

Quarter 2, 2023

COS60016: Programming for Development

Assignment 1: Build a web-based API integration framework

Word/time limit: 1000 (+/- 10%)

Weighting: 40%

Due date: 5pm AEST Monday 29 May 2023 (Week 5)

After you have read this information, head over to the [Assignment 1 Q&A](https://swinburneonline.in) (<https://swinburneonline.in> 691856) discussion board to ask a question about this assignment.

Assignment overview

In this assignment, you will use Python to create a web-based API integration framework. You will submit a 1000-word report to show your understanding of the framework. This assignment supports learning outcomes 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100.

Assignment details

For this assignment, you'll build a web-based framework to retrieve data from an API and write a 1000-word report. Your report should reflect on the lessons you have learned.

Python is the most popular choice for working with APIs. In this assignment, you will use an API, HTTP, JSON, Flask and Python to address a business scenario.

Step 1: Read the scenario

Scenario

A travel blogger has been commissioned by a tourist website to explore England and post regular updates on their trip. In preparation for this blog series, the owners of the website have asked you to develop an **integrated web-based solution** which will **return** weather data for each specific location on the travel blogger's itinerary.

Itinerary:

- Lake District National Park (54.4609° N, 3.0886° W)
- Corfe Castle (50.6395° N, 2.0566° W)
- The Cotswolds (51.8330° N, 1.8433° W)
- Cambridge (52.2053° N, 0.1218° E)
- Bristol (51.4545° N, 2.5879° W)
- Oxford (51.7520° N, 1.2578° W)
- Norwich (52.6309° N, 1.1000° E)
- Stonehenge (51.1100° N, 2.0000° W)
- Watergate Bay (50.4500° N, 5.0000° W)
- Birmingham (52.4800° N, 1.9000° W)



Step 2: Selecting an API

1. You will create a draft solution that describes how this solution integrates with an API, based on the scenario. This should include a justification for the API chosen.
2. You will then document the solution, demonstrating your ability to use the code to integrate your application.
3. You will also access the API using Python and Flask to create a functional app which can send queries to and receive data from an external API.

Note

- At a minimum, your solution should demonstrate successful integration with an API and provide a text-based response.
- You will demonstrate proficiency by using what you have learned to create an input and output that go a bit beyond basic text.

- To excel, you will need to demonstrate additional elements, for example, a map within your framework that illustrates the destination points of the scenario.

Step 3: Complete your report

Your 1000-word PDF report should include:

1. an analysis of the scenario, an outline plan of your solution and a justification for your choice
2. documentation of your solution, including API methods, description of elements, and examples of usage
3. code snippets demonstrating the use of HTTP, JSON, HTML and Flask and the integration with an API
4. screenshots of the working solution
5. a conclusion and summary

Tables, images and code

Suggested approach

The assessment is broken down into three parts, each with a specific focus on completion and submission.

Part	
1	Analysis and plan <ul style="list-style-type: none"> • Analysis of the scenario • A brief outline of your solution • A justification for this solution
2	Documentation of your solution: <ul style="list-style-type: none"> • API methods, such as example requests and responses and descriptions of elements in each • A brief explanation of how to use your solution
3	Snippets of commented code:

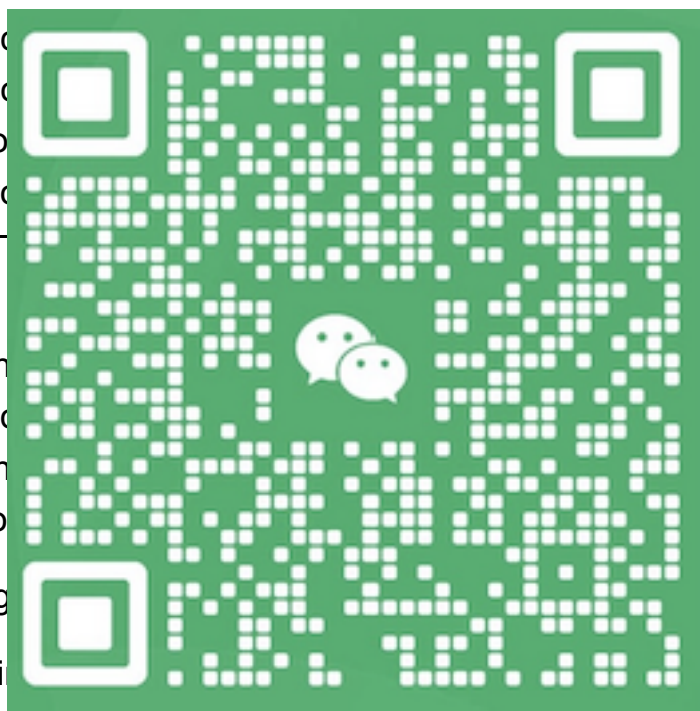
	<ul style="list-style-type: none"> • Demonstrate the use of HTTP, JSON, HTML and Flask, as well as integration with an API • Include code that addresses any identified limitations of the API
4	Screenshots of your working solution: <ul style="list-style-type: none"> • Show a user query and a response within a web framework
5	Conclusion and reflection of learning: <ul style="list-style-type: none"> • Write a summary of the process you followed to complete the assessment. • Include information about how you addressed errors in your code and how you researched solutions to challenges you faced. • Discuss practical solution and how you implemented it. • You could also reflect on your learning into your future work.

