

## Applying NORMAL FORMS

The table contains the following attributes:

R (invention\_id, invention\_name, year\_invention, story, inventor\_id, F\_name, L\_name, DOB, Age, Job\_type, Area, City, Pincode, nomination\_year, Award\_id, Award\_name, Significance, Award\_Category{}, Jury\_id, Jury\_name, Span)

In 1NF, each tuple will have only single-valued attributes. All the multi-valued attributes will be split into individual tuples.

### Now by applying First Normal Form ( 1NF ):

1. An invention can have multiple inventors, so each of the inventor details for a specific invention will be shown in separate tuples.
2. Similarly, an invention can have multiple awards, so each award for a specific invention will be shown in separate tuples.

**From the above Normal Form, the candidate key obtained is:**

invention\_id, award\_id → inventor\_id, invention\_name, year\_invention, story, F\_name, L\_name, DOB, Job\_type, Area, City, Pincode, award\_name, Significance, Category, jury\_id

In 2NF, we eliminate all partial dependencies. Partial dependencies mean that a non-prime attribute is functionally dependent on a part of the candidate key.

### After Applying 2NF:

invention\_id → invention\_name, year\_invention, story,  
inventor\_id, F\_name, L\_name, DOB, Age, Job\_type, Area, City,  
Pincode  
award\_id → award\_name, Significance, Category, span,  
jury\_name, jury\_id  
award\_id, invention\_id → nomination\_year  
award\_id, jury\_id → winner\_invention

Winner	Invention Management	Invention
AwardID	Invention_id	Invention_ID
Jury_ID	Invention Name	Invention_name
Invention_ID	Year_Invention	Year_of_invention
	Story	Story
	Inventor_id	Inventor_ID
	FName	FName
	LName	LName
	DOB	DOB
	Age	Age
	Job_type	Job_type
	Area	Area
	City	City
	Pincode	Pincode
	Nomination Year	
	Award_ID	
	Award_Name	
	Specification	
	Award_Category	
	jury_ID	
	Jury_Name	
	v	

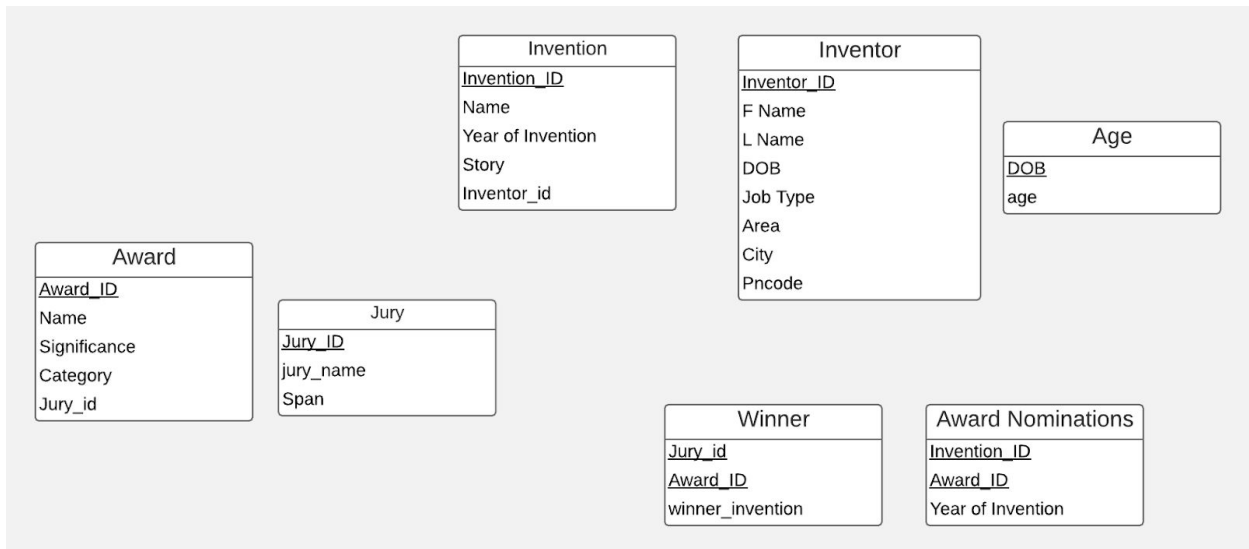
Awards
AwardID
Award_name
Significance
Category
Jury_ID
Jury_name
Span

Nominations
Invention_ID
AwardID
Nomination_Year

In 3NF, we eliminate all transitive dependencies. Transitive dependencies mean that a non-prime attribute is dependent on another attribute which is not a part of the candidate key but is dependent on candidate key.

### After applying 3NF:

inventor\_id → F\_name, L\_name, DOB, Job\_type, Area, City, Pincode  
 DOB → age  
 jury\_id → jury\_name, span  
 invention\_id → invention\_name, year\_invetion, story, inventor\_id  
 award\_id → award\_name, Significance, Category, jury\_id  
 award\_id, invention\_id → nomination\_year  
 award\_id, jury\_id → winner\_invention



## ISSUES:

1. We can't find all inventors name (multiple) with one invention
2. Remove Winner table
3. Remove Age table