

NORTHWEST NAZARENE UNIVERSITY

Improving a General Education Data Pipeline with
Python Scripts, Automated Emails, and Power BI

THESIS

Submitted to the Department of Mathematics and Computer Science

in partial fulfillment of the requirements

for the degree of

BACHELOR OF ARTS

Bryce Z. Miller

2024

THESIS

Submitted to the Department of Mathematics and Computer Science

in partial fulfillment of the requirements

For the degree of

BACHELOR OF ARTS

By

Bryce Miller

2024

Improving a General Education Data Pipeline with
Python Scripts, Automated Emails, and Power BI

Author :



Bryce Miller

Approved :



DALE HAMILTON, Ph.D., Professor of Computer Science,
Department of Mathematics & Computer Science, Faculty Advisor

Approved :



Catherine Becker, Ph.D., Professor Of English,
Department of Communication, Language, & Literature, Second Reader

Approved :



Dale Hamilton, Ph.D., Chair
Department of Mathematics & Computer Science

ABSTRACT

Improving a General Education data pipeline with Python scripts, automated emails, and Power BI

MILLER, BRYCE (Department of Mathematics and Computer Science), HAMILTON, DR. DALE (Department of Mathematics and Computer Science), BECKER, DR. CATHERINE (Department of Language and Literature).

Assessing General Education (GE) student outcomes for accreditation occurs regularly at higher education liberal arts institutions; however, the process can be complicated and time-consuming, and the resulting outcome data may not be meaningful on its own. This project automates Northwest Nazarene University's GE outcome assessment process using Python scripts and the Canvas Application Programming Interface (API). The Python scripts automatically send personalized training and reminder emails to GE instructors, retrieve, clean and enrich GE outcome data, and create GE assessment compliance data. Upon completion, Microsoft Power BI visualizes the data. By automating these actions, the project streamlined Northwest Nazarene University's GE pipeline and transformed years of data into actionable information that could also be shown to accreditors. Furthermore, Microsoft Power BI was leveraged to show how consistently specific colleges, disciplines, departments, courses, and instructors were assessing their GE outcomes, thereby providing college deans and department chairs with information on where to focus their efforts to increase GE outcome compliance.

Acknowledgments

I would first like to thank Bethany Shultz, E.D.D., for her encouragement to start, continue, and complete the process of expanding my education to include the field of computer science. I would not be finishing this project or this degree without her support as my supervisor or her willingness for me to take on this project as part of my duties as Northwest Nazarene University's Learning Management System Administrator. I also want to thank my academic and project advisor, Dale Hamilton, Ph.D., and my second reader and project stakeholder, Catherine Becker, Ph.D., for their advice and feedback throughout this project. Most importantly, I want to thank my cherished wife, Merry, for her love, grace, and support. I cannot thank her enough for all the times she has been there for me throughout this project while simultaneously caring for our eldest and the twins born during this project.

List of Figures

Figure 1: GE Courses Assessed	9
Figure 2: Number of GE Student Outcomes Assessed	9
Figure 3: Average Outcome Scores	10

Table of Contents

Title Page	i
Signature Page	ii
Acknowledgments	iv
List of Figures	v
Table of Contents	vi
Introduction	1
Background	2
Proposal	3
Implementation	4
Creating an Active GE Course List	4
Importing Outcomes into Courses	4
Sending Personalized Introduction and Reminder emails	6
Cleaning and Transforming the Data	8
Displaying Outcome and Compliance Data	8
Resources	10
Results	11
Met Goals	11
Future Work	13
Conclusion	14
References	15

