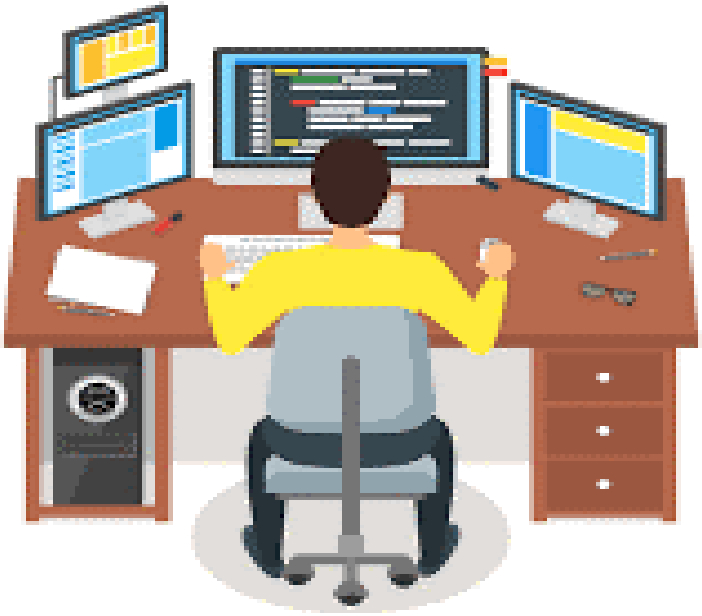
Midterm Sprint

Feb 17 – 26, 2025



- **Each project is to be completed in your groups.**

- **All projects are due by Wednesday February 26 by 11:59 pm. Each project is to be submitted through the assignment portal. Submissions are final – once you submit there is no resubmissions allowed. One member of the group should submit all projects.**

- **There are no lectures during Sprint Week. I will be on-line through Teams at 1:00 pm each day for questions or issues.**

- **I expect that Groups will have a Planning Meeting to set up the Trello Board, morning Stand-Up Meetings to go over the plan for the day and a Retrospective Meeting at the end to review the process - what did you do well, what could be improved.**

Project 1 – Python Program

This Python Project is a group of small individual programs that will be evenly divided amount group members and compiled into a single project. For each project keep the concepts of spacing, constants and comments strong. Also make each input and output screen visually appealing with headings and blank lines.

1. Enter an Employee Travel Claim.

The **NL Chocolate Company** needs a program to process salesperson travel claims when they return from a business trip. As employees return from business trips, they record all required information on a Travel Claim Form, and return the form, with all invoices, to the main office.

Allow the user to enter information from the Travel Claim Form including the employee number (must be entered and be 5 characters), employee first name (must be entered – adjust to title case), last name (must be entered – adjust to title case), location of the trip, start date (Must be entered and valid), end date (Must be entered and after the start date by no more than 7 days), a value to indicate if they used their own car, or if a car was rented (Must be entered and must be the letters O or R only). Adjust to upper case), the total kilometers traveled (Must be entered and cannot exceed 2000) - only enter the kilometers if the employee used their own car, and the claim type as standard or executive (S/E) (Must be S or E – adjust to uppercase). Only **two** validations are required.

Calculate the number of days based on the start and end dates. Calculate the per diem amount by multiplying the total days by a daily rate of 85.00 per day. The mileage amount is calculated using a rate of .17 per kilometer if the salesperson used their car, or a rate of $65.00 per day is the salesperson rented a car. The bonus is calculated based on the following:

If the number of days is more than 3 add $100.00 to the bonus. If the kilometers are over 1000 and the salesperson used their own car, add an extra 4 cents per kilometer to the bonus. If the claim type is Executive add an extra $45.00 per day to the bonus. Finally, if the start date is between Dec 15 and Dec 22 add an extra $50.00 per day to the bonus.

The Claim Amount is calculated as the per diem amount, the mileage amount, and the bonus. The HST is calculated on the Claim Amount using a rate of 15%. The Claim Total is the Claim Amount plus the HST.

The program will display all input and calculated values to the screen as results. Only display the mileage amount if it is calculated. Just do a basic printout with headings and formatted values.

Repeat the program until the user enters a termination value either on the first input, or a prompt at the end of the claim output.

2. Fun Interview Question

A common program used at interviews for programming position is the FizzBiss problem.

