## Problem 1 — Scaling head-to-head

You have `Weight_kg = [58, 62, 65, 66, 190]` and `Height_cm = [150, 160, 170, 175, 180]`.

Tasks:
a) Compute the Standardization for both features by hand on the first three values only.
b) Compute Min–Max Scaling to [0, 1] for all values.
c) Compute Robust Scaling for `Weight_kg` using median and IQR.
d) Which method handles the outlier in `Weight_kg` best and why?

## Problem 2 — When to use which scaler

You are preparing features for a simple model:

- `Monthly_Income` in BDT with a few extremely high earners
- `Room_Temperature_C` within 18 to 30
- `Transaction_Count_Last_7_Days` with many zeros and a few heavy shoppers

Tasks:
a) For each feature, pick one scaler: Standardization, Min–Max, or Robust. Justify each choice in one sentence.
b) Give a quick numeric example showing why your chosen scaler is safer or more interpretable for that feature.

## Problem 3 — Nominal vs Ordinal and encoding

Dataset:

| ID | City | Education_Level | Satisfaction |
|---|---|---|---|
| 1 | Dhaka | High School | Low |
| 2 | Chattogram | Bachelor | Medium |
| 3 | Rajshahi | Master | High |
| 4 | Dhaka | Bachelor | Medium |

Tasks:
a) Mark which are nominal and which are ordinal.
b) One-hot encode `City` by hand for the 4 rows. Use 0 and 1.
c) Ordinal-encode `Education_Level` with High School=0, Bachelor=1, Master=2.

## Problem 4 — Vectors, dot product, norms

Given vectors **a** = [3, −1, 2] and **b** = [4, 0, −2].

Tasks:
a) Compute the dot product a·b by hand.
b) Compute L1 and L2 norms of **a**.
c) Write a tiny `numpy` snippet to verify.

## Problem 5 — Euclidean vs Manhattan distance in practice

Points: P1(2, 3), P2(5, 7), P3(2, 10).

Tasks:
a) Compute Euclidean and Manhattan distances for all pairs by hand.
b) Which metric changes more when a single coordinate has a large jump, and why?
c) Scale the y-coordinate by a factor of 10 and recompute distances between P1 and P2. What did this do to your intuition?