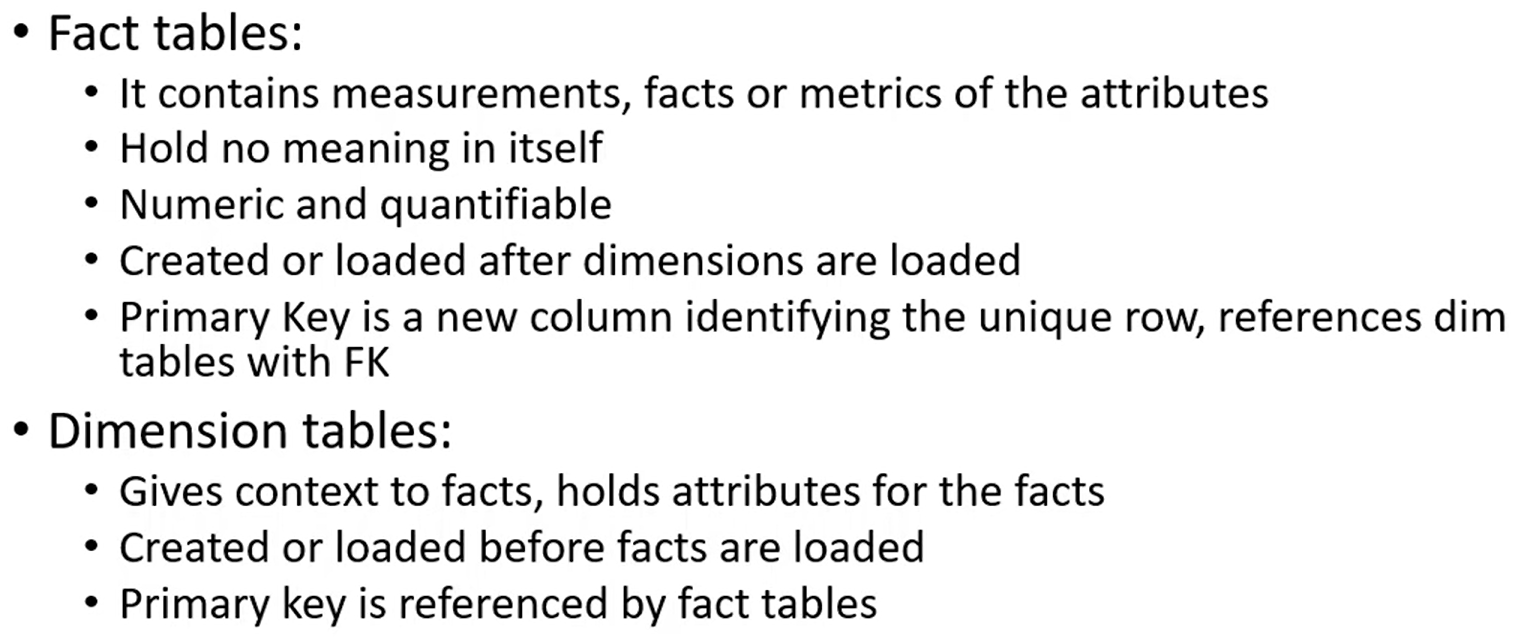
**Day 2: Database Schema & ERD**

**Goal:**

The goal of the database schema is to structure marketing campaign data in a relational format suitable for both analytics and machine learning.  
I used MySQL for implementation and dbdiagram.io for ERD visualization.

