PyWeek

Day 2 — Review & AI-Powered Prototyping

From homework review to building with AI

May 6 • Online Workshop • 2 Hours

Günün Planı

0:00 – 0:10	**Welcome Back**	Quick recap of Day 1 concepts
0:10 – 0:50	**Tapşırığın izahı və yoxlanışı**	Walk through solutions by group
0:50 – 1:00	**Break**	Stretch & recharge
1:00 – 1:40	**AI for Prototyping**	Using AI to build faster
1:40 – 1:55	**Live Demo**	Build a working prototype with AI
1:55 – 2:00	**Homework: Prototype**	Build your own AI-assisted prototype

Quick Recap — Day 1

Python Basics

Variables, types, loops, conditionals, functions

Dictionaries

Key-value pairs for structured data

Working with Files

Reading text & CSV, processing datasets

Mini Challenges

Find max, reverse strings, count vowels, word frequency

Homework Review — Bioinformatics

DNA Sequence Analysis

Nucleotide Counting

Count A, T, G, C in a DNA string using dictionaries

GC Content

Calculate the ratio of G+C to total length

Complement Strand

Generate the complementary DNA strand (A↔T, G↔C)

solution_example.py

```
dna = "ATGCGATCGATCGA"

# Count nucleotides
counts = {}
for base in dna:
    counts[base] = \
        counts.get(base, 0) + 1

# GC content
gc = (counts["G"] + counts["C"])
gc_ratio = gc / len(dna)
print(f"GC: {gc_ratio:.1%}")
```

What challenges did you face? What patterns did you discover?

Homework Review — Neuroscience

Signal Analysis

Load Signal Data

Read CSV of timestamped signal values

Compute Statistics

Mean, max, min firing rates across channels

Detect Peaks

Find signal values above a threshold

signal_analysis.py

```
import csv

signals = []
with open("eeg_data.csv") as f:
    for row in csv.DictReader(f):
        signals.append(
            float(row["voltage"]))

# Basic stats
avg = sum(signals)/len(signals)
peak = max(signals)
print(f"Avg: {avg:.2f} mV")
```

How did you handle noisy data? Any surprises in the signals?

AI-Powered Prototyping

> *Use AI tools (ChatGPT, Claude, Copilot) to go from idea to working code in minutes, not hours.*

Describe Clearly

Tell the AI exactly what you need: input, output, steps

Iterate Fast

Start rough, refine with follow-up prompts

Verify & Learn

Always read, test, and understand the generated code

Writing Good Prompts

❌ Vague Prompt

```
"Write me a Python script
for biology"
```

✅ Specific Prompt

```
"Write a Python function that
reads a FASTA file, counts
nucleotides, and returns GC%"
```

Key Tips for Better AI Outputs

Specify input format	"The input is a CSV with columns: species, count, date"
Add constraints	"Use only standard library, no pip installs needed"
Describe expected output	"Return a dictionary mapping species name to total count"
Ask for explanations	"Add comments explaining each step"

Live Demo — Prototype in Minutes

1

Define the Problem

I need a tool that analyzes a CSV of species observations and outputs a summary report

2

Prompt the AI

Ask for the script step-by-step: read CSV, count species, calculate stats, format output

3

Review & Run

Read through generated code, understand each function, run it on sample data

4

Iterate & Improve

Add features: error handling, visualization, export to new CSV, user input

Homework — Build Your Prototype!

Use AI tools to build a working prototype for your domain. Due: May 10

Bioinformatics

Build a sequence analysis tool
e.g., FASTA parser, mutation finder, or GC content calculator

Neuroscience

Build a signal processing tool
e.g., EEG peak detector, firing rate calculator, or signal plotter

Requirements: Working Python script • Uses concepts from Day 1 • Document your AI prompts • Be ready to present on Day 3