📝 Database Normalization Cheat Sheet

🔹 1NF (First Normal Form)

✅ **Rules**:

- All values must be **atomic** (indivisible).
- No repeating groups/arrays in columns.

🚨 **Example Fix**:

```sql
-- Before 1NF (Violation)
Orders: (OrderID, CustomerID, [Pizza, Burger])

-- After 1NF
Orders: (OrderID, CustomerID, Product)
```

🔹 2NF (Second Normal Form)

✅ **Rules**:

- Must be in 1NF.
- No **partial dependency** (non-key columns depend on the *entire* primary key).

🚨 **Example Fix**:

```sql
-- Before 2NF (Partial Dependency)
Orders: (OrderID, ProductID, ProductName)

-- After 2NF
Orders: (OrderID, ProductID)
Products: (ProductID, ProductName)
```

🔹 **3NF (Third Normal Form)**

✅ **Rules**:

- Must be in 2NF.
- No **transitive dependency** (non-key columns depend on other non-key columns).

🚨 **Example Fix**:

```sql
-- Before 3NF (Transitive Dependency)
Customers: (CustomerID, City, State)

-- After 3NF
Customers: (CustomerID, City)
Cities: (City, State)
```

🔹 **BCNF (Boyce-Codd Normal Form)**

✅ **Rules**:

- Stricter 3NF.
- Every **determinant** must be a **candidate key**.

📌 *Learn more: [BCNF Deep Dive]*

🔹 **4NF & 5NF (Advanced Forms)**

- **4NF**: No multi-valued dependencies.
- **5NF**: No join dependencies (rarely used).

🚦 When to Normalize vs. Denormalize?

Normalize	**Denormalize**
OLTP systems (e.g., banking)	Data warehouses (analytics)
High-write systems	High-read systems
Prioritize data integrity	Prioritize query speed

⚡ Pro Tips

1. **Test Queries**: Normalization can slow down reads—always benchmark!
2. **Hybrid Approach**: Mix normalized + denormalized tables (e.g., cache frequent joins).
3. **Tools**: Use EXPLAIN ANALYZE in SQL to spot join bottlenecks.