Today is the first day of my first internship. Company building is located in Maltepe, Istanbul. Building is located in Asia aside of Istanbul and acts as headquarter for Turkey and middle east areas. After arriving to the building, firstly I got a guest card from consultation area, and communicated to my supervisor Mr. Anıl via phone to await further instructions. Once required steps are completed, they have communicated with me again. One of the first information that I gathered was that I have a buddy – a person that helps me personally to settle in – named Sarpkan. He also contacted me via telephone. Also, I have been informed that before coming to the Office, I needed to complete several things inside the building. Firstly, I headed for the building infirmary for medical examination from workplace doctor. Nurse and doctor were very kind, I have stated my medical situation. They have recorded and asked me to take caution about my chronic situations. Otherwise, there was not a situations that restricts me to work in the office building, thus my medical examination was completed and marked as positive. After finishing health related mission, I have waited some time for my training about Occupational Health and Safety subjects. After waiting some time, I headed towards elevator and get on to the second floor. There I saw another person waiting fort he same training. She was working in Marketing whereas I was working in Design Center as Application and Calibration Intern. We headed towards a nearly empty meeting room inside the floor. The trainer was waiting for us in the room. After introductions, we have reviewed subjects like fire and hazard safety, earthquake instructions, building layout, working hours and System, automatical timed lightning System inside building, floor Office layout now and future plan, working settings, health cautions about Covid19, smart work System, data security etc. We asked several questions to the instructor during the training. Currently, there are no restrictions about remote work hours. In other words, people may attend to work %100 remotely if their work is compatible with this type of work. Sometimes, the worker may need to go office or to customers based on their work. Also, some offices workers may want to go to the office for social needs. After training has been completed, I headed to the 6th floor where design center is in. Today, my supervisor and my buddy was not inside the office but they were working remotely. My supervisor is also a team leader. In our department there are several team leaders. One of the team leaders – Tuncer- was working in the office today. Thus, he helped me to get inside floor gate and sit in one of the office tables. Also, they have communicated with IT department form y work computer. Then I headed fort he second floor again for IT, and got my work computer, set my password, and also I have collected head speakers, keyboard, mouse. After getting all of my work equipment and getting technical information from IT, I headed back to the office. Office style is open style. In other words, all tables and chairs are in a big room. People may sit in a table with their choice. There are also closed rooms called Focus, where one or several people may get inside and work in a silent room. Aside from these rooms, there are also meeting rooms in floors. Floors are very similar in layout. Our department has also a LAB room, where connecting an automobile is simulated with some ECU hardware, support hardware, computer and other equipment. Today, I generally got introduced to my work friends. Mr. Kumar from India helped me especially to try to understand how vehicles work. Also, in coffee break I talked to Mr. Burak about standards, auto systems. There are also different departments inside our department, which work in different areas. The inside department I am working with, is working in automation with Bosch France colleagues. But today, I only worked to understand my work computer and install apps such as Anaconda (Python). There are also, other technical apps for automative calibration, but for confidentiality I do not give the names for now. For me, today passed fastly with very informative and positive colleagues. Then I finished work at 6pm.

Today is the second day of my internship. Today, I tried to use Anaconda Navigator. To install Python in one computer, we may use one of three methods: installing pip, installing miniconda, installing anaconda. Pip and conda are Python module managers. These methods may be explained as:

- Pip: This method may be explained as native python installation. Pip is a module management system for python. With it we may install modules, apps, python etc.
- Conda: This is the external module manager. Have two versions for use:
    - Miniconda: This is the base Conda version without visual interface. In other words, we may need to use cmd – terminal to install modules and analyze situation.
    - Anaconda: This is the supported version of Conda for Bosch. With this we may use Navigator interface or use cmd to install modules and analyze situation.

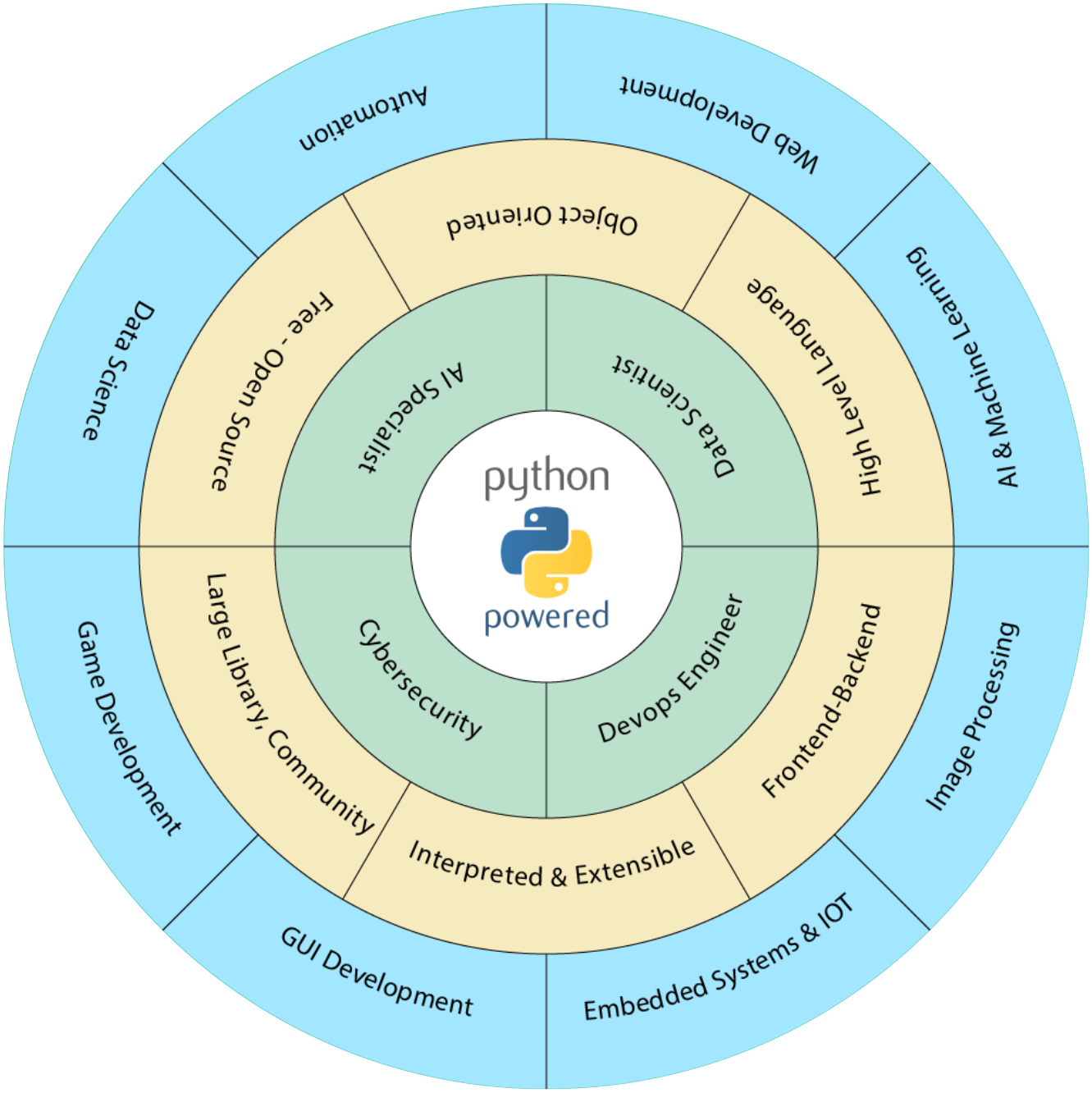
Anaconda was already installed when I took my work computer. Before understanding the system a little bit better, I had always use Navigator to install packages. But later on, I thought that using cmd to install modules are more practical and faster. Also, some modules were not visible in Navigator. I need to search Conda website for the command that I need to write to install module on cmd. And I could not find a way to use these command lines in the visual interface. In this day, I introduced myself to more colleagues and tried to understand what they are working on. I also searced for Conda cheat sheet, in other words the commands required to enter at cmd to do certain tasks. The table is as:

conda info	Verify conda installation and version number
conda update conda	Update conda to latest version
conda update module-name	Update a module that is installed on conda
conda update --all	Update all possible modules (python too)
conda install module-name	Install a module by name
spyder	Launch spyder app if installed
conda install --help	Launch help
conda create --name env-name	Make a new conda environment with a name
conda create --name env-name python spyder	Make new env. With python and spyder
conda list	List all available modules inside environment
conda activate env-name	Activate and use a conda environment
conda env list	List all conda environments available
conda deactivate env-name	Deactivate and stop using a conda environment
conda install module-name	Install a module with conda
pip install module-name	Use pip to install a module
conda install module-name=1.3.5	Install a module by specific version 1.3.5
--no-default-packages	At making env. we may add this to reduce size

I had more than two app options to use Python - Microsoft VS Code and Spyder. Due to the interactive variable view window, I started using Spyder. This is especially prepared for python whereas VS Code may run multiple languages with extensions. VS Code has support for many programming languages but I felt more comfortable using Spyder. Today I also learned how to communicate via internet. We use Skype and/or Teams to join online meetings. Before joining a meeting, I need to click join button. There is also updateable calendar System connected to Outlook mail to follow dates for meetings.