Introduction

Nearly every higher education institution declares itself in some way as accredited as part of its claim to legitimacy and to support the value of its degrees. Accreditation is “focused on the relevance criteria for quality of education and define first and foremost relevance as relevant to the labour market” and “should also refer to other expectations and demands that society ask for” (Bendixen & Jacobsen, 2020). In other words, it is an institution’s accreditation that validates and proves the quality of the degrees it grants. For liberal education higher education institutions, the accreditation process also validates the worth and quality of the institution’s General Education program. General Education (GE) is the foundation of undergraduate studies at liberal education institutions. Northwest Nazarene University (NNU) defines the objective of General Education as being to “provide a common experience for the personal and intellectual growth of each student” and to bring students “into contact with the great persons, ideas, and movement of human culture and the Christian faith” (*The General Education Program | Northwest Nazarene University*, n.d.). Across the field of General Education within higher education, accreditation now requires that specific outcomes be defined for GE-required courses. Particularly outcomes that assess the degree to which students acquire the breadth and depth of the learning that the GE course was intended to teach as part of the GE program. For NNU, this is These assessments are also increasingly used internally for program review, process improvement, and more (Kuh et al., 2014). Overall, the creation and application of GE outcomes are now perceived as increasingly vital for General Education programs. NNU has chosen to measure the outcomes it has defined by assessing the degree to which each GE student achieved the desired outcome within each of the associated courses; however, the reliability and usefulness of the resulting assessment data is directly related to how consistently GE outcomes are assessed. For instance, if

GE assessment was graded and only 50% of the students enrolled in GE courses were assessed, even if all of the students in the assessed courses met expectations, the GE assessment could still receive a failing grade if the other 50% of GE courses were below expectations. This need for consistent assessment, combined with the fact that many GE instructors are short-term adjunct faculty and that the outcomes of a course tend to be measured once per semester, creates a need for a General Education assessment process that is simple and streamlined. Furthermore, there is a need for consistent communication from the overseeing authority to train and remind instructors of GE courses of the assessment requirements. This project has fulfilled these needs for Northwest Nazarene University (NNU).

Background

NNU is accredited by the Northwest Commission on Colleges and Universities (NWCCU). To maintain this, every seven years an NWCCU accreditation committee made up of qualified personnel from peer institutions comes to assess NNU's education programs and recommend necessary changes. In NNU's 2016 Year 7 Accreditation Review, the necessity of a quality outcome assessment plan was made abundantly clear. The Evaluation Committee's report stated:

We observed a detailed GE Assessment Map and a representative sample of syllabi which include outcomes aligned to the core themes and anchored to specific indicators (courses) and data sources (assignments). ... However, of significant note, *data was not presented to* [emphasis added] demonstrate that departments have gone through a complete assessment and review cycle to *evaluate the effectiveness of their programs in reaching their learning outcomes or proposing necessary changes* [emphasis added] (D. Slemmer, personal communication, March 22, 2024).

The university received the recommendation that “the institution implement the newly developed General Education assessment plan in order to *assess student learning outcomes for those programs and utilize said assessment for continuous improvement* [emphasis added]” (D. Slemmer, personal communication, March 22, 2024).

The University’s General Education Council (GEC) promptly implemented the assessment plan. In NNU’s 2019 Mid-Cycle Review, an Ad Hoc report from the evaluation team stated that the implementation was “largely successful” (D. Slemmer, personal communication, March 22, 2024). As a result, the status of the recommendation was set as fulfilled. However, while the new system was effective, it was prohibitively complicated, requiring numerous manual steps to create, retrieve, and record the outcome assessment data, and it was difficult to turn the resulting outcome data into usable information. This, combined with the infrequent and uneven communication and training regarding GE assessment, made it particularly difficult for the GEC to maintain the integrity of NNU’s General Education outcome data through consistent measurement.

Proposal

NNU’s General Education Council (GEC) was very aware of these issues with the GE data pipeline, and it was believed that, if left unchanged, NNU would receive another recommendation regarding necessary changes to its GE program at the 2023 Seven Year Accreditation Review. Dale Hamilton, Ph.D., a member of the GEC, and Catherine Becker, Ph.D., the chair of the GEC, proposed to automate and simplify as much of the GE data pipeline as possible. The goals of the project were to automate the following:

1. Import the most recent version of GE course outcomes into their associated GE courses for that term.

2. Send personalized introductory, reminder, and alert emails to GE course instructors.
3. Retrieve and clean up the resulting outcome data.
4. Record final GE course compliance status
5. Display GE outcome data and GE assessment compliance data in a usable format.
6. Retrieve outcome artifacts.

