

Python Course

Tutorial 6 - Database and SQL Introduction

UNI
FREIBURG



Albert-Ludwigs-Universität Freiburg

Felix Häschel

Department of Quantitative Finance, Albert-Ludwigs-University Freiburg

Winter Term 2025/26

Exercise

Solution

Exercise

Solution

Exercise

Consider the following unnormalized table:

OrderID	CustomerID	CustomerAddress	Item	Quantity
1	1001	123 Main St	Widget A	2
			Widget B	1
2	1002	456 Elm St	Widget C	5
3	1001	123 Main St	Widget A	3
			Widget D	2

- 1 Convert the unnormalized table into First Normal Form (1NF).
- 2 Identify the functional dependencies and the primary key in your 1NF table.
- 3 Normalize the 1NF table into Second Normal Form (2NF).
- 4 Normalize the 2NF table into Third Normal Form (3NF).

Solution



(i) Conversion to First Normal Form (1NF)

Exercise

Solution

Definition (1NF):