WORKING DEPARTMENT	**Application and Calibration Engineering Department**	**PAGE**	**1-30**
FIELD OF WORK	**Python Automation, Matlab, SQL, Calibration**	**DATE**	**27/06/2022- 11/08/2022**

Today, there was a monthly meeting for our department. I got introduced to the department, and I tried to explain my work areas and interests. Currently, I am motivated to work on automation via Python. One of the advantages of Python is (beside data science support) that it is very practical to prepare automation apps to reduce manual work and reduce work hours. I also prepared Python beginner and advanced training manuals for orentation for new workers. One infographic that I have prepared:



Company Representative Approval	Title Name, Surname	Application and Cal. Eng. Selim Çökren	Company Signature Stamp	

Today, I have continued Python advanced training. I would like to add some of my writings in beginner and advanced Python training files in the following area:

OUTPUTS
- Learn the meaning and purpose of Python programming language
- Learn how to start coding in your work computer
- Learn technical terms related with Python
- Learn the difference between native Python (pip) and conda Python
- Learn the meaning and the purpose of a Python environment
- Learn how to make a new environment in two methods
- Learn addresses of help pages for Python related issues
- Learn possible ways (roadmaps) after getting proficient in Python
- Learn possible work titles related with Python development
- Learn some programming things to keep in mind

WHAT IS PYTHON?
- Python is one of the most popular programming languages in 2022.
- Python is easy to learn and use, has high productivity, very flexible, has many libraries and has a supportive community. It is used in many areas in companies from different areas.
- Some of the usage areas of Python are automation, artificial intelligence, data related areas.
- One disadvantage of Python is that it is slower than some other programming languages.

WHAT IS ENVIRONMENT?
Environment is a depository for different Python modules. Developer makes a new environment, selects the modules that he/she wants to use, install the selected modules to the new environment, updates the modules in the environment and removes modules from the environment. (if needed)

THINGS TO KEEP IN MIND
It may be helpful to keep these in mind when writing a new Python program:
- Is the code understandable for other developers?
- Does the code run fast? Is it possible to get it running faster?
- Does the code only use the modules that it requires?
- (If with GUI) Is GUI understandable by users?
- Is the code autonomous and offer flexibility?
- Is the code too short/ too long? Do the developer made unnecessary steps?
- Can same purpose be achieved with in built modules / shorter code?
- (If with Executable) Does the program gave any errors while testing in other computers?
- What are the possible steps to improve program in short and long term?

DO I NEED TO MAKE A NEW ENVIRONMENT?
As default, you have a base environment. But making a new environment helps to install only the modules that you require and install specific versions of required modules for different purposes. These purposes may be compability issues, legacy testing, personal choice and so on.

Today, I have completed Python advanced training. I would like to add some of my writings in beginner and advanced Python training files in the following area:

MEANING OF TECHNICAL TERMS
Meaning of terms are:

- Python: programming language
- Module: Classes+functions+var.
- Conda: Python module manager
- Miniconda: Conda without GUI
- Anaconda: Conda with GUI (Nav.)
- Environment: Module Depository
- IDE: Editor for coding and running
- Spyder: Scientific Python IDE
- VS Code: Multi Language IDE
- Jupyter LAB: Visual Sectional IDE
- .py: Python code extension
- .ipynb: Jupyter Python extension

NATIVE PYTHON VS. CONDA PYTHON
In Bosch, installing Python only (without Conda) and using pip (Python native module manager) is not supported. That means, you need to install Conda first (Anaconda), then open a command line and use Conda to install new modules and update related packages. Meaning you may not use pip to install or update modules, packages. Also, mixing pip and conda is not recommended.

USING CMD TO MAKE A NEW ENVIRONMENT:
Open cmd in computer (more information: conda)
Enter: “conda create --name my_env python” (Add spyder if wanted to end with space)
(If you do not want to install default packages, add “—no-default-packages” to end)
After completed, enter: “conda activate my_env” to activate new environment
Enter “conda list” to see modules inside your new environment
Enter: “conda install module_name” to install a module (If you are stuck or conda install module_name does not work, search Google as conda install module_name. In first result, you may see anaconda website. Copy inside code to install the module)
Enter: “conda update conda” , “conda update python”, “conda install spyder”, “conda update spyder”, “conda update --all” and so on.. if wanted
You may see your new module at Navigator as well. You may select your module, and start the IDE of your choice to use you new environment.

I also write my opinion of what work areas to focus after getting proficient on Python and Pyhon related work titles. In the advanced Python file, I have also wrote (with heavy help of Python community) much used different Python modules for different work areas. I have also written some Python books, community names and International AI and data science competition websites.

Today, I have trained myself on data visualization programs. In my work computer, some of the data visualization options were Microsoft Power BI and Tableau. In the past, from Udemy I have attended a Tableau course. It was very fun and amazing to see the usage areas and effectiveness of data visualization. In a case we are more attracted to colors. Using visual things may help us to understand the data faster and with the combination of data science with data visualization, we may see more insights about the business.To use Tableau in my work computer, I needed a license whereas using Power BI was free. I liked Tableau's user interface more than Power BI, and I approached for a license. Afterwards, I took the license and started using Tableau and Tableau Prep Builder. In the day, I also contacted local IT via an app to get an administrator password to make a new conda environment on cmd. Afterwards, I have installed modules such as pandas, openpyxl, numpy, datetime etc. In the meantime, I have attended the mandatory trainings on our Bosch internal training website. In first 5 trainings, I have learned new information such as international rules, env,roonmental health and safety. I have completed other modules in different days, but to try not to break the flow of the subject, I am writing other training subjects that I attended. I saw courses related to data protection, compliance, codex, calibration, code of business, confidentiality, laws etc. I have also attended another course related to occupational health and safety, but only fort his course I have went for TISK website. In the remainder of the day, I have tried to communicate with other colleagues again. There are experienced workers and interns in the department. My buddy is also an intern at this time. (He went to be a full time worker in the future.) As far as I can see, the work environment is not very strict on time but focuses more on the needed work to be done. There are less female workers than male workers. Everyone is very helpful to give information and everyone is very hardworking. Also, they are very colorful characters. I also realized that I love coding in Python as well. Thus these aspects have became an extra motivation for me. In these days, I focused more on training. I also, tried to understand the LAB System as well. As far as I understand, a car has many ECU's inside. ECU's act as the brain of the vehicle. They have many inputs from many sensors around the vehicle. This reminds me the many sensors of aircrafts. Calibration is kind of connecting to the car's ECU and seeing that with given inputs do we get the correct range of results, if not we change some values. Also, customers may want to add new functionalities to the cars, and they may want the calibration engineer to look especially to the source of an error. In similar cases, we use an app to connect to the vehicle with connection cables, where the app is developed with a wholly owned company of Bosch. When connected we make some changes at some variables, functions for example. W emay also need to drive the car and do specific maneuvers to observe required values. Thus, this work may need more than one person at the same time. Bosch has Turkey based customers and international customers. I also saw that some workers moved to international Bosch work areas to work there. Also, I saw that some colleagues went to different countries to perform calibration tasks. The company pays the required expenses for these trips. Right now, I am working on a more software side. I do hope to attend to a vehicle test one day as well. I especially like electric motors. They make less noise, generally have higher performance than internal combustion engines and they have less carbon footprint. Coming back to the subject, I entered to the LAB room. In LAB room we may use data recorded ECU's to work as like we are working on an automobile. I searched some documents on the lab room about different pins on a sort of electrical and mechanical system. I did not have success on this situation, for me understanding different codes on pins was hard. But again, my colleagues were helpful to reduce my mistakes. But I did not have many success on this pyhsical training today.

Today, I started to attend weekly meetings about a automation task via Python. The system was like this: a previous worker has completed a Python script that automate a task and give outputs. Me and Selim were working on required functionalities to be added. There is a website that records customer wants, in this case France team's several members. For example, we need to add the functionality of producing one more output file from the program. Also, the France member we were talking to was very understanding and positive. For example, in one task we needed to find the defined but not used signals. By saying defined, I mean calculated signals where a signal is calculated by several signals with a relationship. We also needed to add a log file - excel as an output file. After learning these requirements, I looked to the previous (left) employees work, for me it was long and hard to understand because I was not an experienced employee. Also, I was missing some information about native Python and some modules. The program required nearly 5 minutes to give the results after it was started. For input, the program only uses a data file - excel file. There are hardcoded coding lines. One of the things that the France worker asked me of was to increase the autonomy in the program. This concept was very strange for me in the first times. Later on I understood that we may increase autonomy by reducing hardcoded lines, using patterns to acknowledge areas without user input etc. Also, I was recommended to try to use more native Python modules or Python module functions, write less lines of code but make it more readable etc. To not to break the flow of the subject, I write the following days of this Project in here; I studied Python for weeks and in every week we had a meeting to review the codes. Also, because it was using 5 minutes fort he results and it had many loops and some hardcoded lines and it was hard to understand and add functionality for me, I had decided to rewrite the code (refactor) again, with more autonomy. For weeks, we worked together. In the starting days of my internship I was not productive, I had a large Python information knowledge need and I was going to school. Based on the knowledge need, I was requiring more time to understand and write lines of code. Nearing the end of the work finishing date, I have tried to complete the script. In the script, I have used shutil, openpyxl, pandas, numpy, json, sys, os, re. These modules were used in different tasks within the code, such as shutil was used to delete folders and/or files, openpyxl was used to read Excel files and write data inside, pandas was used to make dataframe from the information inside Excel file, numpy was used for mathematical array search task, json was used to convert a nested dictionary to a hierarchial json file, os was used to make folders, sys was used to terminate code with or without an error message, re was used to search for patterns inside a text file or string. The hardest to understand between these modules for me was re in other words regex. Regex is similar to the ctrl+f search System in a .txt file. For example w emay search with a string in a search box and it may search with some string where the string may be inside a Word or is a Word. With regex w emay search for patterns thus it is some kind like conditional search box. For example w emay search for all occurrances of a Word, or w emay search for a substring between two words or characters, we may also search with a word starting with a specific character and continue with a decimal or digit etc. This module has many different uses for different tasks. I could not understand the usage of this module in this Project, thus I have used previous developer's code about this module. Days later, I hardly managed to understand the usage of this module and used in a different project. Coming back to the subject, we were coming close to the project handover date, data science related projects was handling to Bosch Italy. Near the end days, I made a .exe file, added user and developer guides for new developers. We have prepared a developer refactored version with one bug. Where using multiple Excel files were not giving an error but using a single file was giving an error.