Implementation

Creating an Active GE Course List

Acquiring a list of a term's active GE courses was the first step in the automation process. Unfortunately, no accessible list of Active GE courses existed, so a method of creating one had to be written. This was resolved by creating a Python script that used the Google API to retrieve the GE Course Codes and associated outcomes from a Google sheet, which the GEC maintained. Then, the script used the Canvas (NNU's LMS) API to create Comma Separated Value (CSV) reports of the given term's courses, enrollments, and users.

Once the raw data was retrieved, the Python script could search for the GE course codes within the course report for active GE courses and retrieve their unique identifiers (IDs). Each ID could then be used in conjunction with the enrollment report to count the number of students in the course and find the unique user ID of the course's instructor/s. The instructor ID/s could then be cross-referenced with the user report to find the name and email of the course's instructor/s. The result was a list of all active NNU GE course IDs with their associated outcome or outcomes, number of students, and instructor information.

Importing Outcomes into Courses

Once a list of active GE courses was acquired, the first goal was to import the most recent version of each GE course outcome into the associated active GE courses. This could have been

a one-time requirement for each GE course because imported outcomes are included whenever the content of a Canvas course is copied over to another Canvas course. However, because all NNU GE outcomes are remade with different titles every year in case their descriptions change, the new outcomes must be re-imported into their associated courses every school year.

Otherwise, the outcomes used in the course become more and more out of date. The process of importing the outcome and then attaching it to an assignment was a multistep endeavor that new GE course instructors were inconsistently taught, and veteran GE instructors frequently forgot. This issue was further exacerbated by the reality that the process of importing and attaching an outcome to a GE course was generally performed once per GE course taught in that term, and some instructors only taught one GE course which itself at times was only taught every other year.

This goal of automatically importing outcomes immediately ran into an obstacle. Namely, Canvas has no API call to import outcomes into courses. Fortunately, the issue was resolved by using the Canvas course import API call instead. Empty Canvas course shells were made for each outcome and given the unique outcome ID as their unique course ID. A Python script was written that would then iterate through the previously created list of Active GE courses and use the relevant term and year in conjunction with the associated GE outcome to determine the unique outcome ID and corresponding course ID. Once the unique outcome ID was determined, the script could import the outcome by utilize the Canvas API to initiate a Canvas course copy of the course shell with the outcome into the active GE course to the GE course.

Sending Personalized Introduction and Reminder emails

The project's second goal was to send personalized emails to GE instructors to introduce them to the outcome assessment plan and remind them to attach and complete the assessment. This was prioritized because, for a GE Canvas Outcome to be assessed, it must be connected to a published assignment and assessed as a rubric item after being imported into the course. As previously mentioned, communication regarding the requirements of the GE outcome assessment was uneven, which was partly the cause of gaps in the GE outcome assessment data, as veteran instructors occasionally forgot which of their courses were part of the GE program, and new GE instructors (particularly adjuncts) might not have been told that one or more of their courses were GE courses and needed to be assessed against a GE outcome.

The introduction email was easily automated by using a Python script that first retrieved the list of active GE courses with their associated instructor information and then retrieved a CSV of the course data being uploaded from NNU's Student Information System (SIS). The script then would use the start and end date data from the SIS course information and the current date to determine whether the course was an 8 or 16-week course and whether the current date was before, during, or after the target course. The week before the course started (week 0), an introduction email, written and formatted by the chair of the GEC, would be sent to the emails of the course's instructor/s from the email address of the GEC chair using the Gmail API. These introduction emails communicated that the instructor was teaching a GE course, detailed how to attach and assess the relevant GE outcome, and included personalization by using the instructor's/s' name/s, the name of the GE course, and the associated outcome/s.

