DATABASE DESIGN PROCESS

REQUIREMENTS ANALYSIS

↓

CONCEPTUAL DESIGN

↓

LOGICAL DESIGN

↓

PHYSICAL DESIGN

↓

IMPLEMENTATION

↓

MAINTENANCE & EVOLUTION

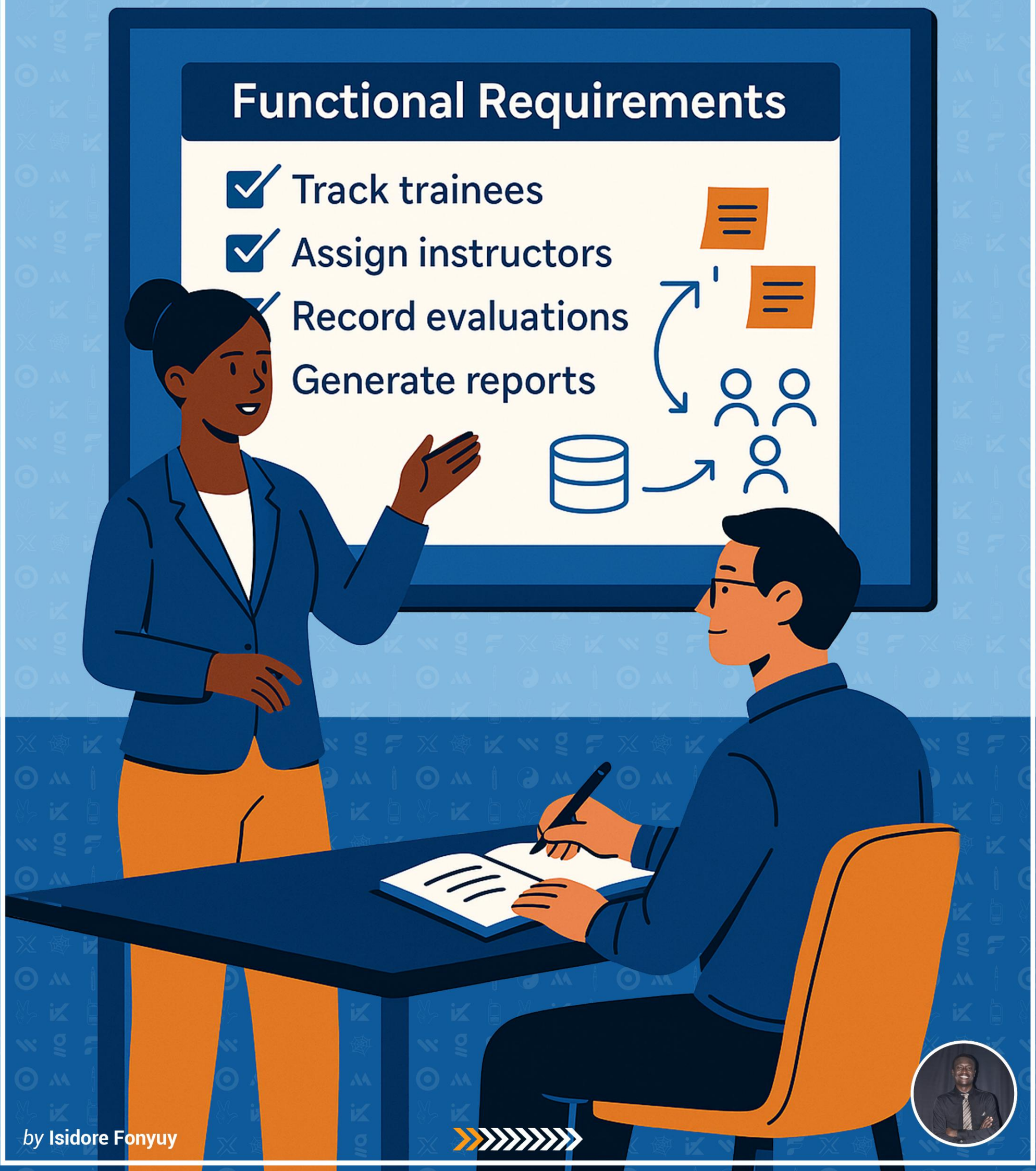
↓

TESTING

by **Isidore Fonyuy**

Database Design

REQUIREMENTS



by **Isidore Fonyuy**

Database Design

REQUIREMENTS GATHERING / ANALYSIS

TO DO

- Understand the purpose of the database *(i.e. problems to solve, end users, data to store)*.

