Week Twelve

Siva Sundar, EE23B151

December 16, 2024

16^{nd} December

- A **database** is an organised system of interlocking tables. **Database Scheme** are categories $\mathcal{C}$, data itself is given by a *'set-valued'* functor $\mathcal{C} \longrightarrow$**Set**, and databases can be mapped to each other via functors $\mathcal{C} \longrightarrow \mathcal{D}$.
- Category theory formalizes **data migration** between databases using **adjoint functors**.
- A category $\mathcal{C}$ consists of four constituents:
  - ⋆ a collection $\mathrm{Ob}(\mathcal{C})$, whose elements are **objects**.
  - ⋆ $\forall c, d \in \mathrm{Ob}(\mathcal{C})$, we specify the **hom-set** $\mathcal{C}(c, d)$, whose elements are called **morphisms**.
  - ⋆ $\forall c \in \mathrm{Ob}(\mathcal{C})$, we can specify the **identity morphism** on c: $id_c \in \mathcal{C}(c, c)$.
  - ⋆ $\forall c, d, e \in \mathrm{Ob}(\mathcal{C})$ and morphisms $f \in \mathcal{C}(c, d)$ and $g \in \mathcal{C}(d, e)$, we can define the **composite morphism** $f \circ g \in \mathcal{C}(c, c)$.

  It should also satisfy two properties:

  - ⋆ *Unitality*: $id_c \circ f = f \circ id_c = f$.
  - ⋆ *associativity*: $(f \circ g) \circ h = f \circ (g \circ h)$.