**USER TABLE**

username -> {email\_address, D.O.B, password, description}

email\_address -> {username, D.O.B, password, description}



The candidate keys are username, email\_address.

Prime attributes = {username, email\_address}

Non prime attributes = {D.O.B, password, description}

There are two functional dependencies.

Both satisfy 1NF as all attributes are atomic.

Both also satisfy 2NF as there are no partial dependencies, no non prime attribute is dependent on any proper subset of the candidate keys. (Candidate keys are atomic)

Both also satisfy 3NF as there are no transitive dependencies. Or we can say that for both the functional dependencies, the LHS is a candidate key, hence it is in 3NF.

Both also satisfy BCNF as the LHS of all dependencies are candidate keys.

**USER\_PHN\_NO TABLE**



Attributes username and phone\_no together form a composite key and there are no other attributes.

Hence it is automatically in BCNF. Since all dependencies are trivial.

**CUSTOMER TABLE**

username -> {street, city, state, first\_name, last\_name}



username is a candidate key.

Prime attributes = {username}

Non prime attributes = {street, city, state, first\_name, last\_name}

There is only one functional dependency.

It satisfies 1NF as all attributes are atomic.

It also satisfies 2NF as there are no partial dependencies, no non prime attribute is dependent on any proper subset of the candidate keys. (Candidate key is atomic)

It also satisfies 3NF as there are no transitive dependencies. Or we can say that for the functional dependency, the LHS is a candidate key, hence it is in 3NF.

It also satisfies BCNF as the LHS of all dependencies are candidate keys.

**ADMINISTRATOR TABLE**

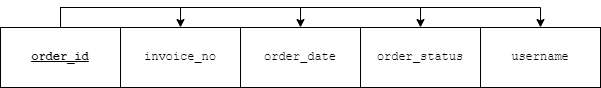
****

There is only one prime attribute and candidate key username.

It is automatically in BCNF because all dependencies are trivial.

**ORDERS TABLE**

order\_id -> {invoice\_no, order\_date, order\_status, username}



order\_id is a candidate key.

Prime attributes = {order\_id}

Non prime attributes = {invoice\_no, order\_date, order\_status, username}

There is only one functional dependency.

It satisfies 1NF as all attributes are atomic.

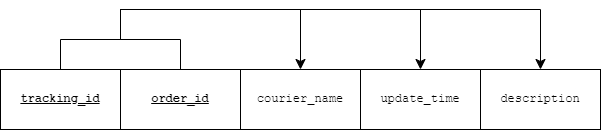
It also satisfies 2NF as there are no partial dependencies, no non prime attribute is dependent on any proper subset of the candidate keys. (Candidate key is atomic)

It also satisfies 3NF as there are no transitive dependencies. Or we can say that for the functional dependency, the LHS is a candidate key, hence it is in 3NF.

It also satisfies BCNF as the LHS of all dependencies are candidate keys.

**TRACKING\_DETAILS TABLE**

tracking\_id order\_id -> {courier\_name, update\_time, description}



tracking\_id order\_id is the candidate key.

Prime attributes = {tracking\_id, order\_id}

Non prime attributes = {courier\_name, update\_time, description}

There is only one functional dependency.

It satisfies 1NF as all attributes are atomic.

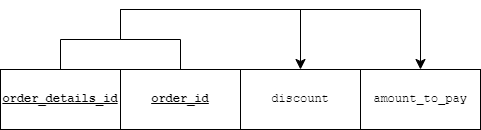
It also satisfies 2NF as there are no partial dependencies, no non prime attribute is dependent on any proper subset of the candidate keys.

It also satisfies 3NF as there are no transitive dependencies. Or we can say that for the functional dependency, the LHS is a candidate key, hence it is in 3NF.

It also satisfies BCNF as the LHS of all dependencies are candidate keys.

**ORDER\_DETAILS TABLE**

order\_details\_id order\_id = {discount, amount\_to\_pay}



order\_details\_id order\_id is the candidate key.

Prime attributes = {order\_details\_id, order\_id}

Non prime attributes = {discount, amount\_to\_pay}

There is only one functional dependency.

It satisfies 1NF as all attributes are atomic.

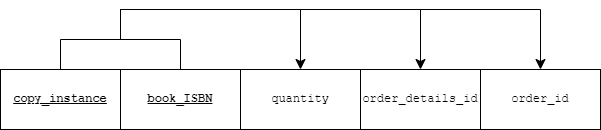
It also satisfies 2NF as there are no partial dependencies, no non prime attribute is dependent on any proper subset of the candidate keys.

