**2024-05-01**

Did some initial onboarding, mainly signing standard tax documents, company agreements and policy reviews. The secretary also gave me a tour of the office, introduced me to all the employees currently in the office and showed me where I would be working. I worked with Kieran a little bit observing some database work, tracking functionalities. Then my supervisor who was not in the office invited me to a meeting to expose me to the data historian software that tracks data trends and allows users to make custom dashboards for tracking specific data sets. Although I do not completely understand what is happening regarding the software it was cool to see how the software makes data easier to read and track.

**2024-05-02**

Today, Sean was in the office, but he was on a call with a site that had a server failure, so he spent a lot of time working with them to get their server back in working order. He had me read some material about the historian software I'd be using to add reports from a different software so that the company that employed us could stop using it. In a nutshell the software is just making SQL queries from a database and displaying them nicely in a similar way to Excel. Afterwards we figured out how much access I had on the remote server housing the data I would need to access. Once we got access, Sean had me compile a list of all the important reports from the old software into an Excel spreadsheet so that they could be cross referenced once I started generating the new ones for the new software. Tomorrow, I'll be tuning in to the training session Sean is hosting for the employees that will be using the software so I can get some. experience using the software myself.

**2024-05-03**

Today opened by entering a training meeting like the one on my first day. This one was a bit lengthier and ran for about four hours. The difference this time around was that I got to follow Sean along and interact with the software. The software as mentioned before is a glorified visualizer of SQL queries made on a database. It also allows for reports generation and manual data entry from users. Overall, it is a comprehensive piece of software that seems to do its job well. After the meeting we had a short meeting with a couple of others from the company employing us, discussing how the training session went overall and some things they could improve the software later down the line. I spent the rest of the day starting on a report transfer to the new system for the company. I chose to start with a big report so that I could get a lot of practice while still only focusing on one report. Didn't get a chance to test whether I built the report correctly or not, so I'll have to take it up again on Monday.

**2024-05-06**

Once again, I am chugging through reports. I made it through a couple larger ones today. Although I initially spent the morning trouble shooting three, I chose to work on since they weren't displaying values. After discussing with Sean, we decided I should just move on and keep going through the reports and anything that isn't working we can fix later. Eventually some of the reports I continued to make did work without me changing anything in my process, which leads me to believe that it is an issue with the data and not with how I'm compiling the report. The old reports which use a less efficient table for data retrieval allows for a footer where the previous report makers have been adding averages for values displayed in the table currently instead of just making a new section of the table. I don't / haven't figured out if there is a way to do this with the table from the template, I am using but I figured I would ask Sean next time I get the chance. There's still a lot to go through so it will likely be a couple more weeks before all the report transferring gets completed.

**2024-05-07**

Back to the report grind, it seems a lot of the employer’s previous reports were broken due to a variety of reasons like weird data entry. All in all, still chugging through. This afternoon I mostly worked on an excel sheet that Sean had made up as a template for quick data retrieval from the database. Basically, they just want a quick reference. It works almost identically to the reports I’ve been making just in excel instead. This took up pretty much my whole day. Sean is going to use the excel sheet I compiled in his training sessions tomorrow.

**2024-05-08**

Tuned in to another one of Sean's training sessions. This one was a bit of a slog, 4 hours long total. However, this session was with a new group of people and this group is much more engaged than the previous. They were asking questions following along well which made the 4 hours pass a bit quicker. I mostly continued with reports during this time, so the meeting served as background noise. Initially the company who employed us wanted this new data system to go live from the end of May to the beginning of June. However, the lead on this project at our employer said that the yearly shutdown they do in September may be moved to June instead. This would give us a little more time to flesh out the system and polish some aspects. We are still operating as if the timeline is the same but will take advantage of the extra time if the decision to change the shutdown ends up being made. Afterwards, I was also in another of Sean's training sessions, but this one was only an hour. It was mostly showing off the trend viewer client app, query tool and excel template that I made yesterday. This session went well too, and Sean got a lot of positive feedback which he appreciated.

**2024-05-09**

Came into the office earlier today. It was quiet today; it seems that not a lot of people were here at the office. Maybe they were working online at home? Anyway, my boss Sean was on site today doing a training session instead of the usual online teaching sessions. Either way I was still chewing through reports, and I got a good chunk done today. Towards the latter end of the day, Sean had me start fixing the engineering units used by the system. For some reason, there is a bunch of the same units just capitalized differently but meaning the same thing. So, I'm currently going through and adjusting all of those to reach an agreement on a standard with the operations team. I also learned that a ton and a tonne are not the same thing. One of them is technically a metric unit but not recognized by SI. Also, there is a short and a long ton which have about 100 kg difference in measurement for some reason. All in all, a success full day.

**2024-05-09**

This morning leads off with an advanced report training. This initial group was less interesting than yesterday’s. They did not interact as much as the last group. This session basically just covered charts in the report studio. The charts are like Excel charts but with a little more convoluted feature. This training went smoothly for the most part, sometimes the report studio has errors when you try and make object near the edge of the report page causes you to use some less than convenient work arounds. New to this training session, Sean let me teach the report template section. I was nervous at first but eventually just focused on doing a good job and forgot about being nervous. Post training meeting both the lead at the company employing us and Sean complimented how clearly and thoroughly I was explaining the report template features, which made me feel a lot more confident in my ability. Then for the rest of the day I worked on fleshing out the engineering unit standardization I was working on yesterday and finished up the document and gave it to Sean. Researching symbols and standards for units made me realize how inconsistent they can be sometimes despite having a standard. Regardless I finish out the last hour of the day by doing some more reports and heading home for the weekend.

**2024-05-13**

Today I did some more reports for half the day, a good old 4 hours. We were focusing on manual entry boxes which allow the user to update data via the web interface itself. This has a little more complexity than the simple objects we were working with before, they also have some styling issues when placed on tables. Most of the 4 hours were spent dealing with the styling issues, which aren't fully resolved yet. Progress is steady. Next, I tuned into Sean's OMI training session, which basically has a visual of the systems being used on site. The system used to make them is extremely like object-oriented programming. You start with a base object that has a very general outline of how you want all the objects to be, then the specific objects inherit the outline of the base object, which is effectively a polymorphism. Also, like objects, they have functions which are akin to methods and properties which give them unique attributes. Then Sean tasked me with going through all the float tags and setting the min and max to be shown in trends which I am using python to complete since the data set is quite large. I will carry on tomorrow.

**2024-05-14**

Today flew by, despite not working on much of a variety of things. I got to use some more python today to filter values in a spreadsheet. I'm liking using python for these large data sets (number of rows > 1000) because it allows me to abstract and containerize functions rather than having a huge formula bar in excel. It also allows us to fine tune the functionality of the filtering and manipulation as precisely as needed. It has also taught me how to do some spreadsheet manipulation using the openpyxl module. I also learned a bit more regular expression pattern matching with Excel does not natively support. I was mostly independent today since Sean was on site doing in person training sessions. Overall, a productively feeling day that just flew by.

**2024-05-15**

Today we took another big step forward to the go live for the new system for our employer. Since most of the reports are finished and compiled, we now must vet their functionality. This morning Sean asked me to make a sheet to track and test all the reports document individual issues with each of their tables or views. This is pretty much going through all the reports again, like I have been since starting. I'm quite proud of the sheet I made. It works decently well and modular which will allow for updates. It seems like a lot of the tags have broken values or the way they are being queried is incorrect. I say this because a lot of the tables have values displaying "N/A" or "Err" meaning no data or there are null values respectively. This is a problem likely with the tag itself, somewhere along the retrieval line, there could be a typo with the tag or something along those lines. For now, we're just taking note of them and then going to return and amend them later. Hopefully, when we go back to fix them, the fixes will be as simple as fixing a type (although it's \*\*never\*\* usually that easy). So, another long but relatively simple task that will likely close out the week for me.

**2024-05-16**

Mostly report verification today, going through the standard tests and verifying everything is in working order. Made it through a good junk report today. However, it is a little annoying because Sean is doing training sessions at the same time I'm using the web view for the reports. Since he is doing training, every so often he must reload the whole project containing all the reports to update changes he makes to his examples. This inadvertently caused me to be logged out of the online portal due to the web view being updated. This causes small interruptions when I'm trying to be efficient, having to log back in causing some productivity halting. But today, a bigger interruption was experienced. Towards the afternoon, the whole project housing the reports crashed and I was no longer able to get onto the report studio software. I got in contact with the IT department for our employer and he basically said he had to go physically to find out what was going on. Later, Sean messaged me saying that the whole server needed to be restarted before I could continue. So pretty much no work got done for the last two hours of today.

**2024-05-17**

Today was more report validation. Sean came into the office today, so it was nice to have some company. He did spend most of his day in meetings (which seems to be regular for him). Report validation went smoothly today except for when I would reach reports with upwards of 20 tables. At that point it got a little tedious expanding the excel spreadsheet that I made to accommodate these large group of rows. Still haven't really figured out why some tables on the new system display data when they didn't on the old system, and vice versa. Likely due to some sort of query or source data issue that we haven't solved yet but will need to solve in the future. In one of Sean's meetings, they talked about how since the shutdown we might upgrade all their systems in place. Either way I'm still going to Sherritt next week to get some hands-on experience. I'll report with what I learn next week.

**2024-05-21**

First day on site. This is exciting because coming into this job I didn't really expect to be able to go to the sites of the companies I'd be working for, so this feels cool. It was also nice to meet the faces of those I had spent training sessions over teams with and really get a feel for the people on the project. It was pretty refreshing to understand how frustrated they are with their old historian system and really want something new, robust and effective. They expressed how they've been banging their heads against the wall and dying for this new system but there were a lot of factors preventing them from reaching this goal. It is a bit odd though because they did have this system in place originally but the way it was configured by a different integrator caused data retrieval to be slow and optimized or nonfunctional. This led them to just use their old system, but the issue is that the previous supporting company behind this old system no longer does business with them, so if anything were to happen to that system, they would be in the hole. Apparently, the company I'm working for never got wind of wanting to revamp their system so never got a bid for the work. Then we arrive in the modern day where we are now fixing the long trail of mistakes and it seems like those on the project are finally happy with the new system despite it not even being fully ready for use. Tomorrow is day 2 of the training so I'll be back on site.

**2024-05-23**

Back at the office today. Sean didn't let me come to the site today because he said he rather me get through some more report verification. So, get through the verification I did. I have less than 10 reports to verify before I try and go through and fix them. Most of the issues with the reports seem to be simple so hopefully the fixes will also be simple as well. This was kind of mind numbing to do today but it was nice to be back in the office and socialize a little bit. On a brighter note, I will get a couple overtime hours since I was at the site for over 8 hours. Overall, it was cool to be on site, but less efficient in terms of work done. Despite the size of the company at the site a whole lot of nothing also happens there. It also brought to light the disconnect between those running a company and those leading the groups that ultimately allow the company to stay afloat. This made me appreciate working for this smaller company where management is very aware of what is going on at the company.

