<u>Block 1. General Skills</u>

Module 1. Documentation
1.1 OneNote
Task: Make sure you have access to the <u>OneNote</u>. Look at this example of the <u>Weekly Meeting</u> template, and complete the following.
1. Open OneNote
2. Click "SOP"
3. Click "Weekly Meeting Example"
4. Make a copy of it and move it to "Onboarding"
5. Rename the page to your first and last name.
6. Find the contact information of your subsystem lead or director. Send a message/email asking for access to Jira.
7. Go back to OneNote, and at the bottom of your personal page, type "I have requested access to Jira by contacting [who you contacted]."

Module 2. Introductory Programming
2.1 Python Basics
Python is an essential programming language for software development in neurotechnology. It is easy to learn and use, making it a great tool for processing neural data, building machine learning models, and automating research tasks. With Python, you can develop skills in data analysis, signal processing, and algorithm design, which are key for working with brain-computer interfaces, neuroimaging, and neural simulations. Its libraries, like NumPy, SciPy, and TensorFlow, help with handling large datasets and applying AI to neuroscience problems. Knowing Python allows you to quickly test ideas, analyze brain signals, and create software for neurotech applications.

Task: Go through this <u>course</u>. Submit a picture of your completion screen/certificate. If you are proficient in Python, submit a screenshot of a previous project instead. If you are stuck at any point, do not hesitate to use the internet or ask for help. However, please refrain from using generative AI during the preliminary stages of learning Python.

<u>Additional resources</u>:
- <u>Tutorial to Download VSCode</u>
- ▶ Python Programming Tutorial | Introduction | GeeksforGeeks
- <u>scikit-learn Machine Learning in Python</u>
- <u>Dive into Deep Learning Book</u>

2.2 Numpy and Time Series
<u>Introduction to Python Libraries</u>:

<u>NumPy</u>: NumPy is a fundamental package for scientific computing in Python. It provides support for large, multi-dimensional arrays and matrices.

```
In [1]:   import numpy as np
```

<u>SciPy</u>: SciPy builds on NumPy and provides additional functionality for scientific computing, including signal processing tools.

```
In [2]:   from scipy import signal
```

<u>MNE-Python</u>: MNE-Python is an open-source package for processing EEG and MEG data.

```
In [3]:   import mne
```

Task: Watch this <u>video</u> to learn about how to use Numpy.

Module 3. Data Analysis
3.1 Statistics
Statistics play a crucial role in both software development and brain-computer interface (BCI) technology by enabling the analysis of large datasets and improving decision-making. In software, statistical methods help identify patterns, optimize algorithms, and ensure reliability through data-driven testing. For BCIs, statistics are essential for analyzing brain signal data, calibrating devices, and improving the accuracy of decoding user intentions. By applying statistical models, these fields can make more informed, evidence-based advancements.

Task: Watch this <u>video</u> to learn the basic concepts of data analysis. Then watch the following videos to understand how python can be used for statistical analysis.
- <u>Lesson 1</u> (Pandas)
- <u>Lesson 2</u> (Matplotlib)
- <u>Lesson 3</u> (Scikit-learn & Statsmodels)