It also satisfies 3NF as there are no transitive dependencies. Or we can say that for the functional dependency, the LHS is a candidate key, hence it is in 3NF.

It also satisfies BCNF as the LHS of all dependencies are candidate keys.

**BOOK\_INSTANCE TABLE**

copy\_instance book\_ISBN -> {quantity, order\_details\_id, order\_id}



copy\_instance book\_ISBN is the candidate key.

Prime attributes = {copy\_instance, book\_ISBN}

Non prime attributes = {quantity, order\_details\_id, order\_id}

There is only one functional dependency.

It satisfies 1NF as all attributes are atomic.

It also satisfies 2NF as there are no partial dependencies, no non prime attribute is dependent on any proper subset of the candidate keys.

It also satisfies 3NF as there are no transitive dependencies. Or we can say that for the functional dependency, the LHS is a candidate key, hence it is in 3NF.

It also satisfies BCNF as the LHS of all dependencies are candidate keys.

**AUTHOR TABLE**

author\_id -> {author\_name, email\_address}

email\_address -> {author\_id, author\_name}



author\_id and email\_address are the candidate keys.

Prime attributes = {author\_id, email\_address}

Non prime attributes = {author\_name}

There are two functional dependencies.

Both satisfy 1NF as all attributes are atomic.

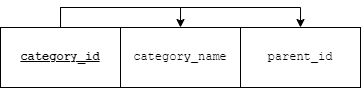
Both also satisfy 2NF as there are no partial dependencies, no non prime attribute is dependent on any proper subset of the candidate keys. (Candidate keys are atomic)

Both also satisfy 3NF as there are no transitive dependencies. Or we can say that for both the functional dependencies, the LHS is a candidate key, hence it is in 3NF.

Both also satisfy BCNF as the LHS of all dependencies are candidate keys.

**CATEGORY TABLE**

category\_id = {category\_name, parent\_id}



category\_id is a candidate key.

Prime attribute = {category\_id}

Non prime attribute = {category\_name, parent\_id}

There is only one functional dependency.

It satisfies 1NF as all attributes are atomic.

It also satisfies 2NF as there are no partial dependencies, no non prime attribute is dependent on any proper subset of the candidate keys.

It also satisfies 3NF as there are no transitive dependencies. Or we can say that for the functional dependency, the LHS is a candidate key, hence it is in 3NF.

It also satisfies BCNF as the LHS of all dependencies are candidate keys.

**BOOK\_CATEGORY TABLE**

****

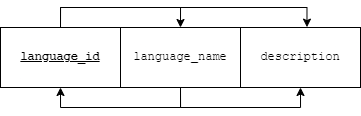
There are only two prime attributes category\_id and book\_ISBN.

It is automatically in BCNF because all dependencies are trivial.

**LANGUAGE TABLE**

language\_id = {language\_name, description}

language\_name = {language\_id, description}



language\_id and language\_name are candidate keys.

Prime attributes = {language\_id, language\_name}

Non prime attributes = {description}

There are two functional dependencies.

Both satisfy 1NF as all attributes are atomic.

Both also satisfy 2NF as there are no partial dependencies, no non prime attribute is dependent on any proper subset of the candidate keys. (Candidate keys are atomic)

Both also satisfy 3NF as there are no transitive dependencies. Or we can say that for both the functional dependencies, the LHS is a candidate key, hence it is in 3NF.

Both also satisfy BCNF as the LHS of all dependencies are candidate keys.

**BOOK\_LANGUAGE TABLE**



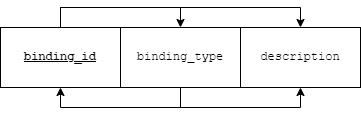
There are only two prime attributes language\_id and book\_ISBN.

It is automatically in BCNF because all dependencies are trivial.

**BINDING TABLE**

binding\_id = {binding \_type, description}

binding \_type = {binding \_id, description}



binding \_id and binding \_type are candidate keys.

Prime attributes = {binding\_id, binding \_type}

Non prime attributes = {description}

There are two functional dependencies.

Both satisfy 1NF as all attributes are atomic.

Both also satisfy 2NF as there are no partial dependencies, no non prime attribute is dependent on any proper subset of the candidate keys. (Candidate keys are atomic)

Both also satisfy 3NF as there are no transitive dependencies. Or we can say that for both the functional dependencies, the LHS is a candidate key, hence it is in 3NF.

Both also satisfy BCNF as the LHS of all dependencies are candidate keys.

