	Demonstrate use of an RGB LED on a Raspberry Pi
L	32 T	Write a Python program on a raspberry Pi that activates LED(s) based on a novel schema
M	33 T	Convert C program Vigenère to Java
S	34	Paper 1-page paper reflecting on Java as you have learned it.
M	35	Demonstrate Tutoring Skills by helping another student WITHOUT giving them code.**
MASTER PROJECT		
Phase A		Submit detailed project proposal
Phase B		Checkpoint 1
Phase C		Checkpoint 2
Completion		Submit final project
Presentation & Defense		Project Presentation, Demonstration, and Defense

Specifications Details for Signoffs and Checkpoints

1. POCO Python class. Minimum content for class:
 - a. Constructor (`__init__`):
 - b. 2 Fields (recommend int and string)
 - c. Getters and setters for all fields
 - d. `displayClass` or `printClass` method - do NOT name it "`displayClass`" or "`printClass.`" (think about our example)
2. Program with 2 POCO classes, one of which contains a field that is of type [the other one]. You only need to exercise the class that hold the other in main because you must initialize the contained class in the constructor of the holding class.
3. Program that draws a picture on the Python canvas. Simple shapes = 3.
4. Program that shows me you learned something significant on your own about python. Something not covered specifically in class or done by over 5 others in class. First come first served. **Assessment (Quiz)**
5. One-page paper on history and background (origins/future) of Java.
6. Program "Hello World" with Eclipse and JDK (Latest) installed.
7. Any PROCEDURE written in MAIN method (Hint: not `main()`) that is called as part of program execution.
8. Any FUNCTION written in MAIN method (Hint: not `main()`) that is called as part of program execution and receives a useful value from function.
9. Program that shows knowledge of casting from one data type to another. **Assessment (Quiz)**
10. Program should loop a fixed number of times and perform some calculation based on the loop, and display results properly and accurately.
11. One-page paper on casting data types. Discuss affect on program correctness. Discuss at least three data types.
12. Program that uses at least 5 String functions from Java library properly (Significant demonstration of their function in program output)
13. Program that demonstrates all three forms of iteration in non-trivial manner. **Assessment (Quiz)**
14. Program properly uses IF, IF/ELSE, and SWITCH statements properly (No fall-through on switch) .
15. Program adds up 25 integers stored in an array and displays the total neatly and properly. Write a named function that returns a value and use it in your program.
16. Program uses two functions in Main, one that calls the other function for a value it needs. Real-world only.
Assessment (Exam)

17. One-page paper explaining getters and setters, and **ENCAPSULATION** <- most important
18. Program has a POCO class in Java (3 field minimum) with full constructor, partial constructor, and default constructor
19. Program exercises POCO class in 18.
20. Program Rewrite main() from 20 to include an ARRAY of your POCO objects. **Assessment (Quiz)**
21. Program demonstrates the use of the JAVADOCS on at least one method. Print JAVADOCS documentation.
22. Program demonstrates use of Java Try-Except statement, with proper syntax of both working and error-throwing code to demonstrate correct exception handling
23. Program accesses a text file and internalizes its content, then prints it in a formatted manner
24. Program reads input from a text file, alters or updates the information, then returns it to the text file. Programmer can determine if it appends or replaces the data. Text file before and after is part of program signoff as well as source code. **Assessment (Quiz)**
25. Program creates and *documents within the code*, a single non-trivial lambda function. Your program must access the function at least three times with different values as input to the function.
26. Program uses and documents the use and understanding of four (4) built-in native Java functions.
27. Program uses and documents the use and understanding of four (4) *other* (additional) built-in native Java functions.
28. Program instantiates and exercises a new POCO class you write, showing me stretch from 18-20. **Assessment (Quiz)**
29. Program performs a single function on a Raspberry Pi (No Electronics required).
30. Program turns on and off, a single LED on a Raspberry Pi.
31. Program activates all three colors on an RGB LED at programmer's discretion.
32. Program activates multiple LED lights on a Raspberry Pi in some manner determined by the programmer. **Assessment (Exam)**
33. Program converts Vigenère program from C to Java(porting) **Assessment (Quiz)**
34. One-page paper reflecting on what you have learned about Java. Must be easy to read and thoughtful, canvassing points you may have struggled and how you moved forward.
35. Demonstrate tutoring skills helping another student catch up WITHOUT handing them any code. You must explain process to them, you can help them debug but you cannot hand them solutions in any way. **Assessment (Quiz)** *****If you are struggling and someone is helping you, I will provide you an alternate for this on request if I believe you are truly working to master the work.***