- Gather requirements through interviews, surveys, analysis of the existing system.

- Identify key entities, processes and business rules.

SAMPLE (REQUIREMENTS GATHERED)

Our school is 100% virtual and we have students in different parts of the world:
i.e. Africa, Europe, Asia, and the Americas.

We keep records of the names, gender, date of birth, address, race,
start date, end date and status of both our staff and trainees.

We also track the class a student belongs to, the program
they enrolled for and who their mentor *(designated teacher)* is.

To make our trainings affordable for all we split our students into sections:
- first section = students based in North American countries,
- second section = students based in African countries,
- third section = students based in the rest of the world.

We offer certification courses in Database Administration, Data Analysis & Programming.
A student can take only one of these courses at a time. We also provide diplomas.

The combination of program and section and session *(enrollment date - completion date)*
indicates what class a student can register into.

A student's performance and final result is determined by:
- continuous assessment *(i.e. quizes, active participation in class)*,
- a midterm test,
- a final exam, and
- a capstone project.

by **Isidore Fonyuy**

Database Design

FUNCTIONAL ANALYSIS

ANALYSING REQUIREMENTS

- Identify entities
- Identify attributes of these entities.
- Identify relationships between these entities.

ENTITIES & ATTRIBUTES

Trainee (*name*, *gender*, *dob*, *race*, *address*, *enroll*, *complete*, *status*, *class*, *program*, *mentor*)

Instructor (*name*, *gender*, *dob*, *hiredate*, *status*, *race*, *address*)

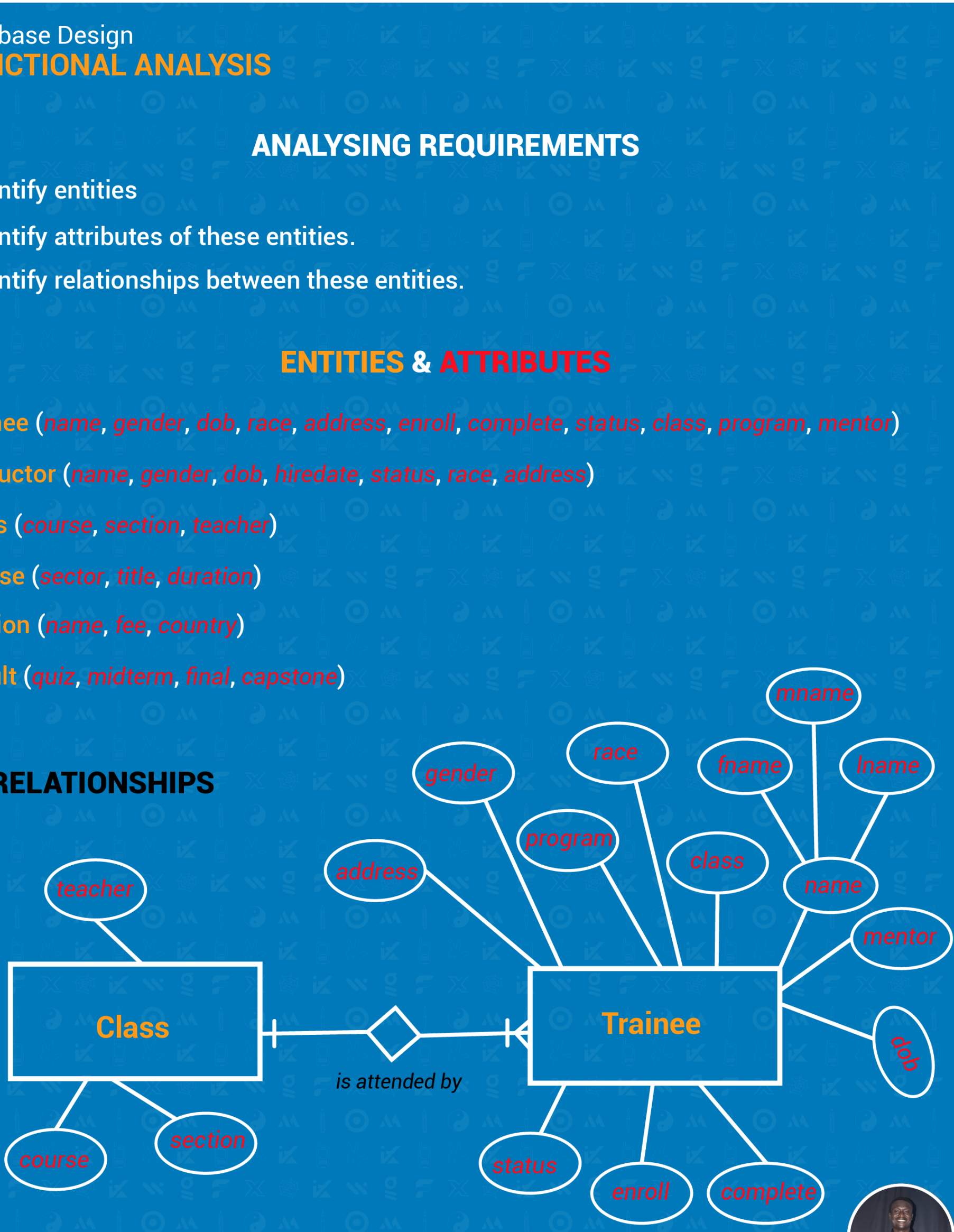
Class (*course*, *section*, *teacher*)

