## NormalizationCheatTable

StudentID	Name	Subjects	DeptID	DeptName	Trainer						
	1 Alice	Math, Science		Computer Sci							
	2 Bob	Physics, Chemi			Ramya, Rajesh			studentID	subjects	name	de
	3 Charlie	Math	D1	Computer Sci					,		
StudentID	Name	Subjects	DeptID	DeptName	Trainer		A	В	С		
	1 Alice	math	d1	comp sci	Ram						
	1 Alice	science	d1	comp sci	Tina						
	2 Bob	physics	d2	physics dept	Ramya						
	2 Bob	Chemistry	d2	physics dept	Rajesh						
	3 Charlie	Math	d1	comp sci	Ram						
StudentID	Name	Subjects	DeptID	Trainer				Deptid	DeptName		
	1 Alice	math	d1	Ram				d1	comp sci		
	1 Alice	science	d1	Tina				d2	physics dept		
	2 Bob	physics	d2	Ramya				dep table			
	2 Bob	Chemistry	d2	Rajesh							
	3 Charlie	Math	d1	Ram				subjectId	SubjectName	trainerId	
								s1	Math	t1	
								s2	Science	t2	
studentId	Name	Deptld	SubjectId					s3	Physics	t3	
	1 Alice	d1	s1					s4	Chemistry	t4	
	1 Alice	d1	s2								
	2 Bob	d2	s3								
	2 Bob	d2	s4					trainerId	TrainerName		
	3 Charlie	d1	s1					t1	Ram		
								t2	Tina		
								t3	Ramya		
								t4	Rajesh		
studentId	Name	Deptld			studentId	SubjectId					
	1 Alice	d2			1	s1					
	2 Bob	d2			1	s2					
	3 Charlie	d1			2	s3					
					2	s4					
					2	s1					

## NormalizationCheatTable

Normal Form	Definition / Condition	Goal / Purpose	Example (Violation)	Example (After Normalization)
1NF (First Normal Form)	A table is in 1NF if: • All attributes have atomic (indivisible) values • No repeating groups or arrays	Eliminate repeating data and ensure atomicity	Violation: Students(StudentID, Name, Subjects) where Subjects = "Math, Science"(multi-valued)	1NF: Students(StudentID, Name, Subject) with separate rows for each subject
2NF (Second Normal Form)	A table is in 2NF if: • It is in 1NF • No partial dependency exists (non-key attribute depends on part of a composite key)	Remove partial dependency	Violation: Orders(OrderID, ProductID, ProductName, Quantity) where (OrderID, ProductID) is the key but ProductName depends only on ProductID	2NF: Split into Orders(OrderID, ProductID, Quantity) and Products(ProductID, ProductName)
3NF (Third Normal Form)	A table is in 3NF if: • It is in 2NF • No transitive dependency exists (non-key attribute depends on another non-key attribute)	Remove transitive dependency	Violation: Employees(EmplD, Name, DeptlD, DeptName) where DeptName depends on DeptlD, not directly on EmplD	3NF: Employees(EmplD, Name, DeptlD) and Departments(DeptlD, DeptName)
BCNF (Boyce–Codd Normal Form)	A table is in BCNF if: • It is in 3NF • For every functional dependency $X \rightarrow Y$ , $X$ is a superkey	Handle anomalies not covered by 3NF	Violation: Courses(CourseID, Teacher, Dept) where Teacher → Dept but Teacher is not a superkey	BCNF: Split into Teachers(Teacher, Dept) and Courses(CourseID, Teacher)
4NF (Fourth Normal Form)	A table is in 4NF if: • It is in BCNF • No multi-valued dependencies exist (one key relates to multiple independent attributes)	Remove multi-valued dependencies	Violation: Projects(EmpID, Skill, Project) where Skill and Project are independent	4NF: Split into EmpSkills(EmpID, Skill) and EmpProjects(EmpID, Project)
5NF (Fifth Normal Form)	A table is in 5NF if: • It is in 4NF • No join dependency exists (table can't be reconstructed from smaller tables without loss of data)	Remove join dependencies	Violation: Sales(Product, Store, Supplier) where data comes from multiple smaller relations	5NF: Split into ProductStore(Product, Store), StoreSupplier(Store, Supplier), ProductSupplier(Product, Supplier)
6NF (Sixth Normal Form)	A table is in 6NF if: • It is in 5NF • All attributes depend on the key, the whole key, and nothing but the key over time (temporal databases)	Handle time-variant data	Violation: Table with multiple time-dependent facts in one row	6NF: Separate each time-dependent fact into its own table