Example

### **What is Normalization?**

**Normalization** is the process of **removing unwanted redundancy** from database tables by organizing data into smaller, well-structured tables using **functional dependencies (FDs)**.

### **What is a Normal Form?**

A **normal form** is a **rule** or **standard** for how data and dependencies should be structured in a table. Each higher normal form removes **more types of redundancy**.

* **1NF:** Eliminate repeating groups (atomic values)
* **2NF:** Remove partial dependency
* **3NF:** Remove transitive dependency
* **BCNF:** Stronger version of 3NF

### Example

### **UNNORMALIZED TABLE (UNF)**

We have the following table:

| **StudentID** | **StudentName** | **Courses** | **Instructor** |
| --- | --- | --- | --- |
| 101 | Alice | {CS101, MA101} | {Dr. Smith, Dr. Lee} |
| 102 | Bob | {CS101} | {Dr. Smith} |
| 103 | Charlie | {MA101, PH101} | {Dr. Lee, Dr. Ray} |

**Issue:** Multi-valued attributes (Courses, Instructor)

## **1NF (First Normal Form)**

**Remove multi-valued attributes**

| **StudentID** | **StudentName** | **CourseCode** | **Instructor** |
| --- | --- | --- | --- |
| 101 | Alice | CS101 | Dr. Smith |
| 101 | Alice | MA101 | Dr. Lee |
| 102 | Bob | CS101 | Dr. Smith |
| 103 | Charlie | MA101 | Dr. Lee |
| 103 | Charlie | PH101 | Dr. Ray |

**Now each cell has atomic values.**

## **2NF (Second Normal Form)**

It should be in 1NF

**Remove partial dependency**

* Composite primary key: (StudentID, CourseCode)
* StudentName depends only on StudentID (partial dependency)

### **Split into two tables:**

#### **Student Table**

| **StudentID** | **StudentName** |
| --- | --- |
| 101 | Alice |
| 102 | Bob |
| 103 | Charlie |

#### **Enrollment Table**

| **StudentID** | **CourseCode** | **Instructor** |
| --- | --- | --- |
| 101 | CS101 | Dr. Smith |
| 101 | MA101 | Dr. Lee |
| 102 | CS101 | Dr. Smith |
| 103 | MA101 | Dr. Lee |
| 103 | PH101 | Dr. Ray |

**No partial dependency remains.**

## **3NF (Third Normal Form)**

It should be in 2NF

**Remove transitive dependency**

* Instructor depends on CourseCode, not on (StudentID, CourseCode)

**Split further:**

#### **Course Table**

| **CourseCode** | **Instructor** |
| --- | --- |
| CS101 | Dr. Smith |
| MA101 | Dr. Lee |
| PH101 | Dr. Ray |

#### **Enrollment Table (Final)**

| **StudentID** | **CourseCode** |
| --- | --- |
| 101 | CS101 |
| 101 | MA101 |
| 102 | CS101 |
| 103 | MA101 |

**No transitive dependency remains.**

## **BCNF (Boyce-Codd Normal Form)**

It should be in 3NF

**Ensure every determinant is a candidate key**

* In Course table: CourseCode → Instructor (CourseCode is a key)
* In Enrollment: (StudentID, CourseCode) is the only determinant

**Already in BCNF. No more decomposition needed.**

### **Final Tables (BCNF)**

1. **Student(StudentID, StudentName)**
2. **Course(CourseCode, Instructor)**
3. **Enrollment(StudentID, CourseCode)**