**2024-05-24**

Last day of another week. Once again, some more report validation. I rounded out verifying all the issues with the reports which means it was time to go through and start fixing them. I started with easy fixes like formatting or super simple functional changes. This didn't take a lot of time and really minimized the amount of time that I thought this process would take. Then as a break I fiddled around with the engineering units as Sean sent me another list to try and fill out the master list. Unfortunately, there are still a lot of missing ranges, as not all these lists are complete it seems as if even manually going through the lists to find the incomplete ranges, they do not appear in any of the files I was given. With that out of the way, I went back to verification. Earlier in the week, Sean had mentioned to me that there is a list he had of tags that are dead, meaning they are no longer providing data to the historian. The list is long so before I go and try and fix all the reports that have data retrieval issues. This would stop me from banging my head against the wall when I can't figure out why the reports aren't displaying data. Using some python wizardry with an Excel manipulation library, I was able to write a script that gets all the tags in all the transferred reports, then gets all the tags labeled "dead" or "old" then find the matches between the two. It was quite a time save and satisfying to get this working. It seems it's easier to do pattern matching with string in python than in Excel itself.

**2024-05-27**

Today I started by continuing what I had begun last week, finding dead tags. I was tasked with finding the dead tags in reports if they exist. As previously mentioned, I used python to find the dead tags that were in use, which saved a lot of time. However, I made a sheet that tracks which tags are dead, what table they are in and what report that table belongs to. This was painfully boring and tedious because due to how we set up the tracking for what tags go in what tables when the transfers were made. Even using python this would prove to be a task way too lengthy, ultimately, I ended up doing it manually, boring but more efficient than either of the two methods. Then after that I made a new column in the acceptance testing sheet which identifies the dead tags in each table for each report. Afterwards, I updated the notes of all the reports with dead tags to mention that they have one. Now, the remaining problematic tables that do not have dead tags need to be dealt with. Sean informed me this is likely due to a driver issue, mainly the integral driver to improper math, causing an error thus no values being displayed. He was busy this afternoon so he couldn't help me set up or change the drivers that need fixing, but we can get to that tomorrow. These drivers do math in the background when pulling data from the historian given specific parameters. This is how we dynamically display data and specific units for data. Basically, the data gets queried from SQL Server then the appropriate math operations get performed on all the data pulled from the query. Tomorrow, the remaining driver issues will be tackled.

**2024-05-28**

Just some more, report verification for today. Except Sean and I did a little bit of a deep dive into some of the tags that are causing us issues. Some of the tags were producing weird results and 0s where there shouldn't be. So, we managed to figure out why the initial one was due to a miscalculation involving engineering units. Basically, since the engineering units have been messed up from the start, this is a fundamental way that the AVEVA software calculates specific values, especially flows which are just calculated using integration. This gives wildly different values despite the values being logged to be used in the calculation are the same on both sides. A little annoying but not a big issue. Next, we came across the fact that the last integrator just copied things over without really checking or fixing anything. This means that an undetermined number of reports could just have wrong drivers causing the wrong type of data to be displayed in the first place. Again, annoying but not the worst. But by far the worst thing we came across today is that whoever chose the name for tags did a terrible job. Apparently, there are a bunch of manual entry tags that only point to the old system and not the new one which is fine cause some of them end in "ME" for "Manual Entry", clearly identifying them. Which would be great, if they were all like that. The vast majority of about 500 or so tags don't have any indication that they are manual entry tags. Once again, they show data on the old system leading you to believe that you messed something up or something is wrong with the tag if don't recognize that it is a manual entry tag. Every day I'm reminded as to how poor a job the last integrator has done. Its problems are built on the foundation of more problems. But everyday Sean and I attempt to solve more and more of them, which is nice.

**2024-05-29**

Today was problematic. First, report verification was going well. With the information Sean and I uncovered yesterday going through reports was much quicker and easier. There were still a couple problems that I could not resolve, such as data mismatch when all the parameters are correct on both reports. This is likely due to how the data is being queried before even reaching the report. These will likely be harder to trouble shoot down the line as the solution will be more completed than meets the eye. But that was just the start. Later into the afternoon Sean asked me to hop off the server so he could implement the new dev server for the project. They've been wanting to separate the servers so issues with multiple users developing the software don't arise like before. This great because it allows for versioning of the project as well, so if someone implements something wrong or some update causes the project to break and isn't recoverable from its current state, we can go back to a previous version and not lose all the data. This is a common form of version control like Git, which is the gold standard when it comes to version control for programming. This software's version control is exactly like Git in the sense of commits which are when you make changes to the project and want to "save" those changes to the project on the remote machine. This is what essentially is kept as a record for previous versions. We brainstormed and decided that we should probably not let everyone automatically commit to the project and have admins control whether a commit is acceptable or not before is pushed remotely. Also, one of the dialogs for viewing the version history has the "revert to previous version" button right next to the close button, which could be a dangerous source of miss clicks. We fiddled around with some of the settings and then I went to carry on my verification. However, to my surprise, all the reports on the online version were no longer there and I had a connection issue. This prevented me from doing report verification as I need to compare values from the new and old reports and this portal is what allows me to do so. So, I was stuck doing nothing for the latter half of the day.

**2024-05-30**

Today was more report verification. Sean came in today to deal with some calls and set up a PC for another client. The report verification was going well but it got to the point where I would be alt tabbing and control f’ing between multiple documents which got quite tedious. So, like I usually do, I made a script to handle all this. Basically, the script gets the old system tag name, looks its new system name then finds if the tag is in the list of dead tags, LIMS tags or manual entry tags. This sped me up quite a bit of clicking but made the process much faster. I think this is where my programming skills have really been shining at this job. I may not be doing much programming, but I get to make my workflow easier with what knowledge I have, which is fine for me for now. Otherwise, it was business as usual. Sean also told me that the company we are doing work for wants a round 2 of training for operators. This is a big undertaking as they’re about 80 operators that would need to be trained. This gives us a lot more work, probably more than when I was brought on to handle this project. That's fine with me if it doesn't start bleeding into my day-to-day life outside of work as it sounds like for some guys that work here it does.

**2024-05-31**

More report verification, except I'm back to being alone in my office. I felt motivated today, probably because it's Friday and I know if a power through work time will come quicker, and the weekend will come sooner. No problems today oddly enough which made today a little boring. Never thought I'd be asking for more problems, but here we are. Verification is going smoothly, and I've streamlined a workflow that works for me. Basically, I load up both versions of the report and load the same report in the editor. Then I see how the new version data being generated looks compared to the old version of the report. If the data does not match, I use the script I made to get the new system tag name using the old system tag name, then check the tag in query to see if the values there match the old report. If they match the new one and not the old one that means that the issues lie out of my control and somewhere along the line, the value is changing for a reason unknown to me. If the values do match, I then go to the report editor and make sure that everything is correct, this includes drivers, tag names etc. I also mainly check if the tag is dead or a manual entry. If the tag is dead, it usually just puts out null values that the system doesn't know how to read. The manual entry tags tend to just output the same value repeatedly for some reason because they are only connected to the old system and not the new one quite yet, which explains the discrepancy in data between the report version most of the time. If I can't solve or figure out why the problem is happening, I usually bring it up to Sean and see if he can help. Often though, it is something out of our control. I will pick up verification again next week.

**2024-06-03**

More report verification. Initially I was dreading the fact that I was eventually going to have to tackle a very large report with around 45 tables. However, once I started going through each of the tables today, I found that the changes were very minor. First, a lot of the tags for the tables that weren't working were manual entries which makes sense why they weren't working. A nice and simple note about that on the spreadsheet and then the changes are fixed and made. The other big issue I ran into was when I was initially reviewing the reports, I must've missed the fact that the time range for the report was different than the one on the old system, hence mismatched values when I was comparing the output of the new report. This made the bulk verification of multiple tables extremely easy, as it was glance at the table, check if it's wrong, if not great, else dig deeper. I'm nearing the end of the verification now so hopefully not too much longer. I also heard the higher ups discussing moving me to some new other stuff, so I get a little more hands with actual PLC programming and related technologies.

**2024-06-04**

Almost done report verification!! It has been a long time coming but I'm almost through all the values matching and afterwards it some easy cosmetic cleanup and then the reports should be ready to be pushed to production for the trained users to access. This is pretty much what I've been working on since I started at the company so it's nice to see the effort pay off with compliments from the company that employed us. But more than anything, I'm excited to maybe move to something new as I have been working on this for a while and there isn't much thought involved once you understand the core functionality of the different programs. The office is empty today and a lot of the people I'm used to seeing don't seem to be around today and I don't really know why but maybe they are just working as others seem to be doing. I personally prefer being in the office as it switches me into the working mindset, much like school. Although, I think it could be nice to work from home every once and awhile if I really don't feel like going into the office. But I'm not sure I can ask for that just yet. Tomorrow hopefully can wrap up report verification.

**2024-06-05**

Well, finally wrapped up report verification. It's been a long time coming and even Sean was very excited for me. This was a major chunk of work that needed to be done for the company that employed us. Now that Sean is tackling the engineering unit side of things, there wasn't a whole lot left for me to do. Then Sean had me set up a PC for some sort of crane that isn't built now so that was pretty fun. It was a simple task overall, the pc already windows it just needed a new local user, the workgroup changed and some other standard config options. Then he had me transfer the iso files for the software suite the pc needs from the company server. This took a bit as the files were compressed but still not small. Once the files were transferred Sean helped me configure the AVEVA software suite components. But we ran out of time as was shift was quickly approaching its end, so we decided to pause and wait till tomorrow.

**2024-06-06**

Taking the pc setup back up again today, Sean walked me through installing the correct components for the AVEVA suite of application this pc needs. It was basically just clicking some checkboxes and then waiting patiently. After about an hour everything that needed to be set up was done, Sean said he'd finish up for me when he comes in tomorrow. With that I was truly left with nothing to do. So, then Sean suggested I contact Matt and see if he has anything else for me to do. Within a couple minutes Matt came to my office and explained to me something that I would be doing next. Basically, I need to make access level changes to this HMI (Human Machine Interface) so that based on where it is being accessed the user will have different permissions and can only change certain things on the HMI. These changes need to be made to software I have never used before. It's basically a front-end piece of software that makes actions on the HMI more understandable for everyday operators. This was a lot to take in. Matt had me read the security outline and the changelog for the HMI and how they wanted it configured. Basically, telling me I need to document my every move. Oh boy this is where it starts. So far, the work is quite primitive. However, now this is where the real learning begins, the whole reason I decided to do a coop. Matt assured me I could ask him questions, but my questions would probably be best given to Jeff, who has around 16 years of experience with this piece of software. I didn't really get to make any changes today, but I will start tomorrow.