Spring Master Project Submission Checklist

Python Program (Weighted to a 3) or JAVA (Weighted to a 4)

Must-includes (Graded Directly):

- ☐ External Classes (2) Python, 2XPOCO if Java
- ☐ Module (1) (In addition to main program) (Static class if Java)
- ☐ Methods (3) (May be functions and/or procedures)
- ☐ Lambda Function (1)
- ☐ User Input (multiple) (Python) List<> if Java
- ☐ Output (Results, prompts, etc.)
- ☐ Full documentation including header
- ☐ 3 Python built-in functions, 3 methods in Java per class.
- ☐ Must be meaningful code, that is on a subject unique to your personal interests. This is mandatory. Do not waste time writing code in which you are not interested.
- ☐ Must at a minimum, store information in an external file.

Project Enhancements:

- ☐ Program uses electronics/Raspberry Pi
- ☐ Program reads/writes from a text file
- ☐ Program reads/writes tuples, lists, or other ADS (abstract data structure) from a file
- ☐ Program uses a database on localhost
- ☐ Program uses TKinter()/GUI Note: Proper JavaFx that meets other specs gets +10 **actual extra credit**.
- ☐ Other (Preapproved) _____
- ☐ Other (Surprise) _____

Submission Notes to teacher:

By signing below, I certify that the code I have written is my own. No one else has written this code but me, and any help I have received has been for clarification/understanding alone. No one has formed any algorithms in this work product but me.

Name (Printed)

Signature

Date

Spring Master Project Proposal

Class: Harvard CS50 AP

Period (Circle): 4-Schenk 6-Schenk

Language (Circle): Python Java

Name (FULL): _____

Executive Summary

Specifics to Assist in Project Approval

I understand that all code submitted under my Spring project must be of my own authoring. No one else may generate or otherwise write any algorithms for me. Any code submitted that is 3rd party will be presented to the teacher for prior approval if specific algorithms or libraries are required outside the libraries available in class as part of our normal lab work or language libraries provided.

Signature

Date

Project is: ☐ Approved ☐ Unapproved (See teacher) or
☐ Approved as modified below

Spring Master Project Grading Checklist

Class: Harvard CS50 AP

Period (Circle): 4-Schenk 6-Schenk

Name (FULL): _____

Python Program (Weighted to a 3) or JAVA (Weighted to a 4)

Must-includes (Graded Directly):

- ☐ External Classes (2) Python, 2XPOCO if Java
- ☐ Module (1) (In addition to main program) (Static class if Java)
- ☐ Methods (3) (May be functions and/or procedures)
- ☐ Lambda Function (1)
- ☐ User Input (multiple) (Python) List<> if Java
- ☐ Output (Results, prompts, etc.)
- ☐ Full documentation including header
- ☐ 3 Python built-in functions, 3 methods in Java per class.
- ☐ Must be meaningful code, that is on a subject unique to your personal interests. This is mandatory. Do not waste time writing code in which you are not interested.
- ☐ Must at a minimum, store information in an external file.

Project Enhancements:

- ☐ Program uses electronics/Raspberry Pi
- ☐ Program reads/writes from a text file
- ☐ Program reads/writes tuples, lists, or other ADS (abstract data structure) from a file
- ☐ Program uses a database on localhost
- ☐ Program uses TKinter()/GUI Note: Proper JavaFx that meets other specs gets +10 **actual extra credit**.
- ☐ Other (Preapproved) _____
- ☐ Other (Surprise) _____

Assigned Grade: _____ / _____ = _____ %

Teacher Notes: