

Normalizing Database using functional dependency upto BCNF

Step 1: Define the initial relational schema and fundamental dependencies.

Student Slot (Student ID, Student-name, Email, Dept-ID, Dept-name, course-ID, course-name, credits, slot-ID, slot-name, Data, venue).

1. Student-ID \rightarrow student-name, E-mail, Dept-ID
2. Dept-ID \rightarrow Dept-name
3. Course-ID \rightarrow course-name, credits, course-type
4. Slot-ID \rightarrow slot-type, Data, venue
5. Student-ID \rightarrow Slot-ID \rightarrow course-ID.

Step 2: Convert the relational to 1NF

* Identity and eliminate any repeating groups or in the Student Slot relation.

* Create separate tables of separating

Step 3: convert to 2NF

* Ensure that each non-key attribute depends on the whole primary key.

* Move non-key Attribute to separate relations if they depends only part of primary key.

proposed decomposition:-

1. Student (Student-ID, student-name, Email, Dept-ID)
2. Department (Dept-ID, Dept-name)
3. course (course-ID, course-name, credits)
4. Slot (slot-ID, slot-type, Data, venue)
5. Student-Slot-course (student-ID, slot-ID, course-ID)

Step-4: Convert to 3NF

- * Remove transitive dependencies where a non-key attribute depends on another, non-key attributes.
- * There is no transitive dependencies.

Step 5: Convert to BCNF

- * Ensure every determinant is a candidate key
- * check for overcapping candidate key
- * Decompose relation to eliminate dependency
- * No decomposition needed

Use Ernst tool:-

1. Input relational Schema and functional dependencies
2. Ernst tool generates a dependency graph
3. Analyze normalization suits to transform the Schema.
4. Apply normalization suits to transform the Schema.
5. Verify the resulting Schema meets BCNF criteria.

Ernst tool Steps:-

1. Create a new project in Ernst
2. Define the relational schema and FD's
3. Run the dependency graph tool.
4. Analyze the graph for normalization issue
5. Apply transformations using the "Normalization" tool
6. Verify BCNF compliance using the "BCNF" check tool

Normalized Schema:

1. student (Student-ID, student-name, Email)
2. Department (Dept-ID, Dept-name)
3. course (course-ID, course-name, credits)
4. slot (Slot-ID, slot-type, data, venue)
5. student_slot_course (student-ID, slot-ID, course-ID).

VELTECH	
EX No.	8
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	4
RECORD (5)	14
TOTAL (20)	
SIGN WITH DATE	

Result:- The Implementation of Normalization database using functional dependency BENFD verified successfully