Today I have continued the training file for data visualization for new employees. (Quick start orientation) I am submitting some of my writings in the following areas:

What is Data Visualization?

- Data and information visualization is to graphically represent data.
- We are drawn to shapes and colors. With visualization we understand and share data faster.
- Data visualization is compatible for many sectors.
- Data visualization is very effective in big data projects.

Data & Info Visualization Tools External

- Pie Chart
- Bar Chart
- Histogram
- Gantt Chart
- Heat Map
- Waterfall Chart
- Box and Whisker Plot
- Scatter Plot
- Pictogram Chart
- Timeline
- Highlight Table
- Bullet Graph
- Area Chart
- Choropleth Map …

Do We Need to Memorize How to Draw All Methods ?
No, there are data visualization programs which automate the process . You only need to supply data and method . Some data and information visualization programs are:

- Tableau
- Power BI
- Google Charts
- Grafana
- Chartist.js
- Fusion Charts
- Data Wrapper

Otherwise, I have continued on training. For trainings that are closing on deadline, I tried to complete the trainings. Also, I have gotten my entry card and signed my internship contract. Also, I have gotten my electronic signature which is based on Adobe. With this, I may sign digital pdf files. In HR, I met new people and I saw a book with Artificial Intelligence in HR related which is very important.

Today I have completed the training file for data visualization for new employees. (Quick start orientation) I am submitting some of my writings in the following areas:

What is Tableau?

- Tableau usage is supported by Bosch. You may need to obtain a paid license.
- It is a visual analytics platform transforming the way we use data to solve problems empowering people and organizations to make the most of their data.
- The tool is very strong and simple to use . Please take a look at these data visualization projects that are prepared in Tableau.

What is Power BI?

- Power BI is the Microsoft's tool for business intelligence data visualization. You may use Power BI without paid licence in Bosch.
- Connect to and visualize any data using the unified, scalable platform for self service and enterprise business intelligence (BI) that's easy to use and helps you gain deeper data insight.
- Some project examples are shown in the following website

Extra Information - Data Visualization in Python
If you only wanted to work inside Python , there are some modules that do the task:

- Matplotlib : low level, provides lots of freedom
- Pandas Visualization : easy to use interface, built on Matplotlib
- Seaborn : high level interface, great default styles
- Plotnine : based on R's ggplot2, uses Grammar of Graphics
- Plotly : can make interactive plots

Note that using these methods require understanding of Python Fundamentals.

OUTPUTS
• Learn the meaning of data visualization
• Learn available tools and methods to visualize data
• Learn practicality of visualization tools
• Get introduced to Tableau software
• Get introduced to PowerBI software
• Learn data visualization options at Bosch
• Learn data visualization methods in Python

I have also looked around the building. For food, we may go to nearby shopping center. Also, the work had service and cafeteria services in the past, but they are not active right now and it may be related to the pandemic conditions. At a location side, the building is close to other car companies such as Ford, Mercedes etc. While wondering inside the building, I also saw free monthly Bosch magazines which has many information about car industry, latest Technologies, messages from the board etc. Also, there is one tea/coffee corner per floor, bathrooms and balconies. Also, there is a cloud printer. In these days, I tend to come to the office for 2 days and work remotely for 1 day with personal choice.

WORKING DEPARTMENT	Application and Calibration Engineering Department	PAGE	1-30
FIELD OF WORK	Python Automation, Matlab, SQL, Calibration	DATE	27/06/2022- 11/08/2022

This is my tenth day in my internship. In this day I have found, some standards for Electrical vehicles from different organizations. Please note that these are my referance files.

IEC

Category	Standard Number	Name of the standard	Current Year
Secondary Lithium Cells	IEC 62660-1	Secondary lithium-ion cells for the propulsion of electric road vehicles – Part 1: Performance testing	2018-12
	IEC 62660-2	Secondary lithium-ion cells for the propulsion of electric road vehicles – Part 2: Reliability and abuse testing	2018-12
	IEC 62660-3	Secondary lithium-ion cells for the propulsion of electric road vehicles – Part 3: Safety requirements	2016-08
Swapping Station	IEC TS 62840-1	Electric vehicle battery swap system – Part 1: General and guidance	2016-07
	IEC 62840-2	Electric vehicle battery swap system – Part 2: Safety requirements	2016-10
	IEC PAS 62840-3	Electric vehicle battery swap system – Part 3: Particular safety and interoperability requirements for battery swap systems operating with removable RESS/battery systems	2021 -09
Wireless Charging	IEC 61980-1	Electric vehicle wireless power transfer (WPT) systems – Part 1: General requirements	2020-11
	IEC TS 61980-2	Electric vehicle wireless power transfer (WPT) systems – Part 2: Specific requirements for communication between electric road vehicle (EV) and infrastructure	2019-06
	IEC TS 61980-3	Electric vehicle wireless power transfer (WPT) systems – Part 3: Specific requirements for the magnetic field wireless power transfer systems	2019-06
EV Charging System	IEC 61851-1	Electric vehicle conductive charging system – Part 1: General requirements	2017-02
	IEC 61851-21-1	Electric vehicle conductive charging system – Part 21-1: Electric vehicle on-board charger EMC requirements for conductive connection to an AC/DC supply	2017-06
	IEC 61851-21-2	Electric vehicle conductive charging system – Part 21-2: Electric vehicle requirements for conductive connection to an AC/DC supply – EMC requirements for off-board electric vehicle charging systems	2018-04
	IEC 61851-23:2014	Electric vehicle conductive charging system - Part 23: DC electric vehicle charging station	2014-03
	IEC 61851-24	ELECTRIC VEHICLE CONDUCTIVE CHARGING SYSTEM - PART 24: DIGITAL COMMUNICATION BETWEEN A D.C. EV CHARGING STATION AND AN ELECTRIC VEHICLE FOR CONTROL OF D.C. CHARGING	2014-03
	IEC 61851-25	Electric vehicle conductive charging system – Part 25: DC EV supply equipment where protection relies on electrical separation	2020-12
	IEC 62196-1:2014	Plugs, socket-outlets, vehicle connectors and vehicle inlets - Conductive charging of electric vehicles - Part 1: General requirements	2014-06
	IEC 62196-2	Plugs, socket-outlets, vehicle connectors and vehicle inlets – Conductive charging of electric vehicles – Part 2: Dimensional compatibility and interchangeability requirements for a.c. pin and contact-tube accessories	2016-02
	IEC 62196-3	Plugs, socket-outlets, vehicle connectors and vehicle inlets – Conductive charging of electric vehicles – Part 3: Dimensional compatibility and interchangeability requirements for d.c. and a.c./d.c. pin and contact-tube vehicle couplers	2014-06
	IEC TS 62196-3-1	Plugs, socket-outlets, vehicle connectors and vehicle inlets – Conductive charging of electric vehicles – Part 3-1: Vehicle connector, vehicle inlet and cable assembly for DC charging intended to be used with a thermal management system	2020-03
IP Rating	IEC 60529	Degrees of protection provided by enclosures (IP Code)	2013-08
Electric Motors	IEC 60349-2	Electric traction – Rotating electrical machines for rail and road vehicles – Part 2: Electronic converter-fed alternating current motors	2010-10
	IEC 60349-4	Electric traction – Rotating electrical machines for rail and road vehicles – Part 4: Permanent magnet synchronous electrical machines connected to an electronic converter	2012-12
	IEC/TS 60349-3	Electric traction – Rotating electrical machines for rail and road vehicles – Part 3: Determination of the total losses of converter-fed alternating current motors by summation of the component losses	2010-03
	IEC 60034-1:2017	Rotating electrical machines - Part 1: Rating and performance	2017-05