Course (*sector*, *title*, *duration*)

Section (*name*, *fee*, *country*)

Result (*quiz*, *midterm*, *final*, *capstone*)

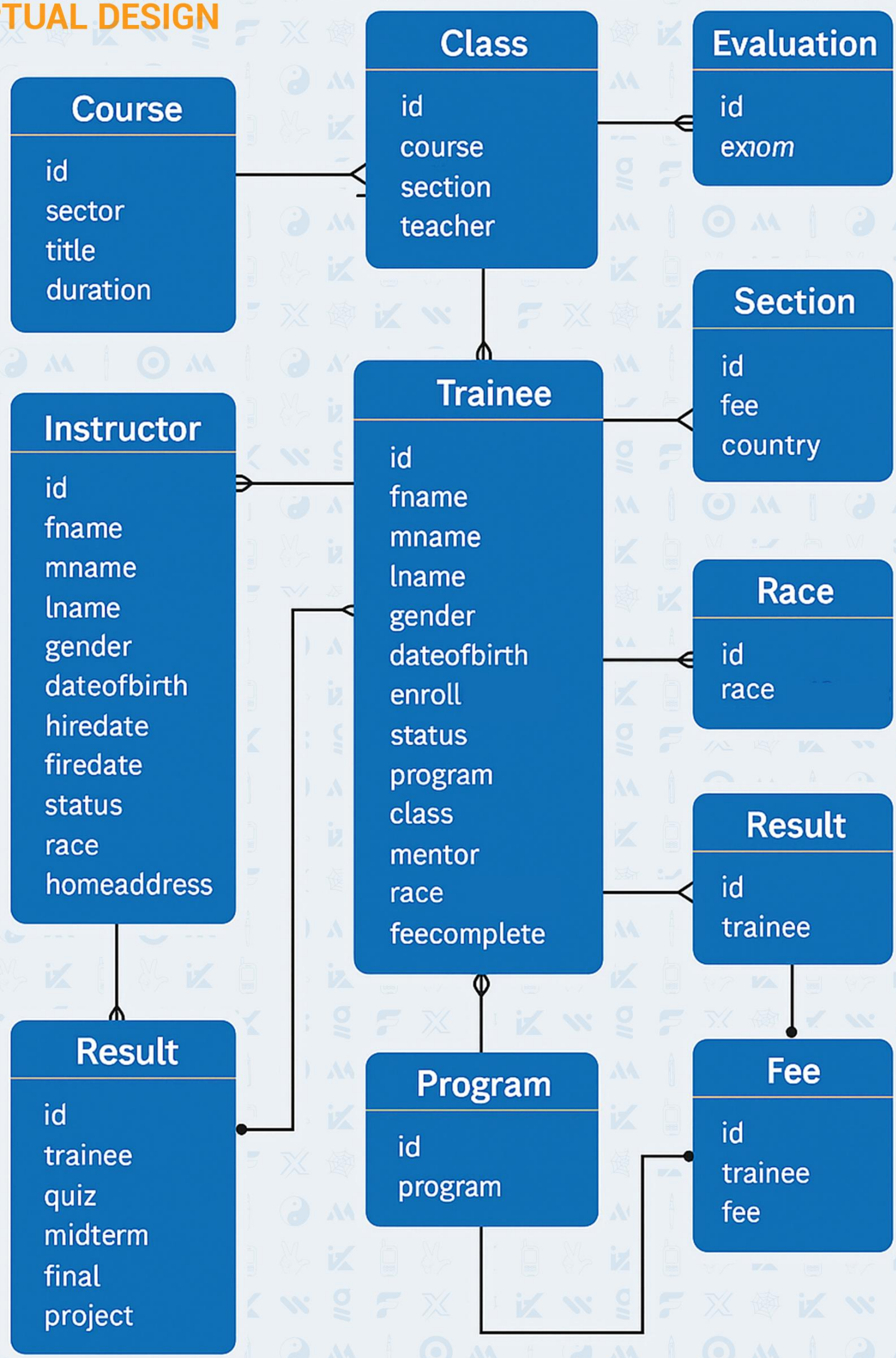
RELATIONSHIPS



by **Isidore Fonyuy**

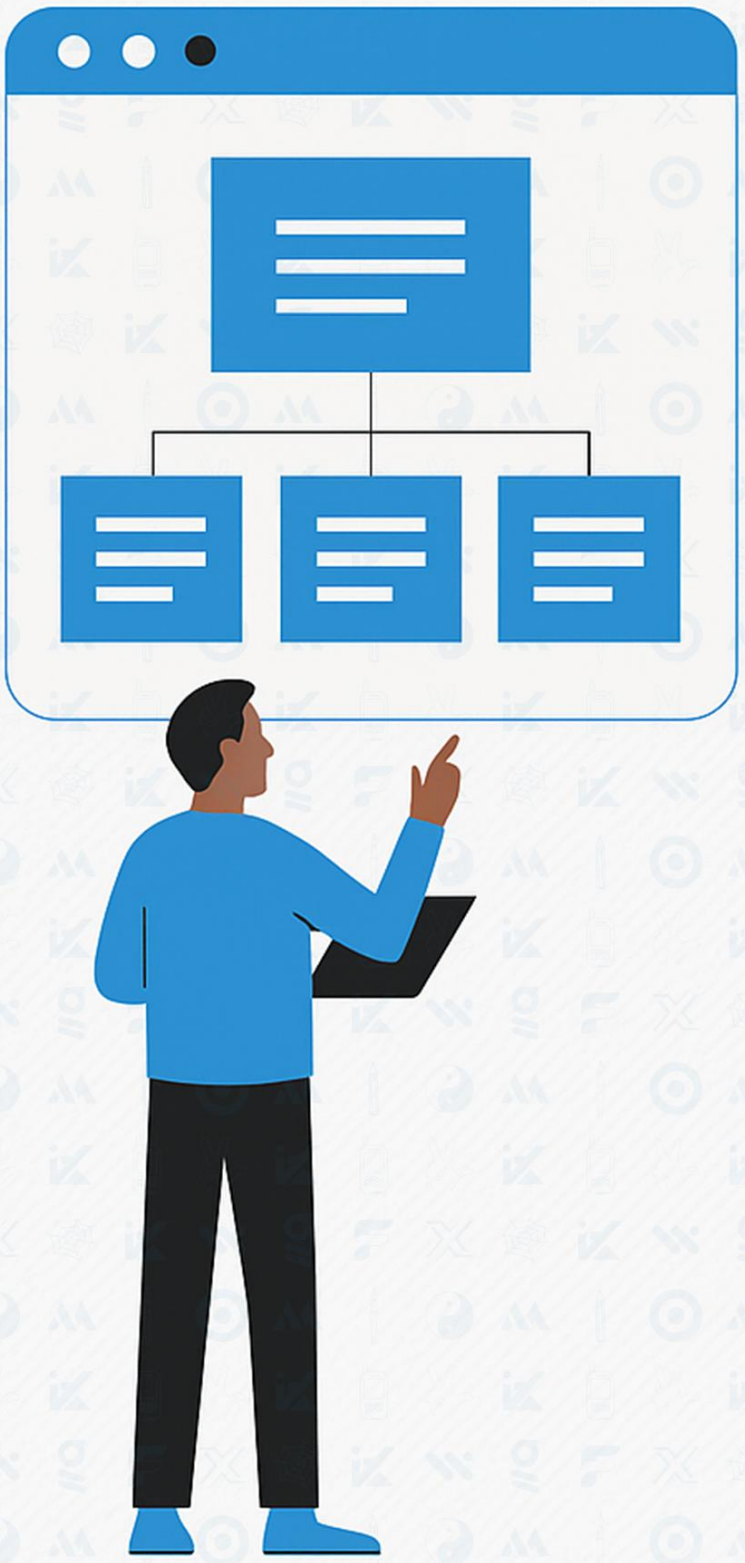
Database Design

CONCEPTUAL DESIGN

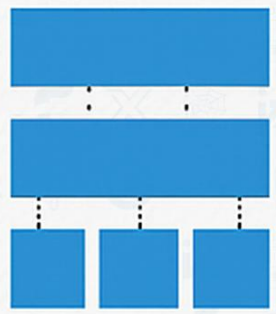


by **Isidore Fonyuy**

LOGICAL DESIGN



Data Models



Normalization

Constraints

by **Isidore Fonyuy**

Database Design

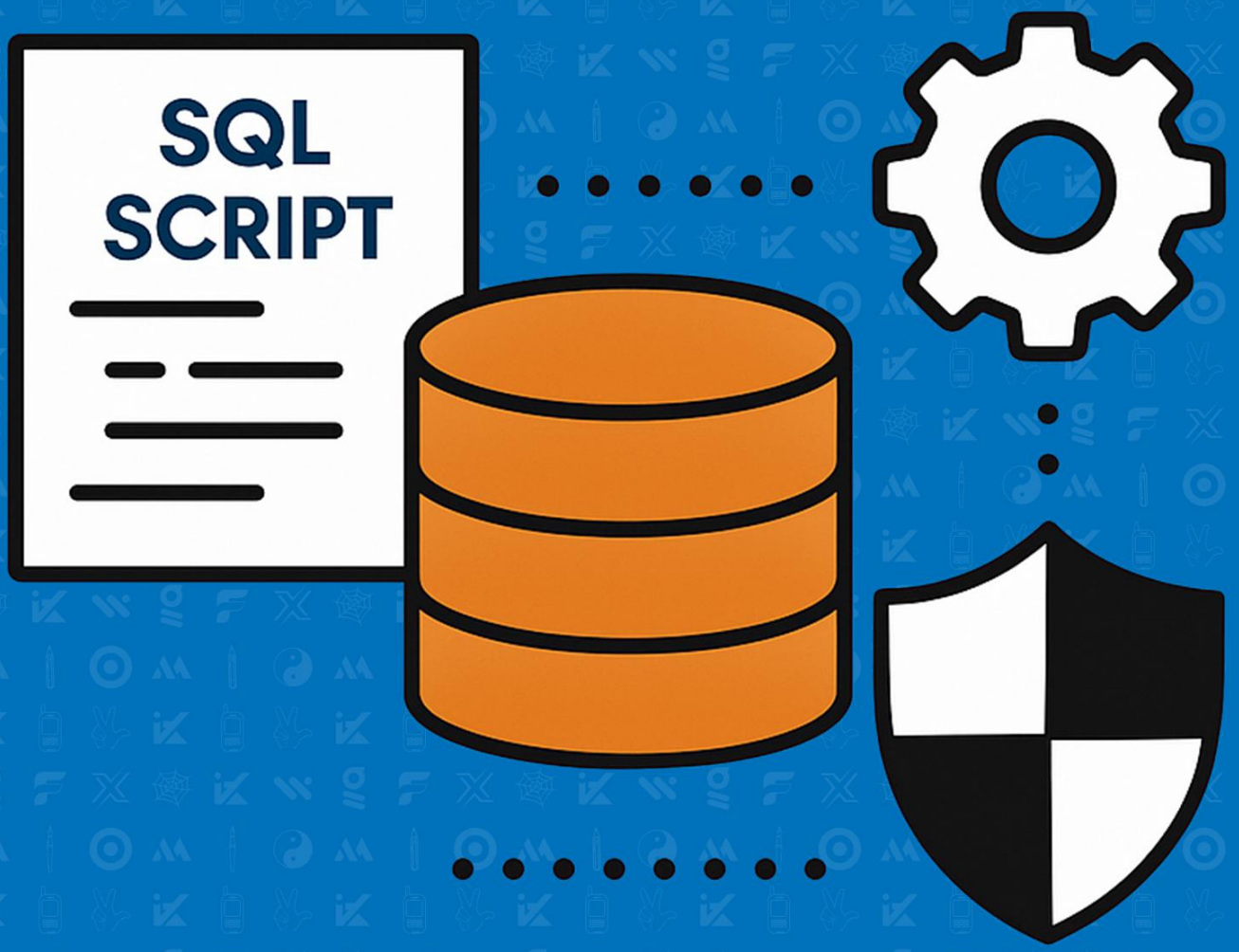
PHYSICAL DESIGN

PHYSICAL DESIGN

by **Isidore Fonyuy**

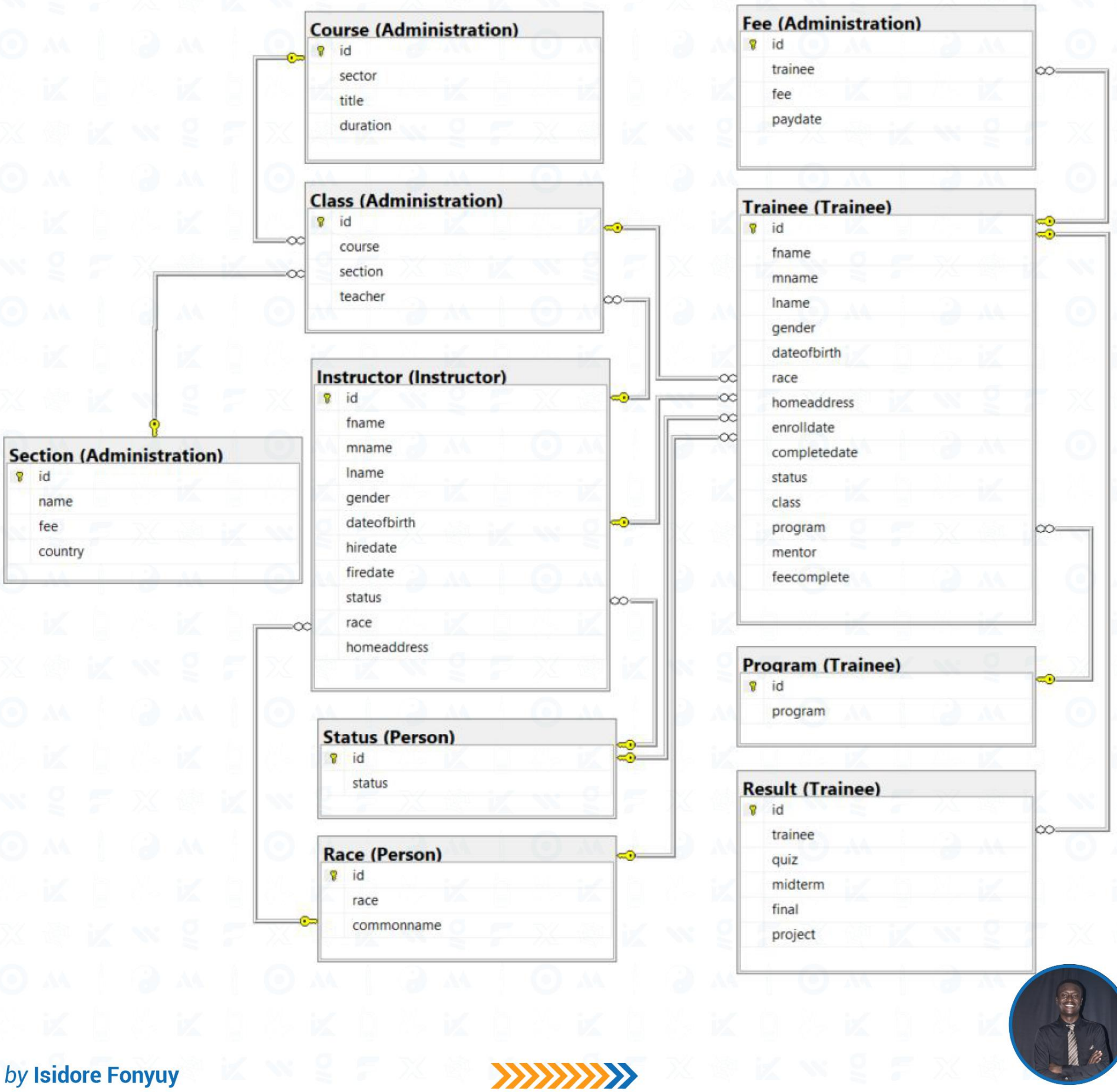
Database Design
IMPLEMENTATION

IMPLEMENTATION



by **Isidore Fonyuy**

DATABASE DIAGRAM



by **Isidore Fonyuy**

Database Design

TESTING & MAINTENANCE

TESTING & MAINTENANCE

by **Isidore Fonyuy**

Database Design

DATABASE DESIGN TOOLS

TOOLS

- Jira