Tips for success:

A Python programmer needs to indicate how to blockers experienced, not of findings (interpretation)

To succeed in this assignment

- Read the guidelines
- Ensure you've completed all the relevant sections.
- Ensure that your writing is concise and clear.
- Provide clear explanations for each step. (For example, "I experienced a problem with the API call, but implemented the following workaround...")
- Proofread your document and ensure that it adheres to basic grammatical principles.




fore, your report L patterns, any and a summary

Submission details overview

This assignment will be submitted through Canvas. When you are ready to submit your assignment, select the 'Start Assignment' button at the top of this page. You will be taken to the 'File Upload' tab where you can choose your file or submit your URL.

Please note: When you submit your assignment through Canvas, you are also submitting the assignment through Turnitin, which is a text-matching service that compares your work with an international database of information sources. You will need to agree to using it.

Once you have submitted your assignment, select 'Submission Details' on the right of your screen to view your originality report if you haven't already done so.

Please allow a 24-hour turnaround for an originality report to be generated. See the 'Turnitin originality report' area of the [Academic practice](https://studenthub.swinburneonline.edu.au/academic-practice)  (<https://studenthub.swinburneonline.edu.au/academic-practice>)_ page in the Study Resources section of the submission process.

**Resubmissions after the
Coordinator may not be**

ur Unit

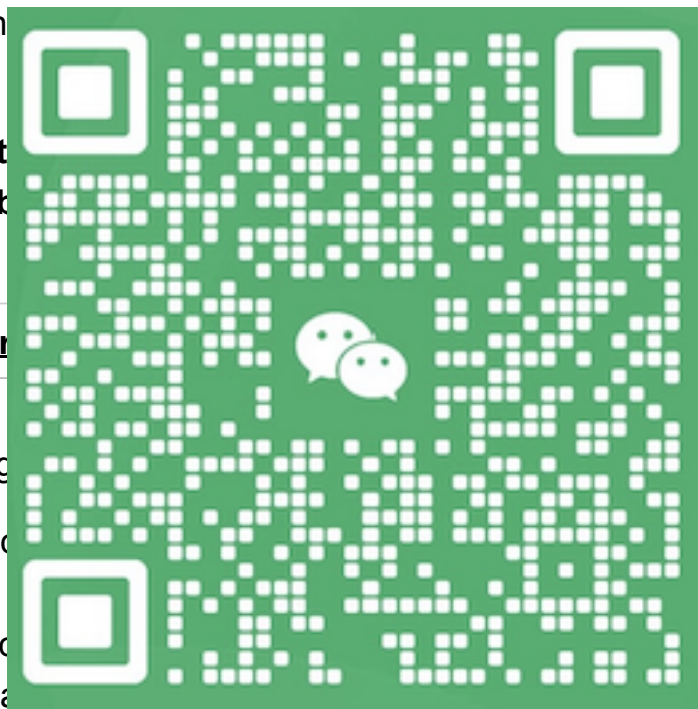
Assignment criteria

You will be assessed against the following criteria:

1. An analysis of the social context of your choice.
2. Documentation of your choice and examples of use.
3. Code snippets demonstrating the use of HTTP, JSON, HTML and Flask, as well as integration with an API.
4. Screenshots of the working solution.
5. A conclusion and short reflection on your learning.