ISO

Category	Standard Number	Name of the standard	Current Year
Safety specifications	ISO 6469-1	Electrically propelled road vehicles — Safety specifications — Part 1: Rechargeable energy storage system (RESS)	2019-04
	ISO 6469-2	Electrically propelled road vehicles — Safety specifications — Part 2: Vehicle operational safety	2018-02
	ISO 6469-3	Electrically propelled road vehicles — Safety specifications — Part 3: Electrical safety	2018-10
	ISO 6469-4	Electrically propelled road vehicles — Safety specifications — Part 4: Post crash electrical safety	2015-09
Performance tests	ISO 8715:2001	Electric road vehicles — Road operating characteristics	2001-06
	ISO 8714	Electric road vehicles — Reference energy consumption and range — Test procedures for passenger cars and light commercial vehicles	2002-11
Battery packs	ISO 12405-1	Electrically propelled road vehicles — Test specification for lithium-ion traction battery packs and systems — Part 1: High-power applications	2011-08
	ISO 12405-2	Electrically propelled road vehicles — Test specification for lithium-ion traction battery packs and systems — Part 2: High-energy applications	2012-07
	ISO 12405-3	Electrically propelled road vehicles — Test specification for lithium-ion traction battery packs and systems — Part 3: Safety performance requirements	2014-05
	ISO 12405-4	Electrically propelled road vehicles — Test specification for lithium-ion traction battery packs and systems — Part 4: Performance testing	2018-07
Vehicle to grid communication interface	ISO 15118-1	Road vehicles — Vehicle to grid communication interface — Part 1: General information and use-case definition	2019-04
	ISO 15118-2	Road vehicles — Vehicle-to-Grid Communication Interface — Part 2: Network and application protocol requirements	2014-04
	ISO 15118-3	Road vehicles — Vehicle to grid communication interface — Part 3: Physical and data link layer requirements	2015-05
	ISO 15118-4	Road vehicles — Vehicle to grid communication interface — Part 4: Network and application protocol conformance test	2018-02
	ISO 15118-5	Road vehicles — Vehicle to grid communication interface — Part 5: Physical layer and data link layer conformance test	2018-05
	ISO 15118-8	Road vehicles — Vehicle to grid communication interface — Part 8: Physical layer and data link layer requirements for wireless communication	2020-09
	ISO 17409	Electrically propelled road vehicles — Conductive power transfer — Safety requirements	2020-02
	IEC 62752	In-cable control and protection device for mode 2 charging of electric road vehicles (IC-CPD)	2018-09
	ISO 18300:2016	Electrically propelled vehicles — Test specifications for lithium-ion battery systems combined with lead acid battery or capacitor	2016-11
	ISO 19363	Electrically propelled road vehicles — Magnetic field wireless power transfer — Safety and interoperability requirements	2020-04
	ISO 20762	Electrically propelled road vehicles — Determination of power for propulsion of hybrid electric vehicle	2018-09
Voltage class B systems	ISO 21498-1	Electrically propelled road vehicles — Electrical specifications and tests for voltage class B systems and components — Part 1: Voltage sub-classes and characteristics	2021-01
	ISO 21498-2	Electrically propelled road vehicles — Electrical specifications and tests for voltage class B systems and components — Part 2: Electrical tests for components	2021-03
Electric Motor, Inverter, DC DC Converter	ISO 21782-1	Electrically propelled road vehicles — Test specification for electric propulsion components — Part 1: General test conditions and definitions	2019-08
	ISO 21782-2	Electrically propelled road vehicles — Test specification for electric propulsion components — Part 2: Performance testing of the motor system	2019-08
	ISO 21782-3	Electrically propelled road vehicles — Test specification for electric propulsion components — Part 3: Performance testing of the motor and the inverter	2019-08
	ISO 21782-4	Electrically propelled road vehicles — Test specification for electric propulsion components — Part 4: Performance testing of the DC/DC converter	2021-05
	ISO 21782-5	Electrically propelled road vehicles — Test specification for electric propulsion components — Part 5: Operating load testing of the motor system	2021-05
	ISO 21782-6	Electrically propelled road vehicles — Test specification for electric propulsion components — Part 6: Operating load testing of motor and inverter	2019-08
	ISO 21782-7	Electrically propelled road vehicles — Test specification for electric propulsion components — Part 7: Operating load testing of the DC/DC converter	2021-05
IP Rating	ISO 20653	Road vehicles — Degrees of protection (IP code) — Protection of electrical equipment against foreign objects, water and access	2013-02

I have also found SAE and UL standards from my Udemy course related to electrical vehicles. These vehicles have advantages over internal combustion engines. Some advantages are:

- They have higher acceleration performance.
- But they may have less top speed.
- Many equipments are fit inside Powertrain boxes in front and back.
- They do not use fuel and have less carbon foot print.
- They have more stable torque curves.
- With less torque you may reach more power.
- Torque may be kept stable.
- They use electricity and with home charging prices are lower.

Company Representative Approval	Title Name, Surname	Application and Cal. Eng. Selim Çökren	Company Signature Stamp	

Today, I have also started to work on two presentatio projects. One project is related with the history of automobiles and the other is related to mobility. Mobility may be explained and be similar as solutions that enable to imporve mobility, moving from one place to other with vehicles. For example, autonomous driving is a mobility solution I also want to add other EV standards as following:

Category	Standard Number	Name of the standard	Current Year
	J1634	Battery Electric Vehicle Energy Consumption and Range Test Procedure	2021-04
	J1715	Hybrid Electric Vehicle (HEV) and Electric Vehicle (EV) Terminology	2021-05
	J2464	Electric And Hybrid Electric Vehicle Rechargeable Energy Storage System (RESS) Safety And Abuse Testing	2021
	J2758	Determination of the Maximum Available Power from a Rechargeable Energy Storage System on a Hybrid Electric Vehicle	2018-12
Use cases Charger Vehicle Communication	J2836-1	Use Cases for Communication Between Plug-in Vehicles and the Utility Grid	2019-07
	J2836-2	Use Cases for Communication between Plug-in Vehicles and Off-Board DC Charger	2011-09
	J2836-3	Use Cases for Plug-In Vehicle Communication as a Distributed Energy Resource	2017-01
	J2836-4	Use Cases for Diagnostic Communication for Plug-in Electric Vehicles	2021-06
	J2836-5	Use Cases for Customer Communication for Plug-in Electric Vehicles	2015-05
	J2836-6	Use Cases for Wireless Charging Communication for Plug-in Electric Vehicles	2021-04
	J2841	Utility Factor Definitions for Plug-In Hybrid Electric Vehicles Using Travel Survey Data	2010-09
Communication Charger and Vehicle	J2847-1	Communication for Smart Charging of Plug-in Electric Vehicles Using Smart Energy Profile 2.0	2019-08
	J2847-2	Communication Between Plug-In Vehicles and Off-Board DC Chargers	2015-04
	J2847-3	Communication for Plug-in Vehicles as a Distributed Energy Source	2021-03
	J2847-6	Communication for Wireless Power Transfer Between Light-Duty Plug-in Electric Vehicles and Wireless EV Charging Stations	2020-09
	J2931-1	Digital Communications for Plug-in Electric Vehicles	2014-12
	J2931-4	Broadband PLC Communication for Plug-in Electric Vehicles	2014-10
	J2931-7	Security for Plug-In Electric Vehicle Communications	2018-02
	J2931-6	Signaling Communication for Wirelessly Charged Electric Vehicles	2015-08
Requirements for EVSE	J2953-1	Plug-In Electric Vehicle (PEV) Interoperability with Electric Vehicle Supply Equipment (EVSE)	2013-10
	J2953-2	Test Procedures for the Plug-In Electric Vehicle (PEV) Interoperability with Electric Vehicle Supply Equipment (EVSE)	2014-01
	J1797	Recommended Practice for Packaging of Electric Vehicle Battery Modules	2016-08
	J1798	Recommended Practice for Performance Rating of Electric Vehicle Battery Modules	2019-11
	J2288	Life Cycle Testing of Electric Vehicle Battery Modules	2020-11
	J2289	Electric-Drive Battery Pack System: Functional Guidelines	2021-08
	J2380	Vibration Testing of Electric Vehicle Batteries	2013-12
	J2936	SAE Electrical Energy Storage Device Labeling Recommended Practice	2012-12
	J2950	Recommended Practices for Shipping Transport and Handling of Automotive-Type Battery System - Lithium Ion	2020-06
	J1772	Electric Vehicle and Plug in Hybrid Electric Vehicle Conductive Charge Coupler	2017-10
	J1773	Electric Vehicle Inductively Coupled Charging	2014-06
EVSE Power Quality	J2894-1	Power Quality Requirements for Plug-In Electric Vehicle Chargers	2019-01
	J2894/2	Power Quality Test Procedures for Plug-In Electric Vehicle Chargers	2015-03
Charging system Energy Transfer	J2293-1	Energy Transfer System for Electric Vehicles - Part 1: Functional Requirements and System Architectures	2014-02
	J2293-2	Energy Transfer System for Electric Vehicles - Part 2: Communication Requirements and Network Architecture	2014-02
EMI/EMC	J1113-11	Immunity to Conducted Transients on Power Leads	2017-06
	J1113-26	Electromagnetic Compatibility Measurement Procedure for Vehicle Components - Immunity to AC Power Line Electric Fields	2014-04
	J551-1	Performance Levels and Methods of Measurement of Electromagnetic Compatibility of Vehicles, Boats (up to 1 5 m), and Machines (1 6.6 Hz to 1 8 GHz)	2020-01
	J551-5	Performance Levels and Methods of Measurement of Magnetic and Electric Field Strength from Electric Vehicles, 1 50 kHz to 30 MHz	2017-11
	J551-16	Electromagnetic Immunity - Off-Vehicle Source (Reverberation Chamber Method) - Part 1 6 - Immunity to Radiated Electromagnetic Fields	2017-10
	J2889-1	Measurement of Minimum Noise Emitted by Road Vehicles	2015-11
	J2929	Safety Standard for Electric and Hybrid Vehicle Propulsion Battery Systems Utilizing Lithium-based Rechargeable Cells	2013-02
	J2344	Guidelines for Electric Vehicle Safety	2020-10
	J1766	Recommended Practice for Electric, Fuel Cell and Hybrid Electric Vehicle Crash Integrity Testing	2014-01
	J2990	Hybrid and EV First and Second Responder Recommended Practice	2019-07
Connectors	J1742	Connections for High Voltage On-Board Vehicle Electrical Wiring Harnesses - Test Methods and General Performance Requirements	2010-03
	J2223-1	Connections for On-Board Road Vehicle Electrical Wiring Harnesses - Part 1: Single-Pole Connectors - Flat Blade Terminals - Dimensional Characteristics and Specific Requirements	2011-12
	J2223-3	Connections for On-Board Road Vehicle Electrical Wiring Harnesses - Part 3: Multipole Connectors - Flat Blade Terminals - Dimensional Characteristics and Specific Requirements	2011-12