The emailed reminders were more challenging. It was decided that if reminders were sent out to all GE instructors without regard for whether they had already attached or assessed the

outcome, there was a risk of encouraging a culture among faculty to ignore the reminders. Consequently, it was necessary to create a new list of any GE courses that had not attached the relevant outcome to a published assignment. A Python script was written to create such a list by examining the rubrics of each GE course to discover whether at least one rubric with the outcome attached was connected to a published assignment. If not, the course was marked accordingly.

By using the list of GE courses that did not have the required outcome attached, the personalized emailer script was able to send reminders to the related instructors. The reminder emails alerted them to the fact that they still needed to attach their GE outcome to a published assignment and again provided them with directions on how to do so. These reminder emails were sent out the week before the course's midterms (week 7 in a 16-week course) and again the week before the course's finals (week 15 in a 16-week) course if the outcome remained unattached.

Lastly, it was desired to send a missing data alert email to the instructors of courses that either had no outcome data for the associated GE outcome or had such data for less than 75% of the course's total students. To enable this, a Python script was developed to use the Canvas API to retrieve the outcome results for the given term, cross reference them with the list of active GE courses, and ultimately save a CSV list of every active GE course and its information with the additional datapoint of whether the course assessed, partially assessed, or did not assess for the relevant GE outcome. Using this data point, the personalized emailer script was enabled to send missing data alerts the week after finals to the instructor/s of GE courses that either did not assess or only partially assessed.

Cleaning and Transforming the Data

The third goal of the project, to automatically retrieve outcome data, was accomplished using the Canvas API. The fourth goal, to record GE assessment compliance, was fulfilled by the same script that created a list of GE courses and labeled them as assessed, partially assessed, and not assessed. However, the retrieved outcome data needed to be cleaned. The GE assessment program wanted one score between 0 and 3 (with 0 being fully unmet, 2 being the target, and 1 and 4 being outliers for above or below expectations) for each student in each GE course, but some students had multiple scores in a single course. Even worse, due to a glitch in the Canvas code, some students had hundreds of results, and some of those scores were not within the allowable 0-3 range.

This issue was resolved by a Python script that took in the results of the Canvas Outcome Results Report, filtered out all non-GE outcomes and non-GE courses, and created a new value for each student result row made up of a combination of the student and course's unique IDs. The script then created a list of every student/course pair, and for each pair, the script found the highest GE outcome score within the allowable range that the student received. The related course and instructor data were also added to the student score, and the cleaned and enriched outcome data was saved within another new CSV file.

Displaying Outcome and Compliance Data

The GE course information file, which held each GE course's assessment compliance status as well as the cleaned GE outcome data files, had all the data necessary to display NNU's GE course assessment compliance and transform the resulting GE data into usable information. Both goals were accomplished with Microsoft Power BI.

GE Compliance | Total Number of Courses Assessed

Tips: Hover over the grid and click on the down arrow to drill down by clicking the desired column.

