🗄️ MySQL Naming Conventions Guide

A standardized approach to database design for clarity, compatibility, and scalability.

1. General Rules

Follow these foundational rules for every database object you create:

- **Lowercase Only:** Always use lowercase. MySQL can be case-sensitive depending on the OS.
- **Snake Case:** Use snake_case (e.g., user_profiles, not userProfiles).
- **Clean Characters:** Use only letters and underscores. No spaces or hyphens.
- **No Reserved Keywords:** Avoid words like order, group, user, select, or rank.
- **Be Descriptive:** Names should be unambiguous. Short is good, but meaningful is better.

2. Table Naming

Tables represent collections of entities.

Rule	**Description**
Plural Nouns	Use users instead of user.
Entity Focus	One table should represent one specific business entity.
No Prefixes	Avoid tbl_ or db_. They add unnecessary noise.

✅ **Examples:** orders_history, customer_addresses, payment_transactions, ai_agents.

❌ **Avoid:** User (singular/caps), Order (reserved), user-data (hyphen).

3. Column (Field) Naming

Columns represent the attributes of an entity.

Feature	Convention	Example
Singular Nouns	Use singular names for attributes.	email_address, phone_number
Primary Keys	Simply use id.	id
Foreign Keys	referenced_table_singular + _id.	user_id, agent_id
Booleans	Use prefixes: is_, has_, or can_.	is_active, has_access

4. Date & Time Suffixes

Consistency here makes filtering and sorting much more intuitive.

- **Timestamps (_at):** Used for specific points in time.
  - created_at, updated_at, deleted_at, last_login_at
- **Dates (_date):** Used when the time component isn't required.
  - start_date, end_date, birth_date

5. Indexes & Constraints

Standardizing these names makes debugging migration errors much easier.

- **Indexes:** idx_<table>_<column> (e.g., idx_users_email)
- **Unique Constraints:** uq_<table>_<column> (e.g., uq_users_username)
- **Foreign Keys:** fk_<table>_<referenced_table> (e.g., fk_orders_users)

6. Reserved Keyword Workarounds

If you feel the urge to use a reserved word, use these industry-standard alternatives:

❌ Reserved Word	✅ Safe Alternative

user	app_user / account
order	purchase_order
group	user_group / category
rank	priority_level / position
date	event_date / entry_date

7. Implementation Example

```
CREATE TABLE training_programs (
    id BIGINT AUTO_INCREMENT PRIMARY KEY,
    program_name VARCHAR(150) NOT NULL,
    program_description TEXT,
    training_status VARCHAR(30) NOT NULL,
    start_date DATE,
    end_date DATE,
    is_active TINYINT(1) DEFAULT 1,
    created_at DATETIME NOT NULL,
    updated_at DATETIME NOT NULL
);
```

✅ Final Checklist

- [ ] Is it **lowercase snake_case**?
- [ ] Is the table name **plural**?
- [ ] Did I avoid **reserved keywords**?
- [ ] Are foreign keys named **singular_table_id**?
- [ ] Are boolean fields prefixed with **is_/has_/can_**?