Category	Standard Number	Name of the standard	Current Year
	UL 2596 2022-01	Test Method for Thermal and Mechanical Performance of Battery Enclosure Materials	2022
	UL 467	Grounding and Bonding Equipment	2022
	UL 83A	Fluoropolymer Insulated Wire	2021
	UL 62133-2	Secondary Cells and Batteries Containing Alkaline or Other Non-Acid Electrolytes – Safety Requirements for Portable Sealed Secondary Cells, and for Batteries Made from Them, for Use in Portable Applications – Part 2: Lithium Systems	2020
	UL 2231-1	Personnel protection systems for EV Supply circuits general requirements	2021-09
	ANSI/CAN/UL/ULC 2580:2021	Batteries for Use In Electric Vehicles	2021
	UL 2231-2	Personnel Protection Systems for Electric Vehicle (EV) Supply Circuits: Particular Requirements for Protection Devices for Use in Charging Systems	2020
	UL 1004-1	Rotating electrical machines - General Requirements	2020-11
	UL 83	Thermoplastic-Insulated Wires and Cables	2020
	UL 1642	STANDARD FOR SAFETY - Lithium Batteries	2020
	UL 2594	Electric Vehicle Supply Equipment - Safety Standard	2016
	UL 2202	Electric Vehicle (EV) Charging System Equipment	2018
	UL 2251	Plugs, Receptacles, and Couplers forElectric Vehicles	2017
	ANSI/CAN/UL/ULC 2271,	Batteries for Use In Light Electric Vehicle (LEV) Applications	2018

Today, I have prepared a list of ideas to present in mobility solutions as such:

1. Autonomy Levels of Driving
2. Amount of Data in a Second
3. Electrical Vehicle Myths
4. Auto-Window Electrical System
5. Automated Valet Parking
6. Comfort Actuators
7. Changing Gear, Brake from Computer and Cybersecurity

My colleagues have also prepared ideas about VCU, eAXLE and such. I also learned that from an external sourse that more than 4000GB may have been transferred per second in an autonomous car. I also studied on tkinter and pyinstaller module on Python. Both of these modules are very practical. Tkinter is used to produce user interfaces in Python where pyinstaller is used to onvert a python script (with all required dependencies) to an executable file. I like both of these modules. My code for a equipment tracking app (partially completed is attached below, I worked on this code as well)

```
# -*- coding: utf-8 -*-"""
from datetime import datetime
startTime = datetime.now()
import openpyxl as op
import tkinter as tk
from tkinter import ttk
path = r"confidential.xlsx"
window = tk.Tk()
# window.geometry("600x600")
window.title("Lab Tracking")
window.columnconfigure(0,weight=1)
window.columnconfigure(1,weight=10)
# First screen enter account via admin given password
login_name_label = tk.Label(window,text="Name:",bg="purple",fg="white",anchor="w")
login_pass_label = tk.Label(window,text="Password:",bg="purple",fg="white",anchor="w")
login_name_label.grid(row=0, column=0, sticky="ew")
login_pass_label.grid(row=1, column=0, sticky="ew")
# Define the style for combobox widget
style= ttk.Style()
style.theme_use('clam')
style.configure("TCombobox",fieldbackground="orange",background="white")
login_name_list = ["Name1", "Name2", "Name3", "Name4", "Name5"]
login_name_comb = ttk.Combobox(window, values = login_name_list)
login_name_comb.grid(row=0, column=1, sticky="ew")
login_pass_entry = tk.Entry(window, bg="navy", fg="white")
login_pass_entry.grid(row=1, column=1, sticky="nsew")
```

In these days, I wrote two automation scripts for my friends. Both of them use openpyxl, thus both of them work with Excel files. We write or gather information from Excel. First script is:

```
import openpyxl

def compare_write(file_path:str, ref_sheet:int, ref_tab:int, ref_offset:int, \
                  target_sheet:int, target_ref:int, target_tab:int,         \
                  target_offset:int, final_tab:int):
    """ compares two columns, if equal copy a column to an empty column """

    # define workbook variable by reading file at path
    wb = openpyxl.load_workbook(file_path, read_only= False)

    # construct specified sheet objects
    ws_ref = wb[ref_sheet]; ws_target = wb[target_sheet]

    # collect reference values as a list
    indices = [ws_ref.cell(row=i,column= ref_tab).value \
               for i in range(ws_ref.min_row + ref_offset, ws_ref.max_row + 1)]

    for i in range(len(indices)): # repeat reference element number times
        for i2 in range(ws_target.min_row+target_offset, ws_target.max_row+1):
            if indices[i] == ws_target.cell(row=i2,column=target_ref).value:

                flag = 1 # check if equal or not and define if condition flag
                # write equal row index values in reference sheet under title
                ws_ref.cell(row=(i+ref_offset+1),column=final_tab).value = \
                ws_target.cell(row=i2,column=target_tab).value
                break

    if flag == 1: # if flag is up, write title cell
        # write the title in reference sheet, specified column
        ws_ref.cell(row= ref_offset, column= final_tab).value = \
        ws_target.cell(row= target_offset,column= target_tab).value

    wb.save(file_path) # save the final file at same path
```

This script uses an Excel file as an input. The aim of compare_write function is to collect indices of reference column (indices) inside reference sheet, compare indices to another sheet's index values. If index values are same, in that row, collects value in a specified tab and writes in a specified tab in reference sheet. Also this script supports different header placements.

After completing this script I felt happy and gave it to my friend as .py file, because he has also Python installed in his computer. If he did not, I may have needed to use pyinstaller module for .exe file.

I wrote the second automation script for my other friend. She left to g oto Mercedes in Germany later. She know German and was very hardworking. Applied for internship post and got accepted. Script is:

```
"""
   This script uses an Excel file as an input. The aim of conditional_deleter function
   is to collect row indices where column x and column y row values are equal. Then
   uses the index list to delete excel rows iteratively while reducing indices.
"""

import openpyxl

def conditional_deleter(file_path:str, sheet_name:str, col_x:int, col_y:int, offset=1):
   """ in excel file,in row, compare column x and y values. If equal, delete row """

   # define workbook variable by reading file at path
   wb = openpyxl.load_workbook(file_path, read_only=False)

   # construct specified sheet's object
   ws = wb[sheet_name]

   # find indices where column 1 and 2 values are equal
   indices = [i for i in range(ws.min_row+offset, ws.max_row+1)\
            if ws.cell(row=i,column=col_x).value == ws.cell(row=i,column=col_y).value]

   # delete rows while reducing index values with order (because row numbers decrease)
   for i2 in range(len(indices)): ws.delete_rows(indices[i2]-i2)

   # save the workbook in specified path
   wb.save(file_path)

""" Example script run scenario: """
# In desktop file, in sheet 1, compare column index 1 and 2 and delete rows
# conditional_deleter(r"C:\Users\Smith\Desktop\Test.xlsx" ,"Sheet1", 1, 2)
```

I may have gotten confused about the order of the automation scripts, I think the script in previous page was the script that I have prepared for her. While preparing these scripts (as well) I have realized that coding and trying to solve problems feels like solving a puzzle for me. I think the reason fort hat is that because I love my job. I have found my job in a very late age but I am very happy. Coming back to the subject, my friend later on used Excel formulas to perform the task and since then became a friend of mine for me. These scripts are conditional scripts. The layout and purpose of the script may change with the excal file. The autonomy reduces and hard code increases. I want to prepare a general autonomous Excel python API in one day if I can. I want to try to make a flexible program. Afterwards, I looked at my emails, performed remaining tasks and at around 6 pm left work again.

Today, I used MATLAB to prepare a alphabetical sorting program for my need. The code is as:

```
function [] = alp_one(w_nm)

% This function sorts strings alphabetically.
% User needs to enter word number as an input.
% With input, the function gets that number of
% variables from the user and prints as sorted.
% If sentences are entered as well, only uses
% the first characters on them for reference.
% Numbers, empty spaces are supported.

% Initial; variables and a newline
i = 1;
fprintf("\n")
str = strings(1 , w_nm);

while(w_nm > 0)
   str(i)=input('Please enter the word: ' , "s");
   w_nm = w_nm - 1;
   i = i + 1;
end

str_fin = sort(str);
fprintf("\n")
fprintf('Alphabetically sorted words are: \n')

for k=1:1:length(str_fin)
   fprintf('%d) %s \n',k,str_fin(k))
end

fprintf("\n")
end
```

I did have experience about Matlab and Simulink in school. I needed to look back at some guides to remember how to make a loop and fprint. I felt happy to change the coding editör for one day to make a different job for one day. These days, in the meantime I attend to weekly and monthly meetings. We had also a department workshop, where we wrote different things and shared ideas among ourselves. These days, smart work conditions were changing in other words, workers may need to come to office in a minimum limit. This may be measured with card reader systems. I have also prepared charts about a project that I am working in. Also, I tried to make a workflow sample of the general process that a calibration engineer inside our department may be in. I got heavy help from my colleagues about these subjects. My Indian colleague left Turkey to go back to India, he was very knowledgeable.