- Trello
- Lucidchart
- Draw.io
- Visual Paradigm
- MySQL Workbench
- dbdiagram.io
- DBDesign
- Microsoft Word
- Google Sheets
- IBM InfoSphere Data Architect
- MySQL
- SQL Server
- Oracle
- PostgreSQL
- DBDesign
- phpMyAdmin
- DBeaver
- SSMS
- IntelliJ IDEA
- pgAdmin
- Tableau
- Metabase
- dbForge Studio
- Redgate
- Flyway
- Azure Data Studio
- ER/Studio
- QuiokDBD
- Altova DatabaseSpy
- Navicat
- Datadog

by **Isidore Fonyuy**

TOOLS

NB: *You'll never need ALL of these tools. However, from among them you might find tools that best suit your style or project.*

🔍 1. Requirements Analysis

Goal: *Understand what the system needs to do and what data it will manage.*

Tools:

- 📝 **Microsoft Word / Google Docs** – Documentation of requirements.
- 📋 **Notion / Confluence** – Collaborative requirement gathering.
- 🧠 **Miro / Lucidchart** – Brainstorming & mind mapping.
- 🗣️ **Interview & Survey Tools** – Google Forms, Typeform, or simple spreadsheets for collecting feedback.

🧱 2. Conceptual Design

Goal: *Build a high-level model of the data, usually with Entity-Relationship Diagrams (ERDs).*

Tools:

- 🧩 **Lucidchart / Draw.io** – ERD creation.
- 🗂️ **dbdiagram.io / QuickDBD** – Simple and fast ERD creation via text input.
- 🧠 **ER/Studio / ERwin Data Modeler** – Professional data modeling software.

🗃️ 3. Logical Design

Goal: *Convert the conceptual model into a logical structure, typically relational (normalized tables, keys, constraints).*

Tools:

- 🧠 **SQL Power Architect** – Logical modeling and schema mapping.
- 🧱 **MySQL Workbench / pgModeler** – Logical model support with auto-generation of schemas.
- 📄 **Spreadsheet Tools (Excel/Google Sheets)** – Initial normalization and data mapping.