**2024-06-07**

Today was cool, I finally got to start working on some HMI stuff. Basically, the HMI is the interface operators and users at a site or plant will use to interact with systems there. HMIs basically integrate a front-end GUI with the backend tags that control machinery at the plant. Since Matt got me set up yesterday with HMI stuff, he pointed me to Jeff who has a lot of experience with this piece of HMI software called AVEVA InTouch. It allows for complex logic involving multiple different systems and interfaces and those systems can be linked to interactable graphics. Custom graphics can also be used and configured with custom properties for super advanced customization. All the work I'm doing is in a VM because it allows the HMI and IDE to be containerized so that the host system doesn't conflict with any of the configurations for the HMI. Basically, right now I'm just following the changelog for the last HMI that was reconfigured appropriately for different access levels. During the afternoon, Sean brought in a big amount of food for lunch including wings, brisket, salad and the like. This basically made the afternoon a right off. This was a lot of fun and let me get to know others at the company a little better.

**2024-06-10**

Continuing this Monday with more HMI config. I'm slowly learning the interface for the HMI more and more and getting faster at navigating. Kieran has also been a big help since he's relatively more familiar with the software and he taught me some useful key bindings that have saved me some time. Then, Jeff and I ran into an important problem. Basically, there are many elements that need to have a custom property changed, but unfortunately, there doesn't seem to be a way to mass modify an object and have it reflect across the objects already in the HMI. So, we had to result to this procedure made by one of Jeff's coworkers, Shane. The procedure isn't exactly the best and didn't really seem to work for modifying all the properties on an object across a whole HMI. This didn't work for me so Jeff said he would try it. After numerous attempts, Jeff could not get it to work also. He also expressed his frustration with the procedure, especially since Shane won't be in till Wednesday this week. Despite this, I kept moving along, making all the changes I could. I will pick it up again tomorrow.

**2024-06-11**

More HMI stuff today. I made my way through the one HMI I have been working on since Friday smoothly, barely having to ask Jeff any questions. It's nice to be self-sufficient and just get work done smoothly, troubleshooting isn't usually fun for most people. The more annoying part of this task is the documentation for sure. Basically, I must record all changes made on the HMI so here is a record. This consists of a screenshot of fixed 1500-pixel width for some reason, a red border around the change on the screenshot for an addition, or hard change, a black border for thing being left as is. This is a tedious process having to be done for every individual change, but it could be worse. I also have some qualms with the InTouch HMI software. The selection of items is done with the mouse (yuck) and like to move stuff if you simply use left click to select the items. This is frustrating and can be hard to reverse. Therefore, it is safer to use shift click which doesn't move the object and just selects it. Then you can edit the properties as you wish. However, my biggest gripe is that there is not a key for deselecting the object, you must shift clicking again. This is just too much for me. I much prefer using hotkeys to navigate and do actions rather than the mouse but alas. Luckily there are some nice hotkeys for common activities and menus which makes some property editing easier. Once I wrapped up this HMI, Jeff instructed me to put it on the server in the appropriate folder. However, the folder has disappeared from the file explorer on my system. Odd situation, but the same thing happened to another team. Basically, I was just told to wait and see if it comes back. With that info I move to the next HMI.

# 2024-06-12

Sick day :(. No work was done.

**2024-06-13**

Well after a brutal sick day, I'm back at the office continuing with HMI stuff. Today I called Shane, who seems to be leading this HMI stuff. He instructed me to update the graphics with the correct properties like Jeff and I were trying to do earlier in the week. He did them for both HMI's so now it was up to me to go through and fix some things. He told me that I now needed to go change the default values for all the newly updated graphics with the changed custom properties. This involved going into each window and manually changing each value of the graphic, multiple times if there are many graphics. This proved to be lengthy as the graphics we changed are used for alarms and a lot of the windows have alarms that contain the graphic. Then he said once I'm done with that I can go through and change the window properties of every popup type window to have +10 height and width then enable size controls. This took even longer as there are windows labeled clearly as popups, and those that aren't. This resulted in me having to go through all windows just to verify that they weren't popup windows with a title bar. This took a couple of hours and made up most of my day. After I finished my documentation and those two lengthy tasks, it was time to move to the next HMI. Guess what I had to do for this one? The same thing all over again! So, on top of those previous two lengthy tasks a whole new set of documentation and all the other more minor changes I made to the previous HMI had to be made to the current HMI. Honestly the thing I'm concerned about the most is the ergonomics of my setup. The software being used doesn't have the most ergonomic keyboard shortcuts on top of me having to do the same keystrokes numerous times, ergonomics are becoming important. My friend Kieran suggested I could ask for a better keyboard, which in that case I would opt for a slightly raised and split keyboard as I've been wanting to try one for a while and be able to have the two halves of the keyboard at whatever distance I felt would be good for me. I am also interested in thumb clusters as the thumbs are only really used to press the spacebar. Thumb clusters could offer macros which would be useful for awkward shortcuts and modifier keys like control or shift thus having the rest of the fingers only pressing letter, number or punctuation keys. I pick up where I left off tomorrow.

**2024-06-14**

Continuing with HMI stuff basically today was spent chugging through the remainder of the changes from the HMI release notes and wrapping any minor things that needed to be added. This new HMI I am currently working on is very similar to the previous one, so the changes went significantly quicker as I am familiar with the software and how the HMIs are laid out. It feels nice to be comfortable with the software and knowing that the only thing slowing me down is my energy level and my ability to stay focused. I wrapped up the changes by noon, ate my lunch and then asked around for more work. Both Sean and Shane said they had nothing else for me to do, so then I sent Matt a message, assuming he would likely have more for me to do. I sent him a message, but then I realized he was in a meeting and wouldn't be free until about 1pm. In the meantime, I read my book that I left at work about automating tasks with python as it has provided useful information about automating workflows with excel spreadsheets and saved me a ton of working time. I usually read it when there are lulls between tasks like this one. 1pm rolls around and I receive a message from Matt. His message tells me that he doesn't have anything for me to do for the rest of the day and won't have anything until next week Monday, he then tells me I can go home for the weekend early! Yippee! Overall, a great day. I begin the cycle again on Monday next week.

**2024-06-17**

Back in the office with a new task for the week. Matt came in and had me set up a new HMI for editing like the last two. This one is also on the same pipeline as the other, just a different site in a different location. This time I'm no longer editing items that already exist on the HMI, I am now making a window from scratch. Basically, the site consists of a few tanks that need to have a sensor because due to re pressurizing the oil along the pipeline, some gas would buildup in the tanks between the roof and the walls of the tank, causing a potentially explosive situation if a spark or some ignition source that could meet the tanks. The window I'm building is meant to trigger alarms when this type of environment occurs. I've been given a map of what the site looks like and where each tank and senor are relative to the site. I'm tasked with reconstructing this in the HMI and making sure that when the gas environment occurs, the operator is notified by an obvious indicator and can easily pinpoint which tank and which sensor is being triggered and why. Most of my day was spent looking and learning how to import and make graphics placing them and making them look nice, as quality and simplicity is the goal with most HMIs. Just like everything at this job, it's a lot to take in at first, but I'll get used to it eventually like everything. I will continue tomorrow.

**2024-06-18**

Continuing with the work from yesterday, I'm making steady progress on my window slowly but surely. I did a little more development in the sense of placing the sensor on the graphic. The sensors are custom made graphics I made that replicated an image I was given of them. These got labels for the sensor’s names and antenna names. The tank graphics also got a label, I also switched out the alarm graphic for a more compact one to fit in the tank graphic. Matt gave me some feedback and told me to update a few things, but he said I was on the right track, he also gave me an email with some name updates for the labels. Around lunch I had a meeting with Sean and the employer for the company we were updating reports for. He wants me to go through the manual entry reports he made and make sure the formatting looks good; a small task compared to the last time I did verification for the entire database. This is pretty much what I did for the remainder of the day. Matt told me to prioritize doing the work for this company as it is much closer to a deadline than the graphic I was working on. I will continue tomorrow with verification.

**2024-06-19**

Continuing with the stuff from yesterday. I finished verifying the manual entry reports today and Sean had me start going through and finding the spots in reports the previous system integrator had labeled as null. Basically, we think that he was labeling things null that he intended to go back and fix, but never ended up fixing. This basically just consisted of me using my tag lookup script to find the Aveva system tag name for each calculated tag. This is because when the old system integrator made the reports, he used a step table that would do a bunch on math within the report itself and bogged down the speed of the report generation drastically. So, I had to find the tag in their old system then use my script to find the new one. Then mark it down on our tracking sheet. After I found them all I went through all the reports that had these null values and put the newfound tags in place. This took a good chunk of time as some of the reports that had tables with all null values weren't even made in the new system, so I had to go through and make a bunch of them. Matt came by and asked me when I would be able to do the tag import for the HMI I was previously working on, and I told him I should be done by today and we could do it tomorrow morning at the latest. He came and told me we need to have a somewhat functional screen by Monday next week when they do the test. I will carry on tomorrow.

**2024-06-20**

Continuing with HMI stuff today, my window for tank levels is coming along nicely. Today I added some custom properties to my custom detector graphics to track different types of alarms that the detectors have. In general, they have a battery alarm which indicates the level, whether the communications are failing or not, a fault alarm for the detector itself etc. This allows the operators to see briefly what type of alarm they are dealing with. I also developed a popup that gives a general view of each alarm status for the detector and the %LEL of the tank. This was cool to do because this uses an object-oriented approach where you make one popup then when you click on the detector it takes the properties from the detector and assigns them to the indirect properties of the popup, so the popup is dynamic to the detector being used. This makes it so that you don't have to make a popup for each detector. Same thing when making custom graphics, if you put all the properties and qualities you want for the base graphic, you can then have those across all the objects in your HMI. This was fun to do and figure out all the useful shortcuts you can take to make your window more comprehensive. Matt also told me to make some alarm bypass pages so just in case the alarms that go off are not real or faulty for some reason, an operator can then dismiss the alarms with the right access level for the site. I'm going to chunk through those tomorrow and Matt said we might even get around to some testing for the screens I've been working on. I will continue tomorrow.