In these days, I have started to look into SQL. I have started some trainings about SQL in Udemy. To connect to SQL from Python I have prepared the following script:

```
from datetime import datetime
startTime = datetime.now()

import pandas as pd
import pyodbc as po

cnxn_str = ("Driver={confidential};"
        "Server=confidential;"
        "Database=TutorialDB;"                          # Change database name
        "Trusted_Connection=yes;"
        "UID=confidential;")                        # Change user name - no password

cnxn = po.connect(cnxn_str)
cursor = cnxn.cursor()

data = pd.read_sql("SELECT * FROM Customers",cnxn)
# Change Customers to your table name

# ----------------------------------------------------------
# pyodbc.drivers() -> To learn possible driver names to input
# pyodbc is no longer continued, so using sqlalchemy is recom.

print("Program runtime: " + str(datetime.now()-startTime))
```

To connect to SQL from Python, beforehand I prepared a new tutorial database and a tutorial table inside the tutorial database. And I have added some rows to analyze. I use this SQL reference:

•Data Manipulation- You will be satisfied after hearing that SQL is especially beneficial at data manipulation. As it permits you to see the specific data and how it functions, you will have a simpler time testing and manipulating the data. Moreover, information compiled in SQL is dynamic like it can be adjusted and controlled whenever using some essential queries.

•SQL Programmers in High Demand- Nowadays, you don't need to worry about your job as a SQL Programmer. There are outstanding opportunities in this field. As you have read above, learning SQL will lead to your better future and there is a high demand for people who know about SQL.

•Data Mining- Learning SQL will permit you to mine data with bigger proficiency. By utilizing essential inquiries you can distinguish explicit information at time stretches, see update occasions, screen table action, and significantly more. Hence, this by itself ought to be reason enough.

Also SQL is very connected to data science. If I can, I aim to educate myself on SQL with Python.

Today, I have worked on with a calibration team for an automation task. Some parts of script I wrote:

```
# Calculations
        if len(fol_nums) == 3:
            if fol_nums[2] == "#":
                fol_nums = list(range(int(fol_nums[0]),int(fol_nums[1])+1))
                fol_nums = [str(num) for num in fol_nums]

        # Calculations
        rep_lrh = [[key,key.lower(),key.title(),key.upper()] for key in rep_lrh]
        rep_lrh = [item for sublist in rep_lrh for item in sublist]

        # Input Operations
        dir_file = os.listdir(inp_path)
        inp_fold = [name for name in dir_file if name.startswith(tuple(fol_nums))]
        cdr_path = [os.path.join(inp_path, name, cdr_name) for name in inp_fold]

        # Counter
        counter = 0

        # File Loop for Specified Folder Contents
        for main_path in cdr_path:
            report_file = [file for file in os.listdir(main_path) if rep_key in file]
            result_file = [file for file in os.listdir(main_path) if res_key in file]

            if len(report_file) == 0: continue

            report = report_file[0]
            report_path = os.path.join(main_path, report)

            wb_rep = op.load_workbook(report_path)
            ws_rep = wb_rep[rep_wsn]

            if raw_file in ["no", "No", "NO"]:
                for i in range(ws_rep.min_column, ws_rep.max_column+2):
                    col = op.utils.get_column_letter(i)
                    ws_rep.column_dimensions[col].hidden = False
                for i in range(ws_rep.min_row+2, ws_rep.max_row+1):
                    ws_rep.row_dimensions[i].hidden = False
```

Script continues and as a different fact, total line number is 365 lines. After compleitng the script, I have also prepared a user guide file. In the user guide, I have tried to explain modes of operation, default values, where to find more information and who to connect etc. After the project folder has been completed, I have sent a mail to all calibration team members to notify them with fol. path.

Today, from previous calibration team, I have worked on another automation task concerning another manueal task that needs to be automated. Some parts of my script as is: (Fitting and un-confidential)

```
for i in range(ws.min_row+2, ws.max_row+1):
    if ws[f"F{i}"].value == "missing" and ws.row_dimensions[i].hidden == False:

        funct = ws[f"A{i}"].value
        label = ws[f"D{i}"].value
        newvs = ws[f"C{i}"].value

        if funct == ""   or label == ""  : continue
        if funct is None or label is None: continue

        sw_name = [name for name in sw_list if name.startswith(funct)]
        sw_name = [name for name in sw_name if newvs in name]
        if len(sw_name) == 0: continue

        sw_ver = [name.split(funct + "_")[1] for name in sw_name]
        sw_ver = [nm.split("_")[0] + "." + nm.split("_")[1] for nm in sw_ver]

        latest_version = max(sw_ver, key=version.parse)
        latest_index = sw_ver.index(latest_version)

        sw_name = sw_name[latest_index]
        sw_path = os.path.join(sw_dir, sw_name)

        if sw_name in list(sw_dict.keys()): text = sw_dict[sw_name]

        # ---------------------------------------------------------------

        if sw_name not in list(sw_dict.keys()):

            pdfFileObj = open(sw_path, 'rb')
            pdfReader = p2.PdfFileReader(pdfFileObj)
            pageNum = pdfReader.numPages

            text = ""
            for i in range(pageNum): text += pdfReader.getPage(i).extractText()

        sw_dict[sw_name] = text
```

After finishing, I prepared a new user guide and sent mails to the calibration members. This script is focused on getting data from an Excel file, reading a PDF file as multiline string, searching string for some words and writing the found word values to the Excel file again.

Today, I have prepared another automation program for a friend. (To be honest sometimes some days pass to complete these scripts, but my work days were different and I want to try to keep the flow of one subject and not jump from one subject to another) Some areas of my script is as:

```
from datetime import datetime
startTime = datetime.now()

import os
import sys
import tkinter as tk
import openpyxl as op

global window
window = tk.Tk()
window.geometry("500x60")
window.resizable(True, False)
window.title("Program Settings")

window.rowconfigure(0, weight=1)
window.rowconfigure(1, weight=1)

window.columnconfigure(0,weight=1)
window.columnconfigure(1,weight=19)

label1=tk.Label(window,text="Path: ",anchor="w")
label1.grid(row=0, column=0, sticky="ew")

entry1=tk.Entry(window)
entry1.insert(tk.END, os.getcwd())
entry1.grid(row=0, column=1, sticky="ew")

def get_info():

    dir = entry1.get().strip()
    all_file = os.listdir(dir)

    csv_file = [filename for filename in all_file if filename.endswith('.xlsx')]
    if len(csv_file) != 1: window.destroy(); sys.exit("Multiple Excel files! Exiting...")
    csv_name = csv_file[0]; csv_path = os.path.join(dir, csv_name)
    wb = op.load_workbook(csv_path); ws = wb.active
```

This script is on hold. This script gets into a Excel file, gets some name strings, gets into a .txt like file and scans the text for the names and gets the value after some lines and writes back to the excel file. Some naming decisions are needed to continue the script. I want to use regex in this code if I can.

Today I trained on some training material. Our team has prepared orientation files for new workers. These files help to start more quickly on different work areas such as VBA, Python, Data Science, SQL, Web modules etc. I have looked into data science training as:

Introductionof Data Science
- Data Science is about data gathering, analysis and decision-making.
- Data Science is about finding patterns in data, through analysis, and make future predictions.

Calculus
- Calculus is a intrinsic field of maths and especially in many machine learning algorithms that you cannot think of skipping this course to learn the essence of Data Science.

Linear Algebra
- Linear algebra is foundational in data science and machine learning. Beginners starting out along their learning journey in data science, as well as established practitioners, must develop a strong familiarity with the essential concepts in linear algebra.

Probability and Statistics
- Probability and Statistics form the basis of Data Science. The probability theory is very much helpful for making the prediction. Estimates and predictions form an important part of Data science. With the help of statistical methods, we make estimates for the further analysis. Thus, statistical methods are largely dependent on the theory of probability. And probability and statistics is dependent on Data.

Matplotlib
- Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible.

NumPy
- NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.

Pandas
- pandas is an opensource, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language.

Later, I want to study data science subjects and try to be a data scientist. I did attend to machine learning courses on Udemy and deep learning course. I also tried to complete some self paced training about Python and data science in Kaggle. If I can, I want to enter artificial intelligence competitions in and outside of Bosch. I need to study these modules, and practice using machine learning and deep learning on different datasets. Image recognition is another subject that I like.

