Normalization exercises

Francesco Di Giacomo, Ahmad Omar

Note: For each of the following exercises complete the following tasks:

- 1. (Complete after lesson 1) Find what normal form each of the following tables satisfies. Motivate the answer according the definition of normal forms seen in class.
- 2. (Complete after lesson 2) Apply the normalization algorithms seen in class to each table. Use intermediate refinements, i.e. if the table is in 1NF first normalize in 2NF and, if necessary, in BCNF.

Exercise 1 - Lockers The table is in no normal form, since it contains a multi-value attribute.

teacher				
teacher_id	name	surname		

locker				
id	$\underline{\text{teacher_id}}$	key_num	size	

In locker the attribute teacher_id is a foreign key to teacher. The table is already in BCNF since all dependencies have the left argument which is a superkey.

Exercise 2 - Library The table is in no normal form since it contains a multi-value attribute.

1NF:

library		
card_num	name	surname

$borrowed_books$]		
$\underline{\text{author}}$	<u>car_num</u>	$\underline{\text{title}}$	$\underline{\text{date}}$	return_date

In member the attribute card_num is a foreign key to library.

2NF:

BCNF:

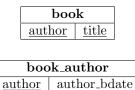
The tables rreated at the previous step are in 2NF because no functional dependencies exist. All the tables at this point are in BCNF because all the dependencies have a superkey as left argument.

Exercise 3 - Books 1NF:

The table is already in 1NF because all the attributes are atomic.

2NF:

Both dependencies have the left argument that is part of a key and the right argument that is a non-key attribute, thus they break 2NF



description				
$\underline{\text{title}}$	genre	page	section	

In the table book the attribute _author is a foreign key to book_author and title is a foreign key to description.

BCNF:

All the tables are already in BCNF since the left argument of all the dependencies are superkeys.

Exercise 4 - Houses The table is in no normal form because it contains a multi-value attribute.

1NF:

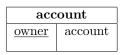
houses					
owner	postal_code	price	size	account	

owner_address					
owner	postal_code	city	street	number	

In owner_address the attribute postal_code is a foreign key to houses

2NF:

Using the decomposition rule on postal_code \rightarrow price, size we get the functional dependencies postal_code \rightarrow price and postal_code \rightarrow size. Both break the 2NF because they are partially dependent on the primary key. Also the dependency owner \rightarrow account breaks 2NF.



property				
postal_code	price	size		

$owner_address$					
owner	postal_code	$\frac{\text{city}}{}$	$\underline{\text{street}}$	number	

In houses the attribute owner is a foreign key to account and postal_code is a foreign key to property

BCNF:

All tables are already in BCNF because the left argument of all the dependencies is a superkey.

Exercise 5 - Port 1NF:

The table is already in 1NF because all attributes are atomic

2NF:

The dependencies $ship_name \rightarrow docked_at$, country, weight, class and captain $\rightarrow cpt_license$ break the 2NF because the left argument is part of a key and the right argument is a non-key attribute (again use the decomposition rule). Note that the dependency $docked_at \rightarrow country$ does not break 2NF because the left argument is not part of any key (the 2NF only considers dependencies where the left side is part of a key)

${ m ship_captain}$			
ship_name	captain		

ship					
ship_name	weight	class	$docked_at$	country	

license		
captain	cpt_license	

In ship_captain the attribute ship_name is foreign key to ship and captain is a foreign key to license.

BCNF:

The dependency docked_at → country breaks BCNF because the left argument is not a superkey. The table ship must be decomposed in BCNF. In the table ship the attribute docked_at is a foreign key to docking.

${ m ship}$			
ship_name	weight	class	docked_at

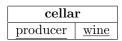
docking			
docked_at country			

Exercise 6 - Cellar 1NF:

The table is in 1NF because it contains only atomic attributes

2NF:

The dependencies producer \rightarrow country,location and wine \rightarrow bottling_date,price/1,grape_variety break the 2NF since the left side is part of a key and the right side is a non-key attribute. The other dependencies do not break 2NF because they do not have a left argument that is part of a key



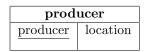
producer		
producer	country	location

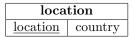
wine				
wine bottling_date price/l grape_variety				

In cellar the attribute producer is a foreign key to the table producer and the attribute wine is a foreign key to the table wine.

BCNF:

The dependencies location \rightarrow country grape_variety \rightarrow price/1 break BCNF because their left argument is not a superkey. The tables wine and producer must be normalized in BCNF.





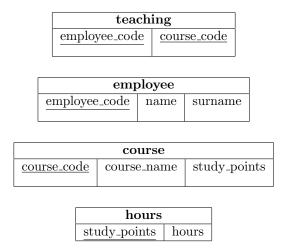
\mathbf{wine}		
$\underline{\text{wine}}$	bottling_date	grape_variety

grape		
grape_variety	price/l	

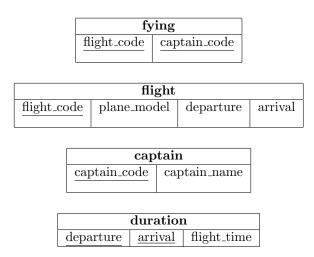
In the table wine the attribute grape_variety is foreign key to grape and in the table producer the attribute location is foreign key to the table location.

Note: The following exercises are analogous to the previous exercises so only the final solution can be found

Exercise 7 - Courses In teaching the attribute employee_code is a foreign key to employee and the attribute course_code is a foreign key to course. In the table course the attribute study_points is a foreign key to hours.



Exercise 8 - Flights In the table flying the attribute flight_code is a foreign key to flight, and the attribute captain_code is a foreign key to captain. In the table flight the attributes { departure, arrival } are a foreign key to duration.



Exercise 9 - Buildings In materials the attribute building_type_code is a foreign key to building and material is a foreign key to the table material. In the table building the attributes { length,width,height } are a foreign key to taxes.

${f building_materials}$			
building_code	$\underline{\text{component_type}}$	material	

material			
material	aximum_pressure	specific_weight	

building					
building_code	building_code building_length building_width building_height				

taxes			
building_length	building_width	building_height	tax_rate