**2024-06-21**

Doing more HMI development today. Today I was mostly focused on making the alarm bundle pages for the HMI. Basically, they let the operator check all the alarms for similar controls on the site on one screen. It also gives a summary of what could be going on if there was a fault or a communication error or something along those lines. This basically involves me putting down a graphic and linking the alarm tag to the graphic to cause a colour change in the graphic when the alarm state changes. The tags that handle the alarms are like Boolean variables that only have two states, true or false. If the alarm is true, that means something is wrong and the graphic will turn to red colour usually. If it's false (which is most of the time) then the graphic will remain green. This is a super cool way of turning simple logic into something more meaningful using PLCs. The alarms on this screen also have a bypass which allows permitted users to turn off the alarms and allow the control to continue functioning as if there was no alarm. There is usually a reason for this but sometimes alarms can be faulty and need to be bypassed. Building out these windows was easy as there are a lot of similar windows. Mostly a lot of copy pasting of the same graphic and making sure they are positioned very nicely on the screen and pixel perfect alignment for the other elements around them. Matt really emphasized to me that the windows should all look like they are made by the same person. Which basically means that all the screens should have consistent formatting and font size alike.

**2024-06-24**

Did some more HMI development stuff today. I wrapped up making the alarm bundles today. I had to do something interesting for the newest one since it had so many alarm fisheyes on it. Since there were upwards of fifty alarms a second page needed to be made. This basically means that a whole other window is made for the alarms and linked to the first one via a button or some sort of object using a script. This script uses a built-in method that when given a window name, brings the window up at a specific x and y location on the screen, all of these are passed as arguments for the method. I linked a method like this to a button object on the HMI screen which just calls this method. Otherwise, the new window is identical, bundle name at the top list of bypass alarms with the same layout except for the page number indicated at the top next to the previously mentioned button. Of course, on page 2, there is another button to go back to page 1. I've really been focusing on adhering to the mantra of the HMI should look like it was made by one guy despite multiple people working on it. Which has involved me having to figure out how certain things were made on the HMI myself to make sure everything looks uniform and correct. Then after that Sean had me delete some tags from the AVEVA system for the company that hired us. They have a bunch of unused tags which are bloating up their system, so we need to go through and delete them. Sean made me a list of ones to delete, and I just went through and filtered out the ones that needed to be removed and deleted them all at once. The system uses pattern matching like SQL, which is basically regular expression syntax, so this felt like no issue for me. I also setup up the VM for Matt for alarm station testing for the HMI. I will continue tomorrow.

**2024-06-25**

Today Matt helped me setup the testing VM which will actually let me control the status of certain alarms for the LEL detectors. Basically, the software lets you change specific bits that PLC uses to communicate to the HMI. These bits are signals that are received from the site controls which tell the plc to flip certain bits that represent the signals being sent. These are reflected on the HMI by animations performed by graphics that represent these bit states in a meaningful way. For example, say a certain measurement that the control takes is out of range, the control then sends a signal to the PLC, the PLC is then able to interpret this signal and identify what control its from and what the signal means. These are then given names in the PLC software pertaining to the signal they represent. Then this name (referred to as a tag) is then used in the HMI, being assigned to graphics via properties or a script the cause a color change or animation to play based on the current tag state. This is the basic functionality of the whole system. The HMI is in the same vein as front-end web development. Placing elements and styling them to make them look nice. The PLC is more like traditional programming and backend development with linking databases and general functionality of the whole system. I'm definitely looking forward to doing some PLC programming in the future as this seems closer to what I'm familiar with in terms of programming. Matt gave me a brief rundown of how Modbus TCP works which is the communication protocol used by industrial controls. He showed me how the protocol sends and receives bits and what each of the hex values means. So now I'm able to test my HMI to make sure things work properly. I also made a bunch of functionality changes to make my HMI better. I will continue tomorrow.

**2024-06-26**

Today was some more HMI programming with a little bit of testing on the side. I finished fixing all the issues I found with my HMI since certain things were not working. These improvements were usually simple and were easy to fix. But then I couldn't figure out why one of my alarms was being triggered in one spot but not the other. This was quite annoying as I compared every aspect of both alarms, and it seems they had no difference. At least, there is no visible difference. For these graphics to have an animation triggered by an alarm, the alarm tag must be put into the properties to tell the graphic what state it should be in. There's two ways you can put a tag in the properties, in the provided text box you can either manual type of the tag or use the built in tag dictionary which will paste a selected tag in for you. Fundamentally there is no difference, but the tag dictionary just ensures that you are entering a tag that exists in the HMI. Since I had a dump of all the tags in the system, I was programming and would just copy paste the appropriate tag in the appropriate spot without a worry. Turns out this would include a newline when I would paste the tag in. This apparently causes the tag to be read as if there is a newline character at the end and treats that as the tag name. This software doesn't throw any errors to reflect this so after some troubleshooting. So that was a bit of a pain, but I eventually figured it out. I will continue testing tomorrow.

**2024-06-27**

More testing today. I made some improvements to the HMI in terms of adding popups for the repeaters and making minor adjustments. Then I just continued flipping bits in the software that allows to program the physical PLC. It's kind of fun flipping the bits and seeing the effort I put into the HMI work with the alarms and system reflecting my inputs. I made some checklists in obsidian that I want to use to keep track of which alarms I have tested, and which are working or need to be changed in some way. The PLC I'm specifically working with is made by General Electric and use functional block style programming. Basically, all the PLC tags are used as sources and inputs and are fed into blocks on the program design and then certain outputs can be achieved. For example, all the basic logic gates are present in functional block programs, this includes OR and AND gates but also all the others you might be familiar with. These looks really like Multisim and familiar software from school. The inputs and outputs of each functional block are labeled much the same way that the multisim components are labeled. This makes the software feel a lot less intimidating. It has been fun figuring out how things work and seeing everything come together in a nice package, which is why I enjoy programming related things ultimately. I will continue tomorrow.

**2024-06-28**

Well today I wrapped up testing on my HMI. The alarm bundles are what took the longest was the alarm bundles which are basically just pages of different alarms. Jeff helped me fix the issue I was having with the bypass buttons. The bypass buttons are meant to allow controls to keep operating despite them having an alarm. This is for when the alarm is not valid or unnecessary and just hasn't been fixed yet. When this button is clicked a prompt should show where the operator must write a reason for bypassing the alarm and when other users clicked the same bypass button, they can see the reason and decide to remove the bypass or not. He explained to me that there are 7 HMIs used by this and the 2nd HMI is where they decided to store all the SQL prompts. If they weren't all on a central location, then only users on a given HMI would be able to see prompts from that HMI instead of all the others. This is why all of them are stored on one. However, since I'm developing the HMI locally, it isn't connected to the HMI housing the SQL prompts. But Jeff implemented a hotkey to trick the HMI into thinking it is connected to the proper HMI with the SQL prompts. So, it was good to know that I wasn't the one needing to do that. Long weekend this weekend so I won't be back till Tuesday.

**2024-07-02**

Back to work after a nice long weekend. I finished off the task Sean gave me last week which involved developing the LIMS tag retrieval methods. I realized I don't know what LIMS means, not that it matters. Basically, these tags are used to reference data that has been sampled by the labs at the company. These samples are usually entered manually and don't have a regular data recording schedule. This causes some weird behaviors when pulling data from other tags in the same table. This is mostly an issue with Aveva reports not knowing how to handle blank data points and usually just choosing to error instead. Due to these the high frequency retrieval methods break tables and cause no data to be shown. To correct this, I made it so when you are retrieving data from a LIMS tag and use a range of per minute or per hour, it instead pulls all the values available from the historian. Otherwise, it pulls the data as normal. Then Sean had me build out a dashboard for some trends in the web client for him to close out the day. Guess I need to figure out what I'm doing tomorrow.

**2024-07-03**

Not much to write about today. I essentially did general office tasks as there was not any work for me that I could do with my current ability. Sean did task me with making a dashboard of historian trends on the web client but that took all of 5 minutes to complete. Other than some minor tweaks, I completed it rather swiftly. Guess I'll see what comes up with me tomorrow.

**2024-07-04**

Finally got some work today. I much prefer having something to do because it makes the days go by faster. Ryan came by and explained to me that the documentation that they use when they do remote plc testing with clients is kind of scattered in the sense that it doesn't follow a specific format. This is where I come in. Ryan sent me a document that he already made up and asked me to basically transfer it over to the old template. This old template has some custom properties is the document so that they can be filled across the document since they are project and FAT specific. Essentially, I just had to remove all that wasn't in his document and fill in the missing things. This took a bit as they are quite extensive since they are reports but was not difficult. After that was done Sean and I found a weird issue in the query tool that we use to check historical data for tags from SQL. When changing the view in this tool, some values being pulled by integral retrieval method the values would be halved. Obviously, this is a big issue as this is essentially false data, so we are going to eventually try and figure out why this is happening. I will continue tomorrow.

**2024-07-05**

Finally, Friday on a short week thanks to the long weekend. Today was a little more eventful. Sean had me come up with an excel formula that would replicate the functionality of an excel spreadsheet that someone from the company that employed use would use to pull and manipulate data from their old historian. The problem is with the new historian if you have two tags with historical data except one of them has data at a given time and the other doesn't, the system doesn't just show a blank, instead it finds the last value recorded and displays that. This is quite frustrating as it is deceiving as to whether a value was recorded or not. One sheet in the document, does some data formatting to be parsed later by some other sheet in the same workbook. The previously mentioned issue messes up the parsing done by the other sheets as they expect there to be blanks in the data retrieval. So, I was tasked with converting the data to match the old format used by the old system. It took a while for me to come up with, but it involved a double xlookup function with some inner string concatenation to replicate the formatting for the parsing sheets. I cried with joy once I figured it out and Sean seemed excited too. I also configured some more drivers for the run hours retrieval methods in reports. A simple day and a nice way to close out the week.

**2024-07-08**

Today wasn't super interesting. I just did some general office stuff again. Sean was in today he got approval to buy a new monitor. It's a 43 inch monitor that's basically like 4 24inch monitors in a 2x2 configuration. It's what he has at home and describing the reason he needed it to Matt was he would be in the office more often. This is beneficial for Matt cause I think he prefers having people in the office in general. Good for me to because it's nice to have some company instead of sitting alone all day. Other than that, I spent pretty much all day trying to figure out some issues the reports server for our employer. They did an upgrade last Friday and managed to break everything due to numerous different issues. One of the big ones is apparently the port the history one runs on changes by default in the newer version causing a bunch of networking issues. Luckily this one is easily resolved. The other one we spent trouble shooting is it seems that we can't connect to the web portal where all the reports are being hosted. Sean seems only to be able to connect on the local box not from the other server. We were looking through manuals and trying different things in the console manager, but we have yet to figure anything out. I will continue tomorrow.

**2024-07-09**

Today I was given work that took up the whole day. First Shane from the liquids team came over and showed what I would be doing. Basically, the template file is a file they use as a base for the HMIs for this customer since their sites are all very similar with minor differences across them. However, they follow a particular pattern when they make these windows. They try and make the comments of each tag related to the symbol in the they are for. However, for these older HMIs, this is not the case. So basically, my task is to go through all the database dump tags for the HMI and update their comments with the name of the graphic they are attached, their contextual to be specific. There are quite a few tags in the windows I was doing this for, so this took up most of the day. Simple task but that's what I'm here for. Then Sean came back into the office after his training and we both got to work with his client. Basically, I just had to make more trends on the web client. Then compile the trends into their respective dashboards. This should also keep me occupied while I go through all of them. I continue tomorrow.