Today, I checked advanced coding expectations. The expectations are as:

- How & when to use "standard" language elements in production ready quality, including good understanding of
  - Basic Python data types (scalars, collections, ...)
  - Python-specific syntax elements (list comprehensions, decorators, context managers, ...)
  - Argument passing semantics
  - Basic OOP (e.g. classes, inheritance, abstract basis classes)
  - Basic functional paradigms (lambdas, closure semantics, ...)
  - Dunder-methods (operator overloading, ...)
- Familiarity with Python package system (pip, venvs, requirements files and pinning, ...)
- Familiarity with core Python standard library
- High proficient with a handful of libraries in the Python ecosystem (data science: numpy, pandas, ..., backend: flask, django, fastapi, validation: pydantic, cloud specific libraries, database libraries, visualization libraries)
- Basic knowledge of how to extend Python with native code
- Solid understanding of testing frameworks like pytest
- Readily applies language elements mentioned above to implement desired functionality based on standard implementation patterns (no extensive trial-and-error)
- Ability to write testable code
- Ability to re-use/extend existing code written by colleagues
- Clean Code mindset: write easily readable/understandable and re-usable code, e.g. by avoiding magic numbers, usage of "telling names", avoid confusing data or execution flow, misleading/insufficient documentation
- Ability to debug SW using different debuggers (e.g. pdb / repl based debugging)

After thinking about requirements, I also thought back on job post requirements that were related with Python work areas. They also require similar technical skills and different ones. Upper level reqs are:

- Standard elements mentioned above, but on deeper level.
- Specific knowledge in advanced topics like
  - Type annotations and type checking (mypy, pyright, ...)
  - Native binding generators like pybind11
  - Advanced OOP concepts (e.g. multi-inheritance, traits, meta class)
  - Runtime inspection
  - Advanced features in testing frameworks (complex mocking, parametrization, configuration, reporting, coverage)
  - Good understand of Python version differences and cross version targeting

I also saw requirements such as SQL, Linux, software test tools, portfolio, Git knowledge, Artificial intelligence related requirements, Docker, Jenkins, web development frameworks and so on.

Today I have prepared some SQL commands guide. My code as is:

```
--To show all available datasets (*)
select name, database_id, create_date
from sys.databases;
-- To make a new database in the server
create database tutorial_database
-- To use a database (enter)
use tutorial_database;
-- To see all tables in a database
select * from tutorial_database.information_schema.tables;
-- To make a new table in database
create table tutorial_table
(
                    Date_Taken datetime,
                    Person_Name varchar(20),
                    Equipment_Name varchar(25),
                    Returned varchar(3)
)

-- To see all content from a table
select * from tutorial_table;
-- To add columns to a table
alter table tutorial_table
add return_date datetime;
-- To drop columns in a table
alter table tutorial_table
drop column return_date;
-- To add a row to the table
insert into tutorial_table(Date_Taken,Person_Name, Equipment_Name, Returned)
values('2022-09-01 22.30.10.15','Carter','ECU','No');
-- To delete a row from the table
delete from tutorial_table where Person_Name='Carter'
-- To delete contents inside a table
truncate table tutorial_table;
-- To delete a table in the database
drop table tutorial_table;
-- To switch to different database
use tempdb;
-- To delete a database in the server
drop database tutorial_database;
```

I did prepare this .txt file for self training and for a colleague.

Today, I needed to prepare layouts for our car history presentation. During the day I calculated and recorded coordinate values on Powerpoint such as: (to insert objects)

Left starting point: 3,12 x 3,71 y
Right starting point: 27,17 x 3,71 y
original line: 28,87 cm width
position: 0,81 x 3,83 y
every picture: 3,08 3,09 3,1
slide size: 30,471 cm
slide height: 17,141 cm

Strategy:
First calculate total margins size
Find remaining length
Divide length to sections by picture size
Find middle points of pictures
Slide width : 30,471
Line width: 28,87
Remaining area: 1,601
Line left end from left: 0,81 -> changed to 0,8
Line right end from right: 0,791 -> changed to 0,8
Total new margin area: 1,6
30,471 - 1,6 = 28,871 = line width
There are 6 images on a slide
28,871 / 6 = 4,811833 cm width per image ~ 4,81
First image starts at: 0,8 x 5,28 y
Every picture width, length : 3,11 4,82
Next image starts at: 0,8 + 4,811832 = 5,611833

Line starts at 0,8 from left
First point is at centered at 0,8 + 4,811832/2 = 3,205916
Every summary textbox h and w: 0,3 ve 4,35 -> 4,811832
Every text h/w: 4,47 4,67 every text y: 10,09
Every add date: 0,82 ve 4,83 hw
position: 0,81 ve 4,07 Every point between: 4,811832
Division between 2: 2,405916
First point: 0,8 + 2,405916 = 3,205916 from left
Points (3,71 y)

3,205916 - 8,017748 - 12,82958 - 17,641412 - 22,453244 - 27,265076

Afterwards, I put the objects in coordinates with specified sizes (default and default %125, 150 …) and prepared new powerpoint layouts inside Powerpoint master slide.

After reaching to the Office the same way as yesterday, I took my time again to learn information via computer. Afterwards, my supervisor forwarded a mail to me about presenting an introductory PowerPoint file. I filled some areas in the template and added technical and non technical information. Today, there was a regular department meeting which was being held in once a month during to the end. Before the meeting time, I have completed the introductory presentation which was a one page file. And the time was on fort he meeting. I can say that I was very excited. In our floor, every department had members inside the meeting. My boss wanted me to introduce myself at some point. I explained my views on Python, Artificial Intelligence, Automation and so on. Afterwards, the meeting continued regularly. The nature of this meeting was for departments to explain and present their work during the month. For example they may explain, introduce concepts, they may compare their performance from last month to this month etc. After a long and very informative (though my technical knowledge at that point was less, thus I had a hard time understanding some concepts and projects) meeting, we entered the midday break. After the break, I had the opportunity to see the virtual classroom challenge competition. Several colleagues have developed a new innovative project on automobiles and they joined the competition. The competitors were graded with votes internationally along Bosch workers. The project was related to some methods on improving hardware know-how. Personally, I support all good education projects, and I voted for them who was representing our group. Afterwards, there was another meeting related to remote work. After the pandemic and towards today, people in many offices have started working remotely around the world. The choice to work remotely was present in our office as well. Thus, more than half of the workers were working from home. Personally, I wanted to be in the office to observe working area and communicate with other people. In the meeting, same decisions continued about the remote work program. Today, I also communicated with related personnel to get my building entry card. It was approved. Thus I was going to get my entry card later hopefully. Otherwise, I entered TİSK learning platform and our internal learning platform. I think I managed to complete one training, concerning data protection. The thing was as a corporate worker, I needed to be very careful about not losing my work computer, work documents or information. People might use different tactics to steal information from somebody. We need to make strong passwords, not left computer open (keep It locked) when we are not around, use a vision filter on the computer screen. After these precautions, if our work computer got stolen, we need to report (send a mail) to the related department to inform them and for them to cut the connection. There were a lot of critical information on the training. Nowadays, my supervisor did not appoint me a project, which I was grateful for, because I was learning my surroundings, work area, work conditions, hours, breaks etc. But, I tried to start practicing on Python. From the app, you may install modules, other apps (like Visual Studio Code – another IDE), make a new environment etc. Making a new environment is very critical. While working with Python you may use virtual environment from offline IDE, you may use online IDE (automatized) or you may use offline IDE but only include the modules you want to use with making a new environment.) I also installed additional technical apps, which were not fully in my work area but was related. Again, throughout the day; I tried to communicate with people and learn new information. After the day ended, at home I checked my emails and education program.

Day started as usual, the virtual classroom voting mail came again. Which I had done yesterday. To give about the challenge a little bit more info, the challenge worked with teams around the world, but all team members needed to be internal workers, I guess. Also, there is a deadline to submit ideas and a deadline for voting. Today also the final approval of building entry card had come via mail. Also, I realize I did not write about the duration of my work. Originally, my contract was a long part time work contract, which was for 11 months. In these months, I wanted to fill internship papers. For some apps in software center, I only needed to mark that I want to use them in software center, an automated process takes care about installation process. But if the app is paid, then I need to apply to a license which is done with department funds if approved. Thus, I applied to get a license in MATLAB. The reason was, I thought about no coding artificial intelligence systems (machine learning, statistics, deep learning) toolbox add ones in MATLAB. I do find those things very efficient, because the process is very faster than coding. MATLAB automatically selects several methods while making a prediction based on data type, contents etc. and you need to choose a method, and output with the highest accuracy for example. But fundamental analysis skills and AI knowledge is required while making a selection. Also MATLAB is a very nice tool while working with control subjects. I am not sure if MATLAB is used extensively in our department for now. I think automotive related apps like INCA, MDA and such are used much more because the work area is very similar and apps are easy to implement on the current work and testing. Now I come to what calibration means. Today, I researched about related subjects. Calibration is adding new functionality, input or output to an existing system, or looking for sensor errors, analyzing and informing the customer. One overall process may be explained as, the customer contact us for calibration, if necessary a test personnel goes to the test site, near the car; with related connection equipment and laptop, the test personnel connects to the car, looks for data in real time, (analysis program shows non error signals in green but if a signal has got past its thresholds (min, low, high, max) it is represented in different color, in final error situation the signal is red) observe error signals, check for abbreviations and report the error area to customer, if needed connect to ECU and add new functions etc. Thus, it is closely related to customer relationship. And also there may be work between countries with international Bosch workers. By specification, the building I am working is a center for middle east countries but there are people who work with people in Europe as well. Especially Germany is an interaction center for our country. We also need to know that Bosch is a Germany based company. So, these past days were mainly about learning the job that I am in, work areas, formalities, standards, equipment etc. Some times, I needed to go to Human Resources floor to sign a document as well. A new thing that I did today is I also looked at Portal website. The way it works is that there are internal websites in Bosch which are constructed for usage by personnel. I discovered a community website, which had Python communities within. Also, this website was the one where I gave my vote for the virtual classroom challenge. I joined some communities, looked at some informative posts. Also, throughout the day, I communicated with my buddy many times to learn many things that I am behind in. I also discovered that there are charging systems for work computers in case I forgot my charging cable. Today, there were many new information to be learned.