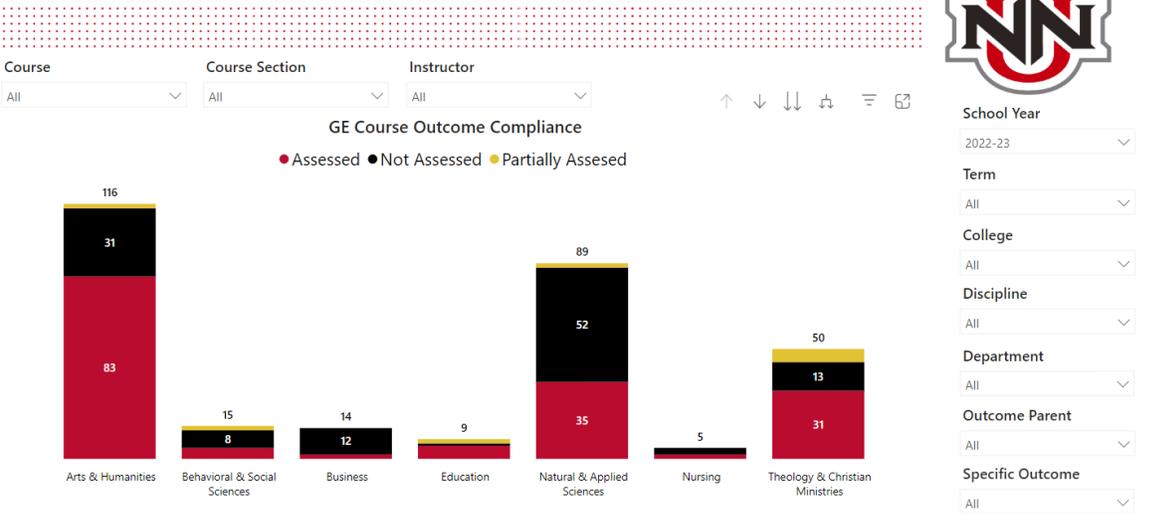


Figure 1: GE Courses Assessed

In figure one, the number of assessed, not assessed, and partially assessed courses are displayed by college for the targeted school year/s. Numerous additional filters based on the associated data collected through the previously mentioned steps are available, and viewers can also drill down within a college to view by department and then by course code.

GE Outcomes | Displayed by College, Discipline, & Department

Tips: Hover over the grid and click on the down arrow to drill down by clicking the desired column.



Figure 2: Number of GE Student Outcomes Assessed

Additionally, as shown in Figure 2 above, viewers can change their view to see the total number of GE students who enrolled in that college (or department or course code) compared to the total number of GE students assessed.

GE Compliance | Total Number of Students Assessed

Tips: Hover over the grid and click on the down arrow to drill down by clicking the desired column.

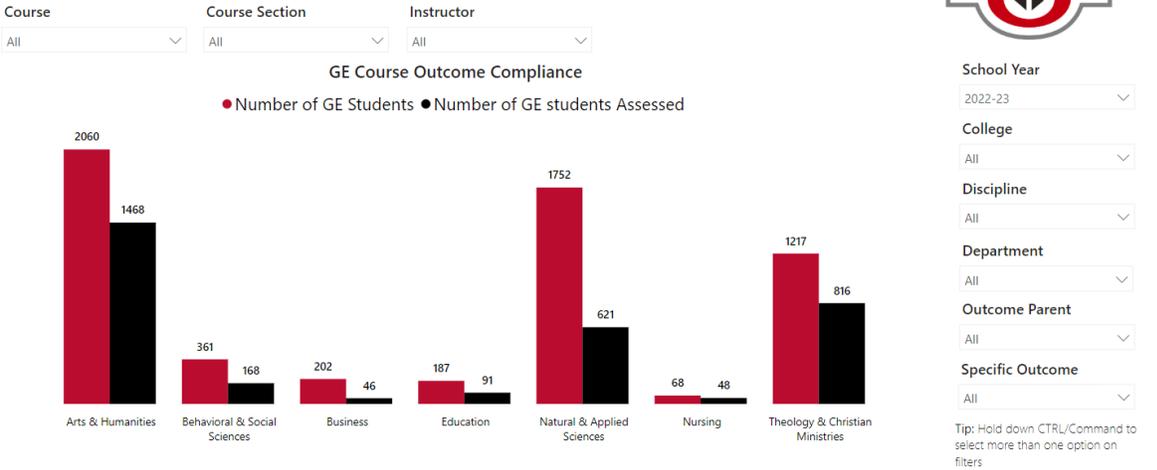


Figure 3: Average Outcome Scores

Similarly, as shown in Figure 3, Power BI was also used to take the clean outcome data and calculate/display the average outcome score on different graphs, including the following: college/department, outcome family/individual outcome, and date/outcome family. Each can be filtered by many of the same data points as those figures 2 and 3.

