

Normalisation Task

The goal of this activity is to normalize an un-normalized data table to the Third Normal Form (3NF).

The starting table is as follows:

Student Number	Student Name	Exam Score	Support	Date of Birth	Course Name	Exam Boards	Teacher Name
1001	Bob Baker	78	No	25/08/01	Computer Science	BCS	Mr Jones
					Maths	EdExcel	Ms Parker
					Physics	OCR	Mr Peters
1002	Sally Davies	55	Yes	02/10/99	Maths	AQA	Ms Parker
					Biology	WJEC	Mrs Patel
					Music	AQA	Ms Daniels
1003	Mark Hanmill	90	No	05/06/95	Computer Science	BCS	Mr Jones
					Maths	EdExcel	Ms Parker
					Physics	OCR	Mr Peters
1004	Anas Ali	70	No	03/08/80	Maths	AQA	Ms Parker
					Physics	OCR	Mr Peters
					Biology	WJEC	Mrs Patel
1005	Cheuk Yin	45	Yes	01/05/02	Computer Science	BCS	Mr Jones
					Maths	EdExcel	Ms Parker
					Music	AQA	Ms Daniels

As can be seen, the initial table contains information on students, courses, teachers, and exam boards altogether. In addition, some columns could include more than one value, and it was not clear which was the primary key because there was no column with a unique code.

All the columns depend on attributes that are not the key, and some do not depend on the student.

To normalize the table, I started creating different tables that define the different entities.

Student					Course		Exam Board		Course Teacher	
Student number	Student Name	Date of Birth	Exam Score	Support	Course Name	Course Number	Exam Board	ExamBoard Number	Course Teacher	Course Teacher Number
1001	Bob Baker	25/08/01	78	No	Computer Science	CS01	EDExcel	ED01	Mr. Jones	JNS01
1002	Sally Davies	02/10/99	55	Yes	Maths	M02	AQA	AQ02	Mr. Parker	PKR02
1003	Mark Hanmill	05/06/95	90	No	Physics	P03	OCR	OC03	Mr. Peters	PTR03
1004	Anas Ali	03/08/80	70	No	Music	MC04	WJEC	WJ04	Ms. Daniels	DNL04
1005	Cheuk Yin	01/05/02	45	Yes	Biology	B05	BCS	BC05	Mr. Patel	PTL05

All tables correspond to an entity, and each entity has a unique code that will be used in the link tables.

Then, I created link tables that represent the relationship between the entities. In this case, I created the tables “Student-Course,” “Course-Exam Board,” and “Course-Teacher.”

As it is possible to see in the following image:

Student Course		Course Exam Board		Course Teacher	
Student Number	Course Name	Course Name	Exam Board	Course Name	Course Teacher
1001	CS01	CS01	BC05	CS01	JNS01
1001	M02	M02	AQ02	M02	PKR02
1001	P03	M02	AQ02	M02	PKR02
1002	M02	M02	ED01	P03	PTR03
1002	B05	P03	OC03	MC04	DNL04
1002	MC04	MC04	AQ02	B05	PTL05
1003	CS01	B05	WJ04		
1003	M02				
1003	P03				
1004	M02				
1004	P03				
1004	B05				
1005	CS01				
1005	M02				
1005	MC04				

The “student-course” link table is used to link the students to the courses. A student can follow more than one course, and each course can be followed by more than one student. The student number and course number are foreign keys, while the primary key is composed of the student number + course number. The table “Course-Exam Board” links each course to an exam board; a course can be provided by more than one exam board, and an exam board can manage more than one course. The course number and exam board number are foreign keys, and the primary key is composed of the coursenummer + exam board number.

The Course-Teacher table links the course to its teacher. In the former table, each teacher follows only one course; therefore, the relationship is that each course is linked to one teacher. The course number and teacher number are foreign keys, while the primary key is composed of the course number + teacher number.

To normalize the table, it is necessary to satisfy three normal forms (1NF, 2NF, and 3NF). To achieve 1NF, each column must contain only one value, and each row must include only one record. In the initial table, this aspect was not respected because some columns had more than one piece of information, and the student, course, teacher, and exam board data were mixed. The second normal form was not respected because, in the former table, there were attributes such as teacher and exam board that do not depend on the student but on the course. To solve this problem, the table was divided into more distinct entities so that each attribute depended completely on the primary key of the entity. In addition, by dividing all the entities, all the transitive dependencies are deleted, and the structure respects the third normal form.

In conclusion, the initial table was confusing and did not follow the normalization rules. To normalize it, it was necessary to distinguish the different entities in different tables and create unique codes for the link tables. The first normal form was met, creating tables in which each row and column contained only one piece of data. The second normal form was satisfied because all attributes depended on the primary key of the entity. The third normal form was satisfied because the structure of all transitive dependencies was deleted.