Create a loop to execute 100 times. For each value if the number is divisible by 5 display the word Fizz. If the value is divisible by 8 display the word Buzz. If the value is divisible by both 5 and 8 display the word FizzBizz – be careful of the order of the if's in this problem. Otherwise display just the number. A sample of the output is shown.

```
1
2
3
4
Fizz
6
7
Bizz
:
:
38
39
FizzBizz
41
:
```

3. Cool Stuff with Strings and Dates

Keeping with the theme of the program, create a program that does something cool with strings and dates based on the employee first name, last name, phone number, current date, the employee's start date, and the employee's birthdate. You can input the values or just assign values to variables.

Some examples might be to build an Employee ID, Username or Password. Determine how long the employee is working with us, how long till retirement – assuming 65 years is the retirement age, how long till their birthday, etc. Include at least 5 – 6 results based on string or date processing.

4. A Little Bit of Everything

Write the program based on the guidelines below. Add validations, formatting, and a well-designed output. XYZ Company is setting up a maintenance schedule for major pieces of equipment. Allow the user to enter the cost and the purchase date. They must perform basic cleaning in 10 days, tube and fluid checks in 3 weeks, and a major inspection in 6 months. Determine each date in the maintenance schedule. Determine the monthly amortization for the equipment based on a useful life of 15 years (180 months) and a salvage value based on 10% of the purchase cost. The formula to calculate the monthly amortization is **amortization = (Cost – Salvage value) / Number of months.** Display all input and calculated results in a well formatted output.

5. Something Old, Something New

Research something new in Python – could be a new library or process. Show how that topic is used by setting up an example. Include comments in the code to tell me what you are doing and print statements in the program to tell the user what is going on. Please do not to include anything that must be installed – it makes it more difficult on my side.

6. Set up a menu-based system to run the different programs.

A bit of fun!! Prepare the basic features for a menu that will be used to run each of the programs created in Parts 1 – 5 above. The menu will appear as follow. Based on the selection, print a message that reads "This is option X from the menu." – you can place the code for each program in the if as well – but for now just the print is fine. Place the menu in a loop to continually repeat until the user selects option 6. NOTE: The full processing of the menu will be covered when we start back after the midterm sprint.

Midterm Sprint - Main Menu

1. Complete a travel claim.
2. Fun interview question.
3. Cool stuff with strings and dates.
4. A little bit of everything.
5. Something old, something new.
6. Quit.

Enter choice (1-6):

Project 2 – Technical Writing

Each member of the group is to prepare an instruction sheet for one of the options in the Python program completed above. Normally these pages should be consistent, but in this case each of you may format your page as you wish to show your style and creativity. **Columns, pictures, and cool features are nice.**

Instruction Sheet Evaluation Criteria:

1. **Content & Writing Style:**
    - **Use simple yet clear language, ensuring users understand the instructions without oversimplification.**
    - **Language must match the target audience (e.g., avoid overly basic terms unless the manual is for children or beginners).**
2. **Layout:**
    - **Columns: Break up long text using columns if applicable.**
    - **Alignment & Formatting: Ensure uniform alignment and formatting throughout.**
    - **Font Size & Style: Use consistent fonts and sizes for headings and body text.**
3. **Visual Aids:**
    - **Use relevant screenshots, images, graphs, and tables to support instructions.**
    - **Visuals must match the steps or sections they explain.**
4. **Headings:**
    - **Apply proper and hierarchical headings for easy navigation.**
5. **File Format:**
    - **Format the manual in Word and save it as a PDF before submission.**

Project 3 – Bash – Prepare the following in a document.

Write the command(s) to complete each of the following in a document. You can test in Bash to check that everything is OK – but all I want is the document at the end.

- Create 3 files – one called **PySample.py**, one called **ExtraP.txt**, and a third called **Stuff.py**. Add code to each python program and add some text to txt file. Add the words **"RetailCost"** and **"getName"** in one or more of these files.

- What is the current directory? List the contents of the current directory. List the contents of the current directory with any hidden files. List the contents of the current directory with any hidden files. List the files with the permissions displayed.

- Display the full contents of each of the files created above. Display only the first 3 lines of **PySample.py** and the last 3 lines of **Stuff.py**. Display the last 3 lines of all files. If you have a file that is large, what options are available to have it appear one screen at a time?

- Create a directory called **PythonCurrent**, one called **PythonDone** and one called **TextFiles**. In the PythonDone directory create 2 other directories called **2022** and **2023**. In the **TextFiles** directory create 2 other directories called **Python** and **Other**.

- Change the current directory to **PythonDone – notice how the prompt changes to show the current directory**. Use pwd to confirm you are in the PythonDone directory. List the files.