Resources

Throughout the project, all scripts were written for Python 3.11.4 (*Python Release Python 3.11.4*, n.d.; *Welcome to Python.Org*, 2024). They were developed in Microsoft’s Visual Studio 2022 (*Visual Studio 2022*, n.d.). Ideas and significant segments of code were generated by Chat GPT and GitHub Copilot (OpenAI, 2024; *Visual Studio With GitHub Copilot - AI Pair Programming*, n.d.). API details and starting code were retrieved from Google’s Gmail API and Instructure’s Canvas API documentation (*Canvas LMS REST API Documentation*, n.d.; *Python*

Quickstart / Gmail, n.d.). Significant Python libraries used include but are not limited to Pandas, CSV, Datetime, Threading, RE, Logging, Google, and the Google API Client (*Csv — CSV File Reading and Writing*, n.d.; *Datetime — Basic Date and Time Types*, n.d.; *Logging HOWTO*, n.d.; *Pandas - Python Data Analysis Library*, n.d.; *Re — Regular Expression Operations*, n.d.; *Threading — Thread-Based Parallelism*, n.d.; LLC, n.d.).

Results

Met Goals

Ultimately, the sixth goal of retrieving outcome artifacts (artifacts defined as samples of student work from the assignments associated with the GE outcomes) was the only goal not met. This was due to the higher complexity of selecting, retrieving, and storing the artifacts at that time such that it was decided that this was of secondary importance to perfecting the work done to meet the five other goals. These five goals were met as follows:

1. The most recent versions of GE course outcomes were imported into their associated GE courses by using the Canvas API to import a Canvas course shell containing the outcome into the GE course. A Python script determines which outcome shell/s should be imported into each GE course by using the General Education Council (GEC)'s General Education (GE) course spreadsheet in conjunction with the results of Canvas's prebuilt reports.
2. Personalized introductory emails are sent to GE instructors during a GE course's Week 0 using the Gmail API. The names, email addresses, course names, and outcome names necessary to determine who to send these emails to and to personalize the contents are retrieved from the same active GE course list that determines which outcome shell should be imported into each GE course. Reminder emails are sent on the week before an

individual course's midterms, and again the week before the course's finals, and alert emails are sent the week after the course's fines. Whether or not these emails are sent is based off each course's GE assessment compliance which is determined by other scripts. GE assessment compliance is defined while the course is ongoing as whether the GE outcome/s is/are attached to a published assignment, and it is defined after the course ends as whether there GE outcome data has been retrieved for at least 75% of the enrolled students.

3. Additional Python scripts clean and enrich the GE outcome data by filtering out all but the highest valid score for each GE student/course and adding additional instructor, outcome, and college data.
4. GE assessment compliance is determined as described above and recorded.
5. The final comma separated value (CSV) files that contain the clean GE outcome data and GE assessment compliance status data are pulled into and displayed through Microsoft Power BI.
 - a. The GE Course Compliance Power BI graph enables Deans, Chairs, the GEC, and other interested parties to determine which areas of the GE program would most benefit from focused attention to increase compliance.
 - b. Similarly, the GE Average Outcome Power BI graph displays the outcome scores to enable interested parties to see how well GE courses are succeeding in producing the relevant outcomes in the enrolled students.

Future Work

The following additional enhancements could be made in the future to further improve GE assessment tracking:

1. Write a Python script to quickly retrieve whatever number and quality (based off student outcome score) of GE outcome artifacts is required.
2. All canvas data is currently retrieved using the Canvas reports Application Programming Interface (API). This is functional, but it is not the best option due to the time it takes to generate and download the data and the fact that, when making frequent report API calls, some fail for no code-based reason. The constant use of Canvas's prebuilt reports using Python scripts creates a need for automated error handling and at times the making of repeated API calls due to the reports occasional failures. These issues could be resolved or mitigated using Canvas's Canvas Data 2 feature, which uses a single API call to retrieve the entire Canvas Database for the specified instance in a format that can be stored in a local structured query language (SQL) database. Effectively, the API calls would be replaced with local SQL calls, which are far faster, more reliable, and provide access to more data points than are included in Canvas's API.
 - a. Once the Canvas Data 2 feature is implemented, a Python script to retrieve GE outcome artifacts would be significantly simplified.
3. The entire implementation was built to target GE outcomes specifically, but it was also built to be scalable so that it can be easily expanded to other outcome areas.
 - a. The NNU Department of Undergraduate Education (Undergrad Ed) has requested that the project be expanded to include Undergraduate Education outcomes. The project has been demoed in a subsequent Undergrad Ed department meeting, and

the next step is working with their department chair to create a spreadsheet of what Undergrad Ed course codes and their associated outcomes.