**BOOK\_BINDING TABLE**

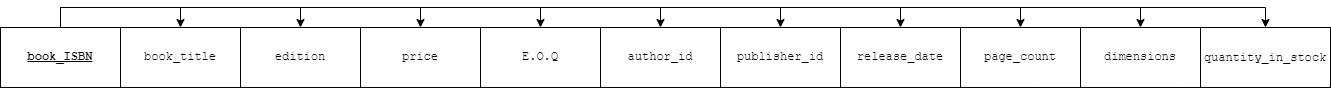


There are only two prime attributes binding\_id and book\_ISBN.

It is automatically in BCNF because all dependencies are trivial.

**BOOK TABLE**

book\_ISBN -> {book\_title, edition, price, E.O.Q, author\_id, publisher\_id, release\_date, page\_count, dimension, quantity\_in\_stock}



book\_ISBN is the candidate key.

Prime attributes = {book\_ISBN}

Non prime attributes = {book\_title, edition, price, E.O.Q, author\_id, publisher\_id, release\_date, page\_count, dimension}

There is only one functional dependency.

It satisfies 1NF as all attributes are atomic.

It also satisfies 2NF as there are no partial dependencies, no non prime attribute is dependent on any proper subset of the candidate keys.

It also satisfies 3NF as there are no transitive dependencies. Or we can say that for the functional dependency, the LHS is a candidate key, hence it is in 3NF.

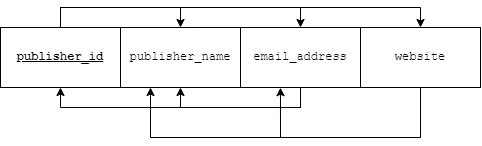
It also satisfies BCNF as the LHS of all dependencies are candidate keys.

**PUBLISHER TABLE**

publisher\_id = {publisher\_name, email\_address, website}

email\_address = {publisher\_name, publisher\_id}

website = {email\_address, publisher\_name}



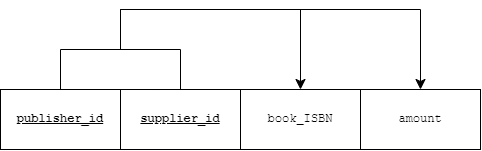
If we check their closures, publisher\_id, email\_address, website are candidate keys.

All three are prime attributes.

It is automatically in BCNF because all dependencies are trivial.

**SUPPLIES\_TO TABLE**

publisher\_id supplier\_id = {book\_ISBN, amount}



publisher\_id supplier\_id is the candidate key.

Prime attributes = {publisher\_id, supplier\_id}

Non prime attributes = {book\_ISBN, amount}

There is only one functional dependency.

It satisfies 1NF as all attributes are atomic.

It also satisfies 2NF as there are no partial dependencies, no non prime attribute is dependent on any proper subset of the candidate keys.

It also satisfies 3NF as there are no transitive dependencies. Or we can say that for the functional dependency, the LHS is a candidate key, hence it is in 3NF.

It also satisfies BCNF as the LHS of all dependencies are candidate keys.

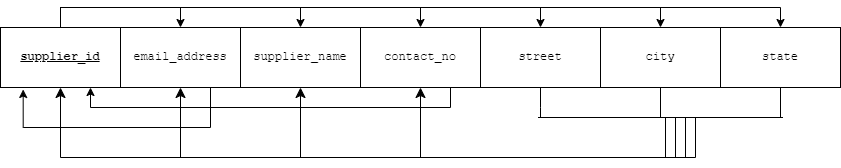
**SUPPLIER TABLE**

supplier\_id = {email\_address, supplier\_name, contact\_no, street, city, state}

street city state = {supplier\_id, email\_address, supplier\_name, contact\_no}

contact\_no = {supplier\_id}

email\_address = {supplier\_id}



supplier\_id, contact\_no, email\_address and street, city, state are candidate keys.

Prime attributes = {supplier\_id, street city state, contact\_no, email\_address}

Non prime attributes = {supplier\_name}

It satisfies 1NF as all attributes are atomic.

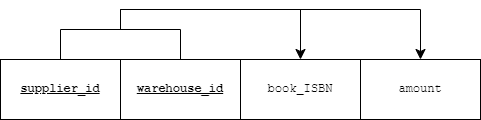
It also satisfies 2NF as there are no partial dependencies, no non prime attribute is dependent on any proper subset of the candidate keys.