**2024-07-10**

Today I continue building out my list of trends to make. However in a turn of events this took longer than expected. There's was lots of trends I built out and they feel like they just keep on coming. Not much to say about these but basically I just look up the tags and replicate them on the web portal. Sean and I did some troubleshooting though for the reports server. There were some networking or license issues cause the upgrade to go badly. We spent quite a few hours going through the console log and trying to figure out how the we could potentially fix everything that broke in the upgrade. We didn't completely fix it but we did manage to get reports working again and it loads a lot faster than it did before for some reason. We were on the call with IT from the company pretty much all day trying to figure it out. I continue tomorrow.

**2024-07-11**

Today I built out a list of all the trends I had to transfer. However this took longer than expected because it turns out there is almost 600 trends to transfer. I did the math and if it took me 5 mins to transfer a report, it would take me almost a whole week of work to finish up these trends. So naturally I turn to finding out a quicker way to make the trends into web trends. So I went into a deep dive to figure out how the web trends actually works. In short, it makes a GET request to an API that pull the data for the tag then some JavaScript build the chart. When you save the chart, a POST request is made and then data is written to the database with one of the columns being the JSON of the request. So I start by finding the table in the Runtime database where the chart data was being stored. I eventually found and made a dummy insert statement to make sure that this concept would work. With some precautions to prevent issues I executed my statement. And it worked! Then I went to the web client and couldn't find my trend inserted via the query in the search. I then went to the URL it should be located at and then it was there! this proves that this will work. I begin writing the script to build them tomorrow.

**2024-07-12**

Today I continued the development of my script to mass import the script to the web client. I got a lot of good stuff going today which is always nice. I got the script to the point where the user has to input the parameters for the script to run. This includes the file name containing a mass dump of all tags, the column containing the tag names and the sheet the tags are in. This is also the same for the engineering unit for the tags. Then the script goes through the folder it is present in, finds all files that are in csv format (since this is the default export format for the trend desktop app), looks at the top row of the file which contains all the tag names, then goes through the master tag file and finds the engineering units for each of the tags in csv. All that's left to implement now is to generate the JSON object for the tags and for the to build the randomly generated url for the entry. My plan is to have the output be a bunch of insert statements for sql to insert a bunch of different charts all at once. I continue next week.

**2024-07-14**

More script development today. We are getting somewhere, but today was kind of a wash. I came into the office and had no internet. I just assumed that it was an office related hardware issue. However it proved to be much worse. Turns out some construction project in the local area had taken out the internet by accidentally cutting one of the fibre lines provided by Shaw. This basically removed the internet from the office for the whole work day. With that I got offered to work from home for the first time ever since I need access to servers. I took that offer and went home. When I got home I basically just built more drivers requested by our client. This were just drivers that could pull min and max values within a certain time slice. Unlike before when they could only pull from a whole data range. I made a bit of progress on the bulk import trend script and should be able to wrap it up tomorrow. It's almost in the testing phase.

**2024-07-16**

Back on the script development. Office has internet today so I went full speed ahead on the script development. I managed to get a working version and got it to generate a SQL statement to update the table that contains the charts for the web client. I ran the statement and it worked! I double checked that it actually made the chart in the web client and yay it was there. Such a relief that I had a working version that means I could tidy up the code a little bit and see if anything can be changed or removed. Originally I was using the csv dumps of the trend files to get the tag names of the chart, but Sean informed me that despite the trend file extensions being "aaTrend", they are basically just XML files. So I just made new function to browse through the XML which was actually a lot simpler than going through the csv. Another thing I discovered is if I don't provide the tag name any engineering units in the JSON object for the SQL table, Then it automatically pulls them for the tag. This allowed to completely omit the previous engineering unit fetching I was doing and saved a lot of headaches. Not to mention it also made the script run significantly faster. Now the only user input needed is the name of the SQL table for the output insert statement and voila. In the same directory the script generates a text document with all the insert statements needed for each chart. I also later added functionality to generate the URLs for each of the charts since they need to be visited once for the web client to notice they exist and populate them in the list of selectable trends. This is a moment where I am very thankful I took the time to learn some form of programming because if I didn't this task would've have taken a whole work week where as the method I made didn't even take half and saved a bunch of time. On to the next task tomorrow.

**2024-07-17**

Since the script is finished now and working, I looked into making it an exe. This is a bit of a different process since this done in python which is usually a script / console based language with some external packages that can allow you to make GUIs. If I had made this in C#, which would have probably taken more time but been quicker (even though my script runs nearly instantly) it would've have been easy to give this a GUI and make very windows like in terms of UX. But for something like this the easy syntax of python and the fact that I already had a good idea on how to do this in python I just stuck with the snake language. Using a module called PyInstaller, you can compile all your python code in your environment into one executable file which can be ran by double clicking and then it will just bring up the console window as if you ran the code from the terminal. I found this to be a nice middle ground as it let me skip making a GUI but also made it easier to use for less terminal savvy users. That being said, most of the people who work are familiar with some applications running through a terminal as other programs like this one have been written but if I can make the experience a little better with little to no effort, then why not. I also made some more comments despite python being a very readable language just for future me in case I come back to the script. I also stripped out some code that was no longer needed which really cleaned up the script nicely. Afterwards I moved back to the work the liquids team gave me which is pretty much just updating a template file they use for their recipe screen on their sites. All in all a pretty good day.

**2024-07-18**

Another day of template file updates. This is where I kinda have a gripe with the difference in certain teams here. Basically the two different teams here like to develop HMI's slightly differently. The Gas team likes to build the windows in System Platform where they are treated as one single graphic with interactable parts. This means that all there objects and tags are defined within the System Platform. This makes for super easy object management with less features for visual customization. On the other hand, the liquids team uses InTouch which is a piece of software specifically designed for making HMI windows and liking tags to object in the HMI windows. This is still hosted through a System Platform Galaxy but all of the tag info and structure in contained within the InTouch application. This makes for some tedious problems for the trade off of nice looking graphics. One of these trade offs is template file which are loaded to form a baseline for new HMI windows. This allows the developers to keep a consistent look across windows they make. Unfortunately these are made by hand and need to be updated when things are changed after a while. This has been my job for two other relatively newer sites. The current one I'm working on is a huge pain because it's basically a Frankenstein of a bunch of other sites made in the past and the template file just never got updated to conform with the others. So it's now my task to do the updating. If this were done in pure System Platform then this would be easy. I could just to a big export. Instead I have to go through and update the file manually. This is gonna take awhile since the site is pretty huge. Friday tomorrow.

**2024-07-19**

More template file updates as I mentioned before. I brought my macro pad from home into work because I thought I could make more use of it here than at home because I don't really need it for any software at home. This turned out to be a good decision. Basically I can copy the things from HMI window that I need with one button press and then just paste them in excel, then highlight row to mark it as finished then move on to the next one. It also allows me to place my arm and hand in a more ergonomic position so reducing the strain on my hands is always a plus. Other than that I pretty much did the same thing as yesterday. I did end off the day with trying to help Sean figure some weird behaviour in the historian server for our client when data is getting entered. This proves to be difficult sometimes though because we kind of just got handed this stuff and have to learn how it works and then figure out the issue since we weren't the first company to implement this for them. Overall good week though I was super excited about importing all those trends with the script I wrote, really made me feel like I could automate a bunch more boring tasks with my current skill. But of course still need to keep on learning. I continue next week.

**2024-07-22**

More template updates today. I managed to finish the updates for PLC 1 today which is nice since it is about a quarter of all the updates that need to be made. I've streamlined my workflow a bit more to make the process a little quicker and more ergonomic since I am essentially just copy pasting. I mentioned my macro pad previously which I have configured to use some useful key binds in the InTouch Window Maker software. For example I have one bind that pulls up the caption menu and instantly copies the caption for me to paste into my spreadsheet. I also have a bind that bring up the tag menu so I know that I'm grabbing the right caption for the right tag. PLC 1 had the most windows to go through as well to look for the right elements with right captions I needed to copy paste into the spreadsheet. I'm onto PLC 3 now which has only around 140 tags vs PLC 1's 694 tags. It also only has around 4 or 5 windows to look through with not that many elements. I'll probably be at this for a few more days assuming that the rest of the PLC's are either bigger or more spread between the near 3000 rows I need to update in my spreadsheet. I continue on tomorrow.

**2024-07-23**

More updates today. I got notice this morning that Sean and I's employer is finalizing their upgrade to the latest 2023 version of the software suite today. Sean CC'd me in some emails and expressed that should be ready and able to help them while they are going through the upgrade today in case anything happens, mainly in terms of reports. During the testing for the upgrade they put on a dev box just to play around with some features they somehow managed to break all of reports and related software. Obviously this is not what we want since it makes reports completely unusable. They later found out it was due to some licensing issues but not even the developers themselves are sure why this happened. They had a couple meeting to try and troubleshoot why this was happening and to see if they could prevent it in the future. Not sure what the exact fix was but they seem to have figured it out. There were still some issues left over though. For the web client it seems they couldn't get anything to load due to some general http request error. This didn't even guess an error code or anything so it seemed like a dead end. Another support request was made and with the team that initially developed the web client software before being acquired by AVEVA. They apparently sent the IT lead that we are doing work for two dll files one for reports and one for the web client. After applying them these magically fixed all the issues they were having. These files are not mentioned in any documentation and don't exist anywhere else. I continue tomorrow.

**2024-07-24**

More template updates today. I'm almost done PLC 5 which means I'm over halfway I think? I'm not entirely sure as I've just been doing the work instead of tracking the progress of my work other than highlighting the cells I've completed. Luckily most of the PLCs seem to have a lot less than the first which makes going through each of them. But again it is more just copy pasting and making sure that the tag name for the caption is in the right spot. This so in the future the liquids team will have a more standardized template across all there sites. This is so all the HMI's this company and team makes look like they are all made by the same person and look relatively the same. It's kinda of the same as normal programming. Typically the companies adhere to a style guideline so that code is easier to understand and read when other people read it and so most programs are laid out the same way. I also got some news from Sean about the upgrade I mentioned. Turns out it broke the web client again. The issue is that the web client keeps reloading back to the login page after logging in. This only happens when using the server domain name not the localhost name. This is a super odd issue and the only ideas as to why it could happen that Sean and I came up with is that the configuration the company is using is hitting some edge case that testing before launching the software couldn't find. Regardless I continue with updates tomorrow.