Today I worked on my code for different projects:

```
for cdr_path in cdr_paths:
    all_file = os.listdir(cdr_path)
    cdr_file = [file for file in all_file if "Confidential" in file]
    res_file = [file for file in all_file if "result" in file]

    for cdr_folder in cdr_file:
        c_path = os.path.join(cdr_path, cdr_file[0])

        wb = op.load_workbook(c_path)
        ws = wb["Calibration details"]
        ws.row_dimensions[1].hidden = False;ws.row_dimensions[2].hidden = False
        for i in range(1, ws.max_column+1): # +1 or not?
            letter = op.utils.get_column_letter(i)
            condition = False if letter in letters else True
            ws.column_dimensions[letter].hidden = condition
        for i in range(3, ws.max_row+1):
            col1 = op.utils.column_index_from_string("H")
            col2 = op.utils.column_index_from_string("D")
            condition1 = False if ws.cell(i, col1).value in checks else True
            condition2 = True if str(ws.cell(i, col2).value).split("_")[0] in labels else False
            condition = True if condition1 or condition2 else False
            ws.row_dimensions[i].hidden = condition

        for res_folder in res_file:
            r_path = os.path.join(cdr_path, res_file[0])
            wb2 = op.load_workbook(r_path)
            ws2 = wb2["Sheet1"]

            e_col = op.utils.column_index_from_string("E")
            g_col = op.utils.column_index_from_string("G")
            d_col = op.utils.column_index_from_string("D")
            j_col = op.utils.column_index_from_string("J")

            for i in range(3, ws.max_row+1):
                for i2 in range(2, ws2.max_row+1):
                    if ws.cell(i,d_col).value == ws2.cell(i2,e_col).value:
                        ws.cell(i,j_col).value = ws2.cell(i2,g_col).value
        wb.save(c_path)
```

After my work hours neared to end, I left the office. Throughout the day I also tried to check my emails and attend to meetings that was assigned to me as well.

Today I worked on user interface part of my code. The parts is as:

```
import os
import tkinter as tk
import openpyxl as op

class  cond:

    def __init__(self):
        """ Set default parameters """
        self.directory_path = os.getcwd()

    def main_process(self):
        """ Main window for input """

        # setting default attributes
        script_path = os.getcwd()
        folder_name = os.listdir(script_path)
        last_folder = " ".join([folder.split("_")[0] for folder in folder_name[-4:-1]])

        # window attributes
        window = tk.Tk()
        window.geometry("700x400") #600*600
        window.title("Operator Details")
        window.resizable(0,0)

        # grid organizer
        window.columnconfigure(0, weight=1)
        window.columnconfigure(1, weight=4)
        window.columnconfigure(2, weight=4)

        #%% label 1 - input directory path
        label_1 = tk.Label(window,text="Input Directory Path:",bg="#ed335f",fg="white", anchor="w")
        label_1.grid(row=0, column=0, sticky=tk.EW)

        ### entry_1 - input directory path
        entry_1 = tk.Entry(window, bg="black", fg="yellow")
        entry_1.insert(tk.END, f'{script_path}')
        entry_1.grid(row=0, column=1, columnspan=2, sticky=tk.NSEW)
```

The script continues, but this was the part I have managed to fit in. Today was another eventful day. Sometimes, our department members go to sports after work. (generally Tuesday) I did not attend to a sports event, if I can, I want to attend one event one day.

Today, I got communicated with another Python developer via our Python community page. He was working on reading images, texts from a pdf file, also reading file from Outlook messages, and paste pdf contents inside Outlook and send mails as far as I can understand. He said that some characters got translated wrongly. (German) and he was working to try to solve the encoding issue.

Today I also worked on regex module:

\d	Most engines: one digit from 0 to 9	file_\d\d	file_25
\d	.NET, Python 3: one Unicode digit in any script	file_\d\d	file_9꧓
\w	Most engines: "word character": ASCII letter, digit or underscore	\w-\w\w\w	A-b_1
\w	.Python 3: "word character": Unicode letter, ideogram, digit, or underscore	\w-\w\w\w	字-ま_۳
\w	.NET: "word character": Unicode letter, ideogram, digit, or connector	\w-\w\w\w	字-ま‿۳
\s	Most engines: "whitespace character": space, tab, newline, carriage return, vertical tab	a\sb\sc	a b c
\s	.NET, Python 3, JavaScript: "whitespace character": any Unicode separator	a\sb\sc	a b c
\D	One character that is not a *digit* as defined by your engine's *\d*	\D\D\D	ABC
\W	One character that is not a *word character* as defined by your engine's *\w*	\W\W\W\W\W	*-+=)
\S	One character that is not a *whitespace character* as defined by your engine's *\s*	\S\S\S\S	Yoyo

+	One or more	Version \w-\w+	Version A-b1_1
{3}	Exactly three times	\D{3}	ABC
{2,4}	Two to four times	\d{2,4}	156
{3,}	Three or more times	\w{3,}	regex_tutorial
*	Zero or more times	A*B*C*	AAACC
?	Once or none	plurals?	plural

Regex module is an amazing module for finding patterns inside text files.

Today, I continued on the regex training:

.	Any character except line break	a.c	abc
.	Any character except line break	.*	whatever, man.
\.	A period (special character: needs to be escaped by a \)	a\.c	a.c
\	Escapes a special character	\.*\+\? \$\^\/\\	.*+? $^/\
\	Escapes a special character	\[\{\(\)\}\]	[{()}]

|	Alternation / OR operand	22|33	33
(…)	Capturing group	A(nt|pple)	Apple (captures "pple")
\1	Contents of Group 1	r(\w)g\1x	regex
\2	Contents of Group 2	(\d\d)\+(\d\d)=\2\+\1	12+65=65+12
(?: …)	Non-capturing group	A(?:nt|pple)	Apple

\t	Tab	T\t\w{2}	T ab
\r	Carriage return character	see below	
\n	Line feed character	see below	
\r\n	Line separator on Windows	AB\r\nCD	AB CD
\N	Perl, PCRE (C, PHP, R…): one character that is not a line break	\N+	ABC
\h	Perl, PCRE (C, PHP, R…), Java: one horizontal whitespace character: tab or Unicode space separator		
\H	One character that is not a horizontal whitespace		
\v	.NET, JavaScript, Python, Ruby: vertical tab		
\v	Perl, PCRE (C, PHP, R…), Java: one vertical whitespace character: line feed, carriage return, vertical tab, form feed, paragraph or line separator		
\V	Perl, PCRE (C, PHP, R…), Java: any character that is not a vertical whitespace		
\R	Perl, PCRE (C, PHP, R…), Java: one line break (carriage return + line feed pair, and all the characters matched by \v)		

All of these symbols mean different things and different situations. Regex is multi language module.

Today was the latest day of my internship. My internship was continuing on the same company in a different department. Thus it was not a goodbye to colleagues. So, I worked like other days, where I check email messages attend meetings and try to write codes. Today I wrote these lines:

```
%% Import libraries
import openpyxl as op  # To read excel,outline level
import shutil  as sh # To delete files if required
import pandas as pd # To read sheet as dataframe
import numpy as np # To search string in array
import json as js # To make json file at end
import sys as sy # To message and exit app
import os as os # To access system files
import re

#%% Import modules for Logging
import datetime as dt # To save time marks
from openpyxl.styles.borders import Border, Side # To format log entries
from openpyxl.styles import Font, Color, Fill, PatternFill # To format log
pd.options.mode.chained_assignment = None # To avoid getting copy warnings

#%% All functions in first usage order in script- Scroll past to code
def start_logger(log_path:str):
    """ Makes blank log Excel file with headers and saves """
    log_wb = op.Workbook()
    log_ws = log_wb.active
    label1, label2, label3 = "Timestamp" , "Message" , "Type"
    label4, label5 = "Consequence/Result", "Suggested Action"
    log_ws.append([label1, label2, label3, label4, label5])
    column_widths = {"A" : 20, "B": 60, "C": 10, "D" : 60, "E": 60}
    for c,w in column_widths.items(): log_ws.column_dimensions[c].width = w
    headers = ["A1", "B1", "C1", "D1", "E1"]
    for tab in headers:
        log_ws[tab].font = Font(bold = True)
        log_ws[tab].fill = PatternFill(start_color = "BCBCBC", fill_type = "solid")
        log_ws[tab].border = Border(bottom = Side(style = 'thick'))
    log_wb.save(log_path)
```

This is the completion of Vocational I internship. I had the opportunity to work with very smart, hardworking and helpful people. I also like my job and the company I am working with.In this internship, I have learned about Python, VBA, SQL, machine learning, automative technical programs, calibration, automative equipments etc. My supervisor had left to join Bosch Germany AI team. I think that is a very nice place to work as well. I continue my internship in Vocational II internship.