Your work will be assessed using the following marking guide:

Assignment 1 marking guide



Criteria	No Pass	Pass 50–59%	Credit 60–69%	Distinction 70–79%	First Class 80–89%
Analysis and plan Justify the appropriate use of APIs in a business context. (10%)	No attempt to justify or contextualise decisions.	An attempt has been made to justify decisions made, with some context provided.	Clear justification and context for decisions made.	Clear justification and context for decisions made.	Excellent justification and context for decisions made.
Documentation Write and document an API call. (20%)	No method examples provided.	Some method examples provided.	Method examples provided.	Complete documentation of API methods, with clear examples provided. Usage clearly described.	Excellent documentation of API methods, with clear examples provided. Usage clearly described.
Commented code samples Use Python to write a simple program. (30%)	No attempt to complete the code. No attempt to address limitations.	Code displays limited understanding of Python programming. Some attempt to address limitations.	The code is mostly complete, may lack commenting or logical flow but works on a simple task.	Code is complete, with clear commenting. Evidence of efficient coding.	Excellent code, with clear commenting. Evidence of efficient coding.

Criteria	No Pass	Pass to address limitations. 50–59%	Credit Attempts made to address limitations. 60–69%	Distinction structure. Key limitations are addressed. 70–79%	High Distinction structure. Key limitations are addressed. 80–89%
Working solution Use Python libraries to access data through APIs. (30%)	No attempt has been made to build the framework.	The API call does not work, but an attempt has been made to build the framework.	The API call is almost entirely complete and mostly functional. May have bugs. The framework	The API call is complete and fully functional, without bugs. The framework fully meets the requirements.	
Reflection of learning (10%)	No evaluation of application	the assignment. Thoughts not clearly organised.	unclear in some parts and concepts are not communicated logically.	Clear justification and evaluation of choices made. Writing is organised, with concepts communicated in a mostly coherent and logical manner.	

Request assignment extension

We understand that at times your studies may be adversely affected by illness, misadventure or some other extraordinary cause or circumstance reasonably beyond your control. If you feel this is the case and you need an extension to support you in completing your assignment, please select 'I would like to request an extension' below to access the request process.

Please note: For team assignments, each team member must submit a request.

I would like to request an extension (../external_tools/2948).

I would like to view an existing request (../external_tools/2948).

Assessment declaration

All students must agree to the following declaration and statement of understanding before submitting assignment items.

Declaration and Statement of Understanding

1. I have not impersonated any person for the purposes of this assignment.
2. This assignment is my own work and has not been copied from any other source except where acknowledged.
3. No part of this assignment has been copied or taken from any source except where acknowledged.
4. I have not previously submitted this assignment for assessment in any unit.
5. I give permission for my work to be used for educational purposes.

I understand that:

- Plagiarism is the presentation of another person's work as though it is your own. It is a form of cheating and is a very serious academic offence that may lead to exclusion from the university.
- Plagiarised material may be drawn from published and unpublished written documents, interpretations, computer software, designs, music, sounds, images, photographs, and ideas or ideological frameworks gained through working with another person or in a group.

- Plagiarised material can be drawn from, and presented in, written, graphic and visual form, including electronic data and oral presentations. Plagiarism occurs when the origin of the material used is not appropriately cited.

I agree and acknowledge that:

1. I have read and understood the Declaration and Statement of Authorship above.
2. I accept that use of my Swinburne account to electronically submit this assignment constitutes my agreement to the Declaration and Statement of Authorship.
3. If I do not agree to the Declaration and Statement of Authorship in this context, the assessment outcome may not be valid for assessment purposes and may not be included in my aggregate score for this unit.
4. I am aware that it is not acceptable to resubmit the same piece of work (in part or as whole) for multiple assignments without permission from the Unit Coordinator.

Penalties for **plagiarism** (<https://www.swinburne.edu.au/current-students/manage-course/exams-results-assessment/academic-misconduct/>) range from a formal caution to suspension. Detailed information is available in the **Student Academic Misconduct Regulations** (<https://www.swinburne.edu.au/current-students/manage-course/exams-results-assessment/academic-misconduct/>) and the **University of Melbourne Regulations/Statutes** (<https://www.unimelb.edu.au/regulations/statutes-regulations/student-academic-misconduct/>).