**2024-07-25**

Finally managed to wrap up template updates today. It did take all day but I'm finally free from the spreadsheet hell. I was told that Cushing is one of the biggest sites and that it should be the last recipe template file that needs to be updated this year. Not really sure what's in the pipeline for me. I did manage to get my site visit organized though which is kind of exciting. So far the company hasn't really given me any indication that I'm doing bad work of any kind so I think that's a good sign. On the other hand I'm not really sure that I'm doing good work either, I like to think that I am though. Either way I can't wait for something new after doing that massive spreadsheet. Friday tomorrow!

**2024-07-26**

Friday now! I got given a new task this morning. Chris one of the company execs came by and gave me a rundown of what I would be doing. He forwarded me a bunch of daily reports that have values for every second of the day they are sent. He told me that some of the meters reporting these values have problems and that the company that owns them would like to know which ones and have them fixed. He wants to make a script where I dump all these spreadsheets into a folder and then look through each of them to see if any of the measurements are "stale". Stale basically means that the values are repeating indicating an issue with the meter. Then the program need to output a table in excel indicating the name of the report and which of the values are stale if any. Obviously python is pretty much designed for stuff like this. Especially with the openpyxl module which has a lot of nice built in classes for dealing with excel data and files. I built the majority of the processing parts of the scripts. Although I think I am going to make it multithreaded so that it can process all of the files in parallel which should hopefully speed up the program a fair amount so that it doesn't take as long. I continue next week on Monday.

**2024-07-29**

New week and I'm ready to work. I continued building out Chris's script. I

managed to get it in better working condition but when I ran it with all 60 of

the files in the folder. It took around 15 minutes. This is not acceptable by my

means. Chris seemed happy with it which is good, but I am not satisfied. I

shouldn't have been surprised it took long though to be fair because the script

had to process 2.1 million rows of data. So I decided to use numpy and pandas

which I wasn't before since I found out that they are capable of drastically

improving data analysis times. I not particularly familiar with the libraries so

there will be some learning involved but I'm excited to help improve my skills.

Also Shane came by for the site visit. It went really well, we talked about how

I've been doing on the job and things the program could improve. Matt gave me

some praise for the work I've been doing and commented on my good work which is

nice hear. Shane also discussed some plans they have for the program.

Overall it was good that the site visit went well. I continue tomorrow.

**2024-07-30**

Still doing some script work. I had an inkling that the script might have some data loss somewhere maybe due to the data type of the data in the dataframe or series. I loaded up the script in visual studio code for the debugger set some breakpoints. After iterating over my code a couple times I found that this was indeed a problem. The vectorization methods I was using took a slight shifted column replica of the data and calculated the difference between the two and returned a number representing the difference of each value across the two copies of the same dataset. However these values were not being maintained as floats to hold their decimal places despite my efforts to do so. So I tried a couple methods to keep them as floats but I couldn't manage to keep them as such. So with this in mind I opted to switch back to a traditional looping construct to manually iterate over the column of data and then compare each value to it's previous and then determine if there is 60 in a row. This took a while to rework cause I had to fiddle with data types but overall it went well and I got the program working again with only a minor performance hit.

**2024-07-31**

Well I wrapped up the script programming today I think. I reverted back to the traditional looping construct I mentioned before to check for values instead of vectorization. I tested this and this made more sense to use then vectorization especially after testing. Much simpler too debug too. So now I am alot more confident that my functions are working correctly and the program is spitting out an accurate value. With this done I have to wait for Chris to come back from vacation next week to show him a more developed version of what I wrote. I had a lot of fun writing the program which makes me kinda wish I was at a position where I got to do more traditional programming but that's fine for now. Afterwards Sean put me onto making an excel file that one of our clients used daily to be compatible with the new system they use and the excel add in that is now compatible with 64 bit version of excel. This is proving to be a little tricky to implement as we need to sort data by the retrieval method but unfortunately the built in sort function in excel does not update cell references on other sheets. This is the main gripe I have as if I move the data manually the cell references get updated automatically. I tried a couple things and couldn't really get anything to work towards the end of the day so I just called it and went home.

**2024-08-01**

More excel work with Sean. I spend all day banging my head against the wall trying to figure out a good way to make this work. The idea is that I use this to make a general template for file transfers for the other excel files the client has. This is made easy in theory since the historian add in excel has either you can pull raw values into excel or use a function that can generate and update values in realtime. However this function option needs some configuration. This is where my gripe arrives. You as a admin are expected to go the registry editor and change some of the values in the AVEVA configuration manually to get the full functionality of the add in that the company you work for paid over 6 figures in most cases to get a license for. Not to mention this has to be done on every machine that wants to use the excel add in. This is ridiculous and lazy in my opinion. Unless there is some reason involving the windows kernel that they can't program this on their end (which I would not believe on my end) there should be no reason for it to be this way. This is super lazy from AVEVA on my end especially when they offer a niche product that costs a ton for people to buy and use. Anyway we closed the day out with a meeting with the client and discussing some points with the project so far. I continue tomorrow.

**2024-08-02**

More head banging trying to figure out the template situation. I got the template to a state where I can replicate the old system data with the new system data in the same location on a different sheet. The problem is we want the data to be organized in a way so that you can do 1 query per retrieval method. This will speed up building out the spreadsheet and the query times for the spreadsheet as since Sean and the clients made the executive decision to stick to the old add in for excel. This one is a lot more like the query tool which is nice because if you're are familiar with query tool then it makes the extension super easy to use. The problem I'm running into is making a serviceable template to be used for future sheets. I don't even really know if it is possible. I would love to program something for this but it's unrealistic to think that an operator using this sheet would be expected to run a script to be able to transfer there data to different system. The main issue with trying to figure this out is cell references. For the most part Excel is good at updating cell reference when you move the data the cell is being referenced to. However, this isn't the case when data is being sorted using the sort tool. For some reason, cell references on other sheets do not get updated when the sort tool is used. My guess as to why is that Excel is moving the raw data and not moving the cells themselves like it might do in other operations. This is why this has taken so long. This while problem would be solved if this wasn't happening. Regardless it's a long weekend this weekend so I won't be back till Tuesday.

**2024-08-06**

More excel stuff after the long weekend but some progress has been made. I came up with the idea that I give each of the columns in the original data set a number. This number represents their positions from the data from the original system. My original idea was to organize the data by retrieval method with the order number previously assigned at the end of the queried data. Then revert back to the original data order so that I can simply replace the sheet name of the old data with the sheet name of the new data and have the data still be accurate. Sean is very up for this idea except for one part. He mentioned that reverting the sorted data won't work. This is because the way the historian plug in for Excel works is it uses an array formula that makes and API request. This formula can not be moved via sorting or it won't work, it even gives you a prompt in excel to let you know that you can't do that. So I need another solution. Seems like I'll have to do some brainstorming tonight. In other news, Sean told me that the script I wrote to import local trend files could be useful with the new client coming up. Anyway I continue tomorrow.

**2024-08-07**

Sean had me tackle something new today. On my way into the office I got a notification for a meeting the morning of. So once I got in and Sean arrived he told me I would actually be doing something for his FAT today. An FAT is basically where the client and the integrator test the individual logic registers and each of the bits on the registers. The registers are one word in length (16 bits). The general setup is the PLC is hooked up here at the office then the client's HMI is connected to the PLC here on a subnet and then mine an Sean computers are patched in to the subnet. Then we connect via Remote Desktop to a VM running on one of our servers that has the simulation software connected to the PLC to be able to access the logic. The point of this is to test the logic of the PLC to make sure that it works with the HMI, it is also to prove the logic can work on a PLC so when it gets pushed onto the clients PLC it should work without breaking anything, in theory. These are usually done in preparation for a site shutdown since usually systems have to be stopped in order to do these types of upgrades. Upgrades are usually done in shutdowns so that if something were to break the site doesn't stop running as some of the sites are moving around $200000 dollars in profit sometimes. So my job was to test the alarms with the client which is pretty simple. The alarms only have two states so they only need one bit to control their logic. The client and I went through all the registers and bits that are controlling alarms, I would flip the bit and they would check if that would register on their HMI. This took the whole day since there was a ton of bits to flip. Other than that it was pretty simple.

**2024-08-08**

Not a whole lot happened today. I helped Sean with some stuff this morning that he did for Enbridge. He has a meeting with them today to try and pitch something to them. Don't remember what is was though. I worked on updating the trend import script I made. Sean wants it to be able to take in a list of tags along with .aaTrend files. This for an upcoming client that has a similar project to the company I started work with when I got hired here. Later I talked with Rod who said he had some stuff I could do awhile ago. Basically we need to build out documentation for users at a site who manually import data so that they can basically give it to anyone and they can replicate the process pretty easily. He also told me there may be some scripting that needs to be used because they are having issues with their current script not deleting files on the host machine after sending them over to the server. He said he'll at least get to me to help write the script since I have some familiarity with scripting. So that sounds like it will be exciting. I've been trying to find some home projects to do to keep my skills up but I'm having trouble coming up with some projects to do.

**2024-08-09**

Got in early to the office today. Sean gave me the spreadsheet to do again since he now got approval to make a script for it that will do all the necessary conversion we need to make it work across pretty much all the sheets that need to be updated. Luckily the script is able to be written in TypeScript (a language processor for JavaScript) which basically allows you to have type hints in JavaScript. Everything is going pretty well basically I am making lists with the parameters I need for each tag and then making an object for that tag. This allows all the values need to be stored in one spot instead of across multiple data structures. But once I was checking my objects to make sure that everything is good. But I noticed some of the reference values were not quite lining up with what they were supposed to be. I troubleshooted assuming it was an issue with my function but nothing was working. So I investigated the data a little bit it turns out that some of the tags don't line up with the lookup values. Apparently this is because the lookup sheet has been made after Sean made some improvements to the tag dictionary. So I fixed one of the values and voila it worked. Now my objects are correct. I decided to phone it in for the day and work on it next week with a fresh brain.

**2024-08-12**

More script development today. I got the script to mock the output that Sean's excel formula did. The issue I'm still running into is that the tag lookup sheet isn't accurate for some of the tags included in the workbook I'm using for testing. I managed to find a workaround but this might not work for other sheets in the future so I'll need to find a better solution. Overall I'm happy with how it works. This morning I decided to start from scratch and refactor everything since last week I wasn't happy with how the script was designed from the start. The way I wrote it this time with inserting the values into an object I made as I got them instead of later proved to be more efficient and easier to implement. This made it much easier to get the desired result. I am having a little bit of trouble implementing the replace all functionality. I'm trying to used the range where cells have value in the function then replacing all instances of a string in that range with another string. This was the hurdle we ran into using standard excel. I am trying to iterate over every sheet in the workbook then get the used range of every sheet(the ranges that contain values in each sheet) and then replace the values that need to be replaced. For some reason this isn't working as expected and the output doesn't throw an error when the script is ran. The issue with excel script is they use like a mini VSCode for editing but the console output is much worse and doesn't explain much when an error is given. I'll do some thinking tonight and see what I can come up with.