4. NNU is in the process of migrating to Anthology as it's student information system (SIS). That migration has required the university to swap from the Google suite of tools to using the Microsoft suite of tools, including using Outlook for email and SharePoint and OneDrive for file sharing. This project will need to be adapted to operate within those new environments instead of Gmail and Google Drive.

Conclusion

The project was a near-total success. It will need to be maintained, and it can be improved, but at this point, it fulfills and exceeds the expectations of the requestors and stakeholders. Ultimately, this project proves the value of automation in the field of General Education by how it streamlines and reduces the work required by GE instructors and how it should increase the reliability of the resulting GE outcome data by increasing the consistency of GE course assessment.

References

- Bendixen, C., & Jacobsen, J. C. (2020). Accreditation of higher education in Denmark and European Union: From system to substance?: Quality in Higher Education. *Quality in Higher Education*, 26(1), 66–79. <https://doi.org/10.1080/13538322.2020.1729310>
- Canvas LMS REST API Documentation*. (n.d.). Retrieved April 25, 2024, from <https://canvas.instructure.com/doc/api/>
- csv—CSV File Reading and Writing*. (n.d.). Python Documentation. Retrieved April 25, 2024, from <https://docs.python.org/3/library/csv.html>
- datetime—Basic date and time types*. (n.d.). Python Documentation. Retrieved April 25, 2024, from <https://docs.python.org/3/library/datetime.html>
- Kuh, G. D., Jankowski, N., Ikenberry, S. O., & Kinzie, J. (2014). Knowing What Students Know and Can Do: E Current State of Student Learning Outcomes Assessment in U.S. Colleges and Universities. *National Institute for Learning Outcomes Assessment*.
- LLC, G. (n.d.). *google-api-python-client: Google API Client Library for Python (2.127.0)* [Python; OS Independent]. Retrieved April 25, 2024, from <https://github.com/googleapis/google-api-python-client/>
- Logging HOWTO*. (n.d.). Python Documentation. Retrieved April 25, 2024, from <https://docs.python.org/3/howto/logging.html>
- OpenAI. (2024). *ChatGPT*. <https://chat.openai.com>
- pandas—Python Data Analysis Library*. (n.d.). Retrieved April 25, 2024, from <https://pandas.pydata.org/>
- Python quickstart | Gmail*. (n.d.). Google for Developers. Retrieved April 25, 2024, from <https://developers.google.com/gmail/api/quickstart/python>

Python Release Python 3.11.4. (n.d.). Python.Org. Retrieved April 25, 2024, from <https://www.python.org/downloads/release/python-3114/>

re—Regular expression operations. (n.d.). Python Documentation. Retrieved April 25, 2024, from <https://docs.python.org/3/library/re.html>

Slemmer, D. (2024, March 22). *Accreditation report* [Personal communication].

The General Education Program / Northwest Nazarene University. (n.d.). Retrieved April 25, 2024, from <https://catalog.nnu.edu/the-general-education-program>

threading—Thread-based parallelism. (n.d.). Python Documentation. Retrieved April 25, 2024, from <https://docs.python.org/3/library/threading.html>

Visual Studio 2022. (n.d.). Visual Studio. Retrieved April 25, 2024, from <https://visualstudio.microsoft.com/vs/>

Visual Studio With GitHub Copilot—AI Pair Programming. (n.d.). Visual Studio. Retrieved April 25, 2024, from <https://visualstudio.microsoft.com/github-copilot/>

Welcome to Python.org. (2024, April 9). Python.Org. <https://www.python.org/about/>