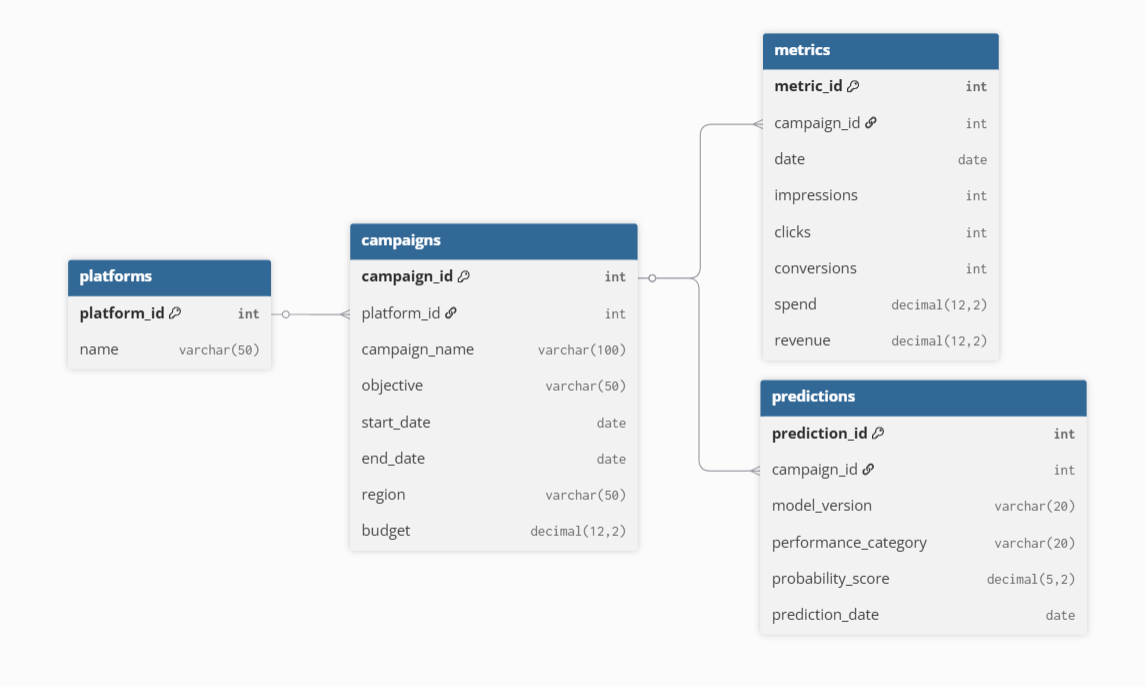
**Fact tables** and **dimension tables** are the two fundamental building blocks of a dimensional data model, which is a design used for storing data in a data warehouse for easy and fast analysis.

****

I’ll create **four main tables**.

| **Table** | **Description** | **Type** |
| --- | --- | --- |
| **platforms** | Stores ad platform details | Dimension |
| **campaigns** | Stores campaign info | Dimension |
| **metrics** | Daily performance data | Fact |
| **predictions** | Stores ML model output for each campaign | Fact |

**Entity Relationship Diagram:**



* One platform has many campaigns
* Each campaign has many daily metrics
* Each campaign can have multiple prediction results

**Referential Integrity:**

It is a rule in relational databases that ensures relationships between tables remain valid and consistent.

In simple terms, it means if a record in one table refers to a record in another table, the referenced record must actually exist.

It is enforced by using Foreign Keys that link to Primary Keys.

**Why I normalized the tables?**

I normalized the tables to remove data redundancy and maintain data consistency.  
For example, instead of repeating platform names or campaign details in every metrics record, I separated them into individual tables (platforms, campaigns, metrics).  
This approach reduces storage duplication, avoids update anomalies, and makes data easier to maintain.  
It also improves query efficiency because each table focuses on one type of entity making joins simpler and cleaner.

**Why does metrics have a foreign key to campaigns?**

The metrics table has a foreign key (campaign\_id) linking it to the campaigns table because each set of daily performance metrics belongs to a specific campaign.  
This relationship ensures that all metric entries are tied to a valid campaign, meaning no metric exists without a parent campaign.  
It enforces referential integrity, helps when joining data, and allows easy tracking of how each campaign performs over time.

**If I add new platforms, can my schema handle it easily?**

Yes, my schema is flexible because of the platforms table.  
Each platform has its own unique ID and name, and campaigns reference it through a foreign key (platform\_id).  
So, if I add a new platform like LinkedIn Ads or TikTok Ads , we only need to insert one new record into the platforms table.  
The rest of the schema remains unchanged, and new campaigns can start linking to that platform immediately.

**Key Learnings:**

* Learned relational modelling and schema design
* Created 4-table database structure
* Implemented constraints for data integrity
* Visualized schema using dbdiagram.io
* Prepared clear interview-ready explanations

**Resources:**

SQ - <https://mode.com/sql-tutorial/introduction-to-sql>

ER Diagram - <https://dbdiagram.io/home>

SQL pronunciation - <https://patorjk.com/blog/2012/01/26/pronouncing-sql-s-q-l-or-sequel/>

Fact and dimensions table - <https://youtu.be/dKyA6oBlKwo>

Referential Integrity - <https://youtu.be/5y28osxcDvk?si=2vbIT2vHFMgmO7oJ>

MySQL Workbench - <https://youtu.be/ckz21e6rItw>