**2024-08-13**

I got the excel script working! YAY! I figured out that since the cell references in excel need a leading single quote so they know that they are text, this messed up the matching for the replaceAll() function. This was awesome to get working and will certainly improve things in the future. This pretty much wraps up all the excel style project assuming the script can work on most spreadsheets. This leads to the next task I was given by Matt directly. Basically they have this tool programmed in python that allows you to calculate ideal PID loop tuning values given a data set. This was made by a guy named Ryan and the project is named CLAR with the "R" for Ryan. I don't remember what the other letters mean, but the more important part is that I can now rename the project CLAW, with the W standing for Wyatt. The reason Matt wants me to make this better is because the project is kinda unfinished and in a semi working state. Mainly he wants me to get the once planned linearization feature that doesn't work. This seems like it will be super fun since I haven't done a gui style project before so lot's to learn! I feel a lot more confident tackling something like this. Matt also sent me some material to try and understand what this type of software is actually analyzing and giving some context behind the purpose. Overall super excited for this especially since it's a backburner project that I can work on if I have nothing else to do. On to tomorrow!.

**2024-08-15**

More script development today. My main goal for today was to get the table view to take the data from a csv file and display in a table format to be viewed. Before I got there though I had to do some GUI configuration figured out. I had to go through the code of the old program and kinda figure out how they made up their GUI. One thing I missed is it seems as though Qt Creator (which is the software the old dev used to build the GUI) has layout object to organize controls on the GUI. This is kind of annoying coming from a C# windows forms app programming standpoint which has a really nice GUI configurator in my opinion. So nice in fact I've been thinking about pitching to Matt about moving this program to C#. My reasons for this are that the GUI designer is much nicer and easier to use. Plus I have a lot more experience with GUIs in general with C# then Python. In fact, I almost prefer to use C# when it comes to GUIs since I just prefer the configurator in general. Using C# would also make the program faster in some cases and have a more familiar feeling with most users here at the office since all machines are Windows machines, spare a couple servers. Obviously I would need to figure how to do the same style data manipulation that the ever useful pandas library does in Python except with C# but I'm sure that it wouldn't be that bad and I could figure it out. Maybe I will end up pitching him the idea to do it, we'll see. Rod talked to me today about getting me setup with Suncor credentials to help with their cybersecurity hardening project. It seems like this will be some Powershell scripting work and other such things. I continue tomorrow.

**2024-08-16**

Sick day.

**2024-08-19**

Just more script development today. I think I've gotten pretty much all the functionality of the main page working today. That is the page that takes in data. I'm basically using the main class where the application runs to hold all the data I need from this page. Essentially every input that is given by the user get stored as a private attribute in the class and is then accessed as a property using the property decorator in Python. This basically allows you to get and set a private attribute of a class instance while treating it like a regular attribute. I'll give an example below. (The example is done is markdown which you can see on the [github](https://github.com/Defl8/coop-work-journal) where have this whole work journal stored). This is just a more convenient way to access the values of private attributes and promotes code safety while still having access to values stored in the class instance. I've been using it a lot as it makes code safer and cleaner in my opinion. It also avoids having to call the getters and setters as actual methods individually. Plus you can add code to modify the value given to fit the attribute better if necessary. Later today Rod talked to me about some more Suncor stuff but basically told me nothing can happen till I get my credentials sorted out. He said hopefully that'll be by next week. Overall a productive day.

**2024-08-20**

More script development today. I cleaned up some of my code for the data tab. The original program lacks a lot of polish in terms of error handling. One of the first things I noticed when launching the console spits out a bunch of errors but somehow still manages to run. To be fair there is a fair amount of unfinished and unimplemented code that could be causing these but my point still stands. If anything this code should be commented out, removed or fixed, which is a big part of why I wanted to start this from the ground up again. Not to mention that the code is very disorganized and has some questionable choices made about its structure and procedures. For example the base program lack error handling for the most part and if it did have some sort of error handling the program wouldn't notify the user at all that they had triggered an error and something would have to be changed whether it be about their input or about the data they loaded into the program. So from my winforms experience (since this GUI software is basically winforms for Python) we should handle errors where it makes sense. On this software users need to a lower and upper range for the values from the data they load. Obviously, the lower range shouldn't be bigger than the upper range, but there isn't anything stopping the users from doing this so it needs to programmatically handled. My solution is when the user submits the data but the lower range is bigger than the upper range it pops up with a message box and says hey you can't do that (basically) and clear the range input boxes. So simple stuff like this will make the software much better. Eventually I'd like to add tooltips to alot of the controls so that it's a little more user friendly and doesn't require someone giving you a whole tutorial on a program that's supposed to be simple and quick to use. Tomorrow I'm gonna try and tackle the plotting of data.

**2024-08-21**

Today I spent racking my brain trying to figure out the implementation of the bump plot the old dev was using. Reading his code made it very hard to understand where, when and why certain things were happening. Not to much his comments are extremely sparse and don't provide much insight into the code at all so I don't even know why they are really there. Clearly the old dev didn't know much about project compartmentalization at all either since everything is just thrown into one file and not separated into different files based on use case making the main script of the application extremely frustrating to navigate through and read. Which is pretty anti Pythonic as they say. I don't know much about data visualization in Python with libraries like matplotlib since I've never really need to use them anyway. So I think I'll have to spend sometime in the future to really learn the library since it is quite big and offers a lot of customization in terms of plotting and visualizing data. So what I'm trying to replicate is the bump test functionality of the original piece of software. The way it is implemented in the original piece of software is that it plots the user loaded data. Then the user can use the provided zoom tool and snap that area of the graph that is zoomed in. Then the program does some sort of background math to deal with this collected data. I'm not exactly sure how the bump test works so I'll probably have to ask Matt about that one. Overall it's progressing nicely so far but I think this data calculation and plotting will take the longest out all the parts.

**2024-08-22**

More script development today with a focus on data visualization. I spent the morning going through the matplotlib documentation and API reference with the intention of learning some more about the functionality and syntax for making visualizations since the code from the old dev is all over the place with no comments so it's hard to figure out what does what for which visualization in the program. I did end up learning quite a bit about the base plotting functionality of the library if it weren't integrated with a GUI. Turns out there is a specific section of the API directly meant to integrate within the GUI software I'm using. It just makes the integration of the visualization onto the GUI more seamless while still following the same design philosophy in terms of programming. There are some quirks that I don't really know how to deal with yet. Since the data I'm looking at has two Y axes for each set of data, I managed to get two Y axes on the graph but the scaling for the second one is a little weird and I'm not sure why. The y scale for the first set of data is scaling fine but the y axis for the second isn't scaling at all which is odd since both lines of data are getting drawn. This will probably require me to do some more testing to figure out why this is happening since I don't have much experience with matplotlib in general especially not in a GUI scenario. So that just means reading some more documentation I guess. I continue next week.

**2024-08-26**

More script development today. I managed to figure out my previous issue with the y axis not scaling correctly. It's turns out that when you use the `twinx()` method it actually returns the secondary y axis as an object. Before I was just plotting both data sets on the same axes but still creating the the two different axis but I just wasn't using the other one. Once I plotted the other data set it auto scaled and then the graph looked how it should. No updates as to when I am going to the site I mentioned before. Matt says he's waiting for a reply from Leon as to when I should come out. He actually came by my office as I was writing this and they still don't know when exactly I am going to go to site. Matt says the earliest I would be going is Wednesday which alright with me the less time I have to spend there the better plus it gives me sometime to mentally prepare myself. It's kind of crazy that I'm going to site by myself. Kieran was telling me that he has never been to site alone and he's a legitimate worker here at DCD. He was even telling me some of the more senior guys have never been to a site alone. This is a little stressful but also exciting for me. Stressful in the sense that I have to the site alone and represent the company but also makes me feel like they believe that I'm capable of good work here since they trust me enough to go alone. It means they think I'm competent at the least. Still not really sure when I'm going but I guess I'll find out soon enough.

**2024-08-27**

Little script development today. I did some development to make an object to store the snaps for the bump plot but other than that I got some other remote support to do for the company that I am supposed to go to site for. Sean had me look at a spreadsheet I made a couple months ago. Basically the spreadsheet allows you to make queries for tag data in Excel. The issue with Excel add in for historian is that you can only use one retrieval method for each query you make. So if you want different retrieval methods you need to make a bunch of separate queries. Obviously this is very annoying as a user, so this sheet aims to fix that. All you need to do is put in the tags you want data for, select the retrieval method for each and you will a nice table with data for each of the tags. Sean just wanted me to double check that it still worked which it did. Next thing I had to check is how the SQL statements for the comment submissions on some of the reports took in the time. Right now they take in the time that you hit the submit button instead of matching the time period of the table. They do want the time to match the time period however. So after some investigation I found out that the time is getting pulled from the current time variable on the report instead of the time period calculation. So basically for them to fix it they just need to figure out if they want he start or end time and use that variable instead. It's kinda fun how Sean just delegates the SQL stuff to me since I've done it in school and it's more fresh to me then him. I really like doing SQL stuff so I don't mind at all. Anyway, I continue tomorrow.

**2024-08-28**

It's a remote support type of day. I've basically just been doing some standard remote support stuff for our client that is switching over there system and needs some help making sure that it at least goes semi smoothly. The big discussion involving me today was how to organize the trends in the web client since they haven't been organized yet. These are the same web trends that I imported a while ago using a script I developed awhile ago. The main reason I made the script in the first place was to import a large amount of trends from local files to the web client since Aveva has no native way of doing this. There was almost five hundred trends that needed to be imported and you bet that I'm gonna try and find a way to automate the process instead of wasting a whole work week remaking things that are already made. So with that in mind, I made the script in about a day, did half a day of testing and then began transferring the reports. The script pretty much just remade the JSON that the web client would push to the SQL table that housed all the web trend data. My script just takes the local file which is basically just an xml file that has data for the layout of the trend in it. So the script just looks at the xml for the tag names for the trend and then builds the json, encodes it then inserts it into the SQL table. I just did this for all of the trends that the company had and it worked out really well. One thing that I did not do was organize the data in anyway. The trends were already in a messy file structure so it was difficult for me to tell if their was any organization to begin with. Ultimately nothing was decided in terms of organization so I am waiting to here what they decide to do about that. The other thing Sean brought up to me when he showed up today was that the same client is looking to revamp their logging system. Right now everything they do is on paper which is really inefficient compared to digital methods of storage nowadays. Sean asked me to look into see if Aveva has any software under their umbrella that could cover this use case and I did find a piece of software called Teamwork that seems to accomplish what they want and more. So I need to look into it a little more but it seems like the right software for the job they need done. Sean even mentioned that I might even get the chance to lead this project a little bit and get some experience with it. Also still no news as to when I'm going to site. I continue tomorrow.