🛠️ 4. Physical Design

Goal: *Translate the logical design into actual database structures optimized for performance.*

Tools:

- 🛢️ **Database Management Systems (DBMS):**
  - 🐘 PostgreSQL
  - 🐬 MySQL
  - 📦 SQL Server
  - ☁️ Amazon RDS / Google Cloud SQL
- 🛠️ **DB Tools:**
  - **DBeaver / HeidiSQL / SQL Developer** – Schema creation and DB interaction.
  - **ERwin / Aqua Data Studio** – Advanced physical design and tuning.

🧪 5. Implementation & Testing

Goal: *Create the database and test it with real data and queries.*

Tools:

- 📦 **SQL scripts / migration tools** – Flyway, Liquibase for schema/version management.
- 🧪 **Postman / Insomnia** – API-level testing.
- 🗜️ **JMeter / k6** – Load testing for performance.
- 🧱 **Unit Test Frameworks** – tSQLt (SQL Server), pgTAP (PostgreSQL).

🔄 6. Maintenance & Monitoring

Goal: *Monitor database performance, tune queries, and apply updates.*

Tools:

- 📈 **New Relic / Datadog / Prometheus + Grafana** – Monitoring and alerts.
- ⚙️ **pgAdmin / SQL Server Management Studio (SSMS)** – Ongoing management.
- 🚦 **Slow Query Logs / EXPLAIN / Query Analyzer** – Performance tuning.
- 🔐 **Backup Tools** – pgBackRest, mysqldump, native cloud tools.

PROJECT RESOURCES

I have uploaded all SQL scripts to GitHub. You can find them here:

https://www.github.com/isidoredelpierro/

I wrote the code using SQL Server. If you are using a different DBMS you will have to update the code to match the SQL syntax of your environment. Also, parts of the code are pointing to specific file PATHs on my disks. Consider creating the same disk structure, or updating the code to reflect your own environment.