🧩 What is Column Transformer in ML?

In real-world datasets, we often deal with **different types of data**:

- **Numerical data**: like age, salary, height
- **Categorical data**: like gender, city, product type

Each type of data needs **different preprocessing** steps. For example:

- Numerical data may need **scaling** (e.g., Standardization or Normalization).
- Categorical data may need **encoding** (e.g., One-Hot Encoding or Label Encoding).

So how do we apply these different preprocessing steps **at once**, without writing separate code for each column?

That's where Column Transformer comes in!

It helps us:

- Apply **scaling** to only numeric columns.
- Apply **encoding** to only categorical columns.
- Leave some columns **untouched**, if needed.

Think of it like a **preprocessing machine** where you send in different columns through different "pipelines" and get one clean, ready-to-use dataset.

Why is it Useful?

- Keeps code **organized and clean**
- Reduces **manual work**
- Ensures each column is **transformed appropriately**
- Works seamlessly with **machine learning models**

Real-Life Analogy:

Imagine you're washing different types of laundry:

- Shirts go into a **normal wash**
- Jackets go into a **delicate wash**
- Jeans go into a **heavy wash**

The **Column Transformer** is like a smart washing machine that knows which items need which wash and processes all at once in separate compartments.