**2024-08-30**

More remote support today. I spent most of the day going through the report testing spreadsheet seeing which reports needed update and additions. As I mentioned before, I'm going through all reports that need a sum or average row at the bottom of their data tables. These are used by operators to get a quick idea if the they are hitting the right daily averages and can quickly identify potential process issues. This is pretty easy to implement in the reports it just takes a little while since you have to do it for every column of data in the table, so if there's a lot of data on the report then it can take awhile. But I don't mind doing this style of work every once in awhile because it's pretty chill and doesn't take a whole lot thought so I just turn on a podcast and get to work. Sean messaged me around noon to tell me that the business individual we made the proposal to is very excited about the software we suggested we implement. So next week I think Sean wants to get a box setup with the software running so that we can build a demo to show the client. This is pretty cool since I think he's gonna let me do most of the work so that I can get a little more hands on experience with the business side of the company when it comes to making proposals and doing demos. This is cool since this is a side of the company I have actually become quite interested in since I've never really gotten to experience anything like it before. I'm really looking forward doing this! I also built out a report earlier to test some values for Sean but nothing of note really came from that pretty standard stuff for troubleshooting. Long weekend so I won't be back till Tuesday.

**2024-08-30**

More remote support today. I spent most of the day going through the report testing spreadsheet seeing which reports needed update and additions. As I mentioned before, I'm going through all reports that need a sum or average row at the bottom of their data tables. These are used by operators to get a quick idea if the they are hitting the right daily averages and can quickly identify potential process issues. This is pretty easy to implement in the reports it just takes a little while since you have to do it for every column of data in the table, so if there's a lot of data on the report then it can take awhile. But I don't mind doing this style of work every once in awhile because it's pretty chill and doesn't take a whole lot thought so I just turn on a podcast and get to work. Sean messaged me around noon to tell me that the business individual we made the proposal to is very excited about the software we suggested we implement. So next week I think Sean wants to get a box setup with the software running so that we can build a demo to show the client. This is pretty cool since I think he's gonna let me do most of the work so that I can get a little more hands on experience with the business side of the company when it comes to making proposals and doing demos. This is cool since this is a side of the company I have actually become quite interested in since I've never really gotten to experience anything like it before. I'm really looking forward doing this! I also built out a report earlier to test some values for Sean but nothing of note really came from that pretty standard stuff for troubleshooting. Long weekend so I won't be back till Tuesday.

**2024-09-03**

More support today, mostly. I kept going through the reports in the list and adding the missing statistical function tables that the operators have requested. This again isn't the most interesting task but very chill and nice when I don't have a lot of brain power on the first day of the week. Other than that I chatted with Rod today about some work I could do for one of his clients. Basically they need some scripts developed to transfer data between a bunch of different sites to one central SQL server. The script is gonna be done in in dos cmd because then we won't have to worry about differing windows versions across all their different sites. That is probably the most challenging part about this is using dos instead of something like python which has much better utility when it comes to this style of projects. But alas this will be good practice and something new to learn. Other than that I just did some more remote support for our client towards the end of the day I continue tomorrow.

**2024-09-04**

I pretty much only did script development today. It's taking me a bit to get a grasp on how batch script actually works and the syntax is probably the most difficult part. I was able to get the script to a point where it can recognize whether the specified file to indicate that it needs to transfer or recognize that there are files in the directory. The hard part I have been brainstorming about is how to recognize if the file is an hour old. Since there isn't even really types in batch scripting it makes a little difficult to deal with dates since there isn't objects to handle more complex things like dates, which can vary in syntax drastically. So basically my general idea is to get the timestamp of the latest file, isolate the time from the date and then express it in seconds and find out if the difference is greater than an hour. The way I thought to do this see if the file time is greater than the system time which would indicate that the file is from an earlier day since that is the only situation where the file time would be greater than the system time. If this is the case then we can subtract the sys time from the file time and if the result is less than -1 then we know it is at least an hour old compared to the system time when the script runs. This method only works if the time is being reflected in a 24 hour format. In the other case the file has a time that is less than the system time which means that it was made on the same day earlier or early on on another day, so we automatically know that it is old enough to move. No math necessary. Obviously, this is easier said then done and I'm likely forgetting something when it comes to actually implementing the code. But that's why you test. Overall, I'm happy with how it's coming along and can't wait to see how it turns out.

**2024-09-04**

I pretty much only did script development today. It's taking me a bit to get a grasp on how batch script actually works and the syntax is probably the most difficult part. I was able to get the script to a point where it can recognize whether the specified file to indicate that it needs to transfer or recognize that there are files in the directory. The hard part I have been brainstorming about is how to recognize if the file is an hour old. Since there isn't even really types in batch scripting it makes a little difficult to deal with dates since there isn't objects to handle more complex things like dates, which can vary in syntax drastically. So basically my general idea is to get the timestamp of the latest file, isolate the time from the date and then express it in seconds and find out if the difference is greater than an hour. The way I thought to do this see if the file time is greater than the system time which would indicate that the file is from an earlier day since that is the only situation where the file time would be greater than the system time. If this is the case then we can subtract the sys time from the file time and if the result is less than -1 then we know it is at least an hour old compared to the system time when the script runs. This method only works if the time is being reflected in a 24 hour format. In the other case the file has a time that is less than the system time which means that it was made on the same day earlier or early on on another day, so we automatically know that it is old enough to move. No math necessary. Obviously, this is easier said then done and I'm likely forgetting something when it comes to actually implementing the code. But that's why you test. Overall, I'm happy with how it's coming along and can't wait to see how it turns out.

**2024-09-06**

The remote support is really ramping up today holy. Pretty much the support is just investigating problems with reports or fixing other minor issues. A bunch of operators are not really familiar with how the Aveva reports works since they are usually against changing their daily workflows and learning a new work to work. So with that in mind the issues that get brought to Sean and I are usually pretty surface level and I can solve them independently. So my days recently have mostly consisted of responding to operators and helping them with achieving what they need to in reports whether that be problems with data showing and inputting comments or trend functionality on the web. Occasionally we get backend problems that Sean usually has to deal with directly since I don't know/have access to the backend, which is something I would be interesting in learning at some point in my next couple months here. I have seen some stuff involving it before and it is basically just like a GUI version of an object oriented language. Like every tag is an object and then the tags underneath tag groups are just like subclasses that have similar properties across all of them similar to how inheritance works in OOP style paradigms. So I'm hoping if I ever get to the point to using Aveva System Platform, it'll be relatively familiar when I get around to learning it. Otherwise I have been chipping away at the script Rod gave me to write since I've been so busy with the remote support. I have to the point where I am handling all the conditions but I'm struggling to find a good way to handle the checking of the file age. It's extremely difficult to really understand what's going on with the batch script since there isn't like a debugger or anything that most modern languages have. Either way it's the weekend so it'll have to wait till next week.

**2024-09-09**

More remote support today. I really wanted to get the script for Rod finished today but the remote support was coming in heavy today. A lot of backend issues it seems today so Sean was doing most of the big changes while I just went and helped him check/change any small things. There also some update to be made to some of the comment reports for the operators. For some reason the comments were not getting pulled and we had to test them in query to figure out why. We determined that they were using the wrong data source for the comment tables. For some reason the previous integrator has two data sources name full and full\_all respectively. Sean and I don't really know what the difference is and we still haven't figured out what the real difference is but for some reason the tag pulls the correct comments with full\_all and not full. So this was a pretty simple change for me to make across all the reports with comments. We are having some issues with committing to the repository with changes overwriting others. Basically there is a dev server where all the reports changes are implemented then send to the prod server for the operators to view and use. However it appears if someone has another instance of the report dev client open and they push changes without pull from the repo after someone else has made a recent change it does not save the changes from the other user and reverts them back to the state that the most recent commit made by someone else was using. Other repo style tracking like git has measures in place to handle this but it's clear that Aveva is not using the same style of implementation. This makes it super annoying and limits the frequency of people making changes we can have for the client, and gives more work to those that are able to use it since they now have to make change for other people when they want to update their reports for their operators. Anyway I'm gonna try and finish the script for Rod tomorrow so we can get it tested on the client servers and such. I continue tomorrow.

**2024-09-10**

More remote support. Sean was not in the office this morning. He texted me

saying that he was at Alta steel getting a new client. He told me something

crazy, he told me that they are paying for an Aveva license but not even using

half the products they are able to. They have a system for the way they make

"reports". Basically, he told me that they go into the trend tool which keep in

mind is designed for viewing data on a graph, then they dump all the tag data

from there into a file, copy paste that into Excel and there is their report.

Meanwhile they are paying for the historian add in from Aveva which basically

allows them to do this while interfacing directly from their Historian setup.

Not to mention that reports studio allows them to do in a much more efficient

manor where you can just make the report and then have on the file data being

pulled with different time steps and such. So I have no idea if I'll be put on

this project or work with this client at all but I imagine that I will be at

some point because it seems very reminiscent of our current project that we are

providing support for. That on top of another upcoming similar reports style

project. Overall lot's of things to keep me busy till the end of my coop

position. It's kind of crazy how September is the fifth month that I've been

here, and I've definitely learned a lot so far. I continue tomorrow.

**2024-09-11**

More support today. We ran into a big issue today with reports. I mentioned it prior but we are running into issues with how the commit system work for reports. Often it seems that changes people make don't \*stick\* and are lost to time. This is extremely frustrating cause it seems that if you don't commit often or someone commits right after you without your updated state of the project, your changes get lost. So for stuff to stick we would have to contact every one using the report client and then ask them to logout of the client or not make any changes until we implement our commit then ask them to pull the new change into their local instance. Obviously with the amount of users this is a huge waste of time and unsustainable. It seems to happen slightly randomly so it feels pretty unpredictable so the changes are often removed without people noticing unless they go back and review changes they had made and notice something is off. The biggest thing happened to where the lead on this project at the client company made some changes to be able to easily identify which folders contain manual entry reports for operators. This is nice since some operators have to enter a lot of data everyday and if we can make the process easier for them by not having them dig through a bunch of files then that's a big win. Unfortunately, this change was lost due to the mentioned commit problems. This is just one example of how annoying and frustrating this problem can be. Not to mention that I have had to go back and change a report two to four times in some cases because the initial changes I would make would not stick at all. This is a big waste of time and is directly reversing progress that we are making. Today Sean suggested that we submit a ticket request to aveva to see if we are missing something or if they can help us fix this issues. Welp, more of this continues tomorrow.