- Change to the following directories: **PythonCurrent**, **2023**, and **Other**. Prove that you are in the proper directory. Go back to your working directory.

- Move the **Stuff.py** file to the **PythonCurrent** folder and give it the same name. Check and make sure the file has been moved. Copy the **ExtraP.txt** file to the **Python** folder in **TextFiles**.

- Change to the **Python** folder and display the first 6 lines of the file **ExtraP.txt**. Once complete move back to the main folder.

- Display the contents of the **ExtraP.txt** file from the Python directory in **PythonDone** directory from the current location – you should be in the working directory.

- From the working directory, find the file called **Stuff.py** searching all subdirectories.

- Change the permissions in the PySample.py so that the owner, group, and everyone else has only read and execute permission.

- Find the text **getName** in a file and indicate which file(s) it is located –search all subdirectories. Do the same with the word **RetailCost** – in this case allow the search by ignoring case. Direct the output for one of these to a file called **FindResults.txt**. How could you send both results to the same file without overwriting it?

- Perform one other command that you feel would be useful. Write it down with an explanation indicating what it is doing and why you feel it would be beneficial.

Bash Part Evaluation Criteria:

1. **Commands:**
    - **Use accurate Bash commands with screenshots of the outputs.**
2. **Formatting:**
    - **Organize commands in a well-structured Word document and save it as a PDF before submission.**

Project 4 – Presentation – to be presented on Mon, Feb 24 @ 12 pm.

Each member of the group – not just one topic for the entire group - must find an article using an IT News site like https://www.pcmag.com/news, https://www.zdnet.com/, or https://www.computerworld.com/news/. Once you find your article, do a bit of research to find more information about the topic selected. I would strongly suggest that you find an IT news site you like and follow it regularly.

Create 2 - 3 slides based on your article and the research that you did. The first slide should summarize your article and the other 2 slides should be based on additional information that you researched.

Put all the slides from your group together with a consistent format. The first slide will be a title, the second slide will introduce the group members and their topics, and the remaining slides will be the ones you each prepared about your topic.

On Monday afternoon at 12 pm you will present your findings to the class. One member of the group will introduce the first two slides, then each member of the group should present their slides. The presentation should last approximately 5 minutes for each group.

Your presentation is to be included as part of your final submission.

Presentation Evaluation Criteria:

1. **Formatting & Visuals:**
    - **Maintain consistent formatting (fonts, colors, layouts) across all slides.**
    - **Avoid excessive text; keep slides concise.**
2. **Content Delivery:**
    - **Present with confidence and clarity.**
    - **Use relevant facts, statistics, and details to support points.**
3. **Design:**
    - **Incorporate visually appealing designs with effective use of images, charts, and graphics.**
4. **Structure & Organization:**
    - **Ensure a clear presentation structure (beginning, middle, end) with smooth transitions.**
5. **Engagement:**
    - **Engage the audience effectively and handle the Q&A session with confidence.**

Project 5 – Web Design

You are to design and create a web site on the topic of your choice. The site is required to contain the main index page and 2 other pages based on links in the navigation. Your site will be graded for **originality** and **design consistency**, as well as efficient use of the following:

- **Page Layout –** make the layout easy to read and navigate. Keep the KISS rule in mind – Keep It Simple Stupid. Make proper use of semantic tags and style each section appropriately.

- **CSS –** Create the page layout using CSS and the Box Model. Use of margins, borders and padding as required for attractive and effective design. Common styles and classes – especially text and links – are to be defined in CSS file and used for each page in your site.

- **Text formatting** – fonts, sizes, colors, bold, italic, underline, etc.

- **Use of colors** - for backgrounds and text.

- **Use of links** – for navigation on the top of each page, at least two (2) links to related external sites, and Top of Page links for pages that exceed the browser screen.

- **Use of images** – at least five (5) related images, not including those on the index page.

- **Use of lists** – at least two lists, ordered or unordered.

- Based on your navigation links, **create pages for the links** to your site – one for each member of the group – one member should complete the index, and each member of the group should prepare one of the pages based on the main navigation using the template. Keep all pages consistent – generally the other pages are the same as the index page, and the main content is changed to match the option you are working on.

  I suggest working on the template as a Group, then each group member create their pages based on the template.

You may create a banner for your site using any program you wish – but it must be saved as a .jpg file. The remainder of this site is to be completed using **HTML / CSS only**. Note that your site will be graded using Google Chrome.

HINT – I love looking up examples of sites using google and searching for Website Templates. You can even do a search on a similar topic for some website ideas. There are thousands of examples that you can use for ideas for your page.