DESCRIPTION OF DAY6

- Collaborated with Rutuja to explore biomimetic joint control using an EMG (Electromyography) muscle sensor.
- Learned how muscle signals can be captured and used to control hand gestures—specifically to open and close a robotic hand.
- Understood the complete hardware setup:
    - EMG sensor connected to the analog input of a Raspberry Pi Pico.
    - Servo motor connected to a PWM pin.
    - Used an external 5V power source for stable servo operation.
- EMG signals were processed and mapped to servo angles using analog-to-digital conversion and PWM control in code.
- Learned how to set threshold levels to detect valid muscle activation and prevent false movement.
- Worked alongside Tejaswini to understand how MicroPython code controls hardware:
    - Explored how `machine.PWM`, `machine.ADC`, and loops are used to read sensor data and move servos.
    - Understood how small changes in code (like angle mapping or delays) directly affect robot behavior.
- Gained confidence in writing and modifying MicroPython code for real-time sensor-based actuation.
- Realized the importance of software-hardware integration in translating biological signals into smooth robotic motion.
- PWM in MicroPython maps joint angles by converting degrees (0–180°) to 16-bit duty cycle values that produce 1–2 ms pulses.
- This pulse width is what the servo reads to rotate to the desired position.
- The mapping allows for precise control of joint angles, essential for humanoid robots and biomimetic motion.