It also satisfies 3NF as there are no transitive dependencies. Or we can say that for the functional dependency, the LHS is a candidate key, hence it is in 3NF.

It also satisfies BCNF as the LHS of all dependencies are candidate keys.

**PROVIDES\_BOOKS TABLE**

supplier\_id warehouse\_id = {book\_ISBN, amount}



supplier\_id warehouse\_id is the candidate key.

Prime attributes = {supplier\_id, warehouse\_id}

Non prime attributes = {book\_ISBN, amount}

There is only one functional dependency.

It satisfies 1NF as all attributes are atomic.

It also satisfies 2NF as there are no partial dependencies, no non prime attribute is dependent on any proper subset of the candidate keys.

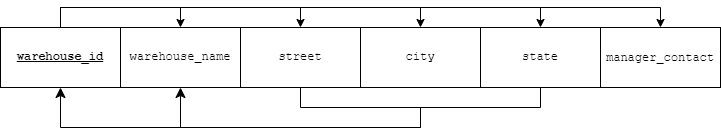
It also satisfies 3NF as there are no transitive dependencies. Or we can say that for the functional dependency, the LHS is a candidate key, hence it is in 3NF.

It also satisfies BCNF as the LHS of all dependencies are candidate keys.

**WAREHOUSE TABLE**

warehouse\_id = {warehouse\_name, street, city, state, manager\_contact}

street city state = {warehouse\_id, warehouse\_name}



warehouse\_id, street city state are the candidate keys.

Prime attributes = {warehouse\_id, street, city, state}

Non prime attributes = {warehouse\_name}

It satisfies 1NF as all attributes are atomic.

It also satisfies 2NF as there are no partial dependencies, no non prime attribute is dependent on any proper subset of the candidate keys.

It also satisfies 3NF as there are no transitive dependencies. Or we can say that for the functional dependency, the LHS is a candidate key, hence it is in 3NF.

It also satisfies BCNF as the LHS of all dependencies are candidate keys.

**USER\_INTERACTION TABLE**

username book\_ISBN = {rating, upvote, downvote}



username book\_ISBN is the candidate key.

Prime attributes = {username, book\_ISBN}

Non prime attributes = {rating, upvote, downvote}

There is only one functional dependency.

It satisfies 1NF as all attributes are atomic.

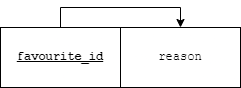
It also satisfies 2NF as there are no partial dependencies, no non prime attribute is dependent on any proper subset of the candidate keys.

It also satisfies 3NF as there are no transitive dependencies. Or we can say that for the functional dependency, the LHS is a candidate key, hence it is in 3NF.

It also satisfies BCNF as the LHS of all dependencies are candidate keys.

**FAVOURITES TABLE**

favourite\_id = {reason}



favourite\_id is the candidate key.

It is automatically in BCNF as the LHS are all candidate keys.

**USER\_FAVOURITE TABLE**

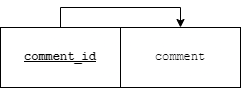


All are prime attributes.

It is automatically in BCNF as all dependencies are trivial.

**COMMENTS TABLE**

comment\_id = {comment}



comment\_id is the candidate key.

It is automatically in BCNF as the LHS are all candidate keys.

**USER\_COMMENT TABLE**



All are prime attributes.

It is automatically in BCNF as all dependencies are trivial.

**WISHLIST\_TABLE**



There is only one prime attribute.

It is automatically in BCNF as all dependencies are trivial.

**USER\_WISHLIST TABLE**



All are prime attributes.

It is automatically in BCNF as all dependencies are trivial.

**HAVE\_BOOKS TABLE**

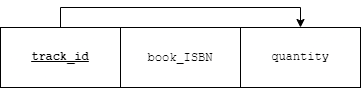
****

All are prime attributes.

It is automatically in BCNF as all dependencies are trivial.

**BOOKS\_WITH\_SUPPLIERS TABLE**

track\_id = {book\_ISBN, quantity}



track\_id is the candidate key.

It is automatically in BCNF as the LHS are all candidate keys.

**CONTAINS\_BOOKS TABLE**

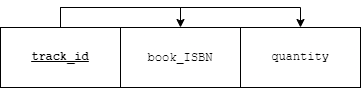
****

All are prime attributes.

It is automatically in BCNF as all dependencies are trivial.

**BOOKS\_IN\_WAREHOUSE TABLE**

track\_id = {book\_ISBN, quantity}



track\_id is the candidate key.

It is automatically in BCNF as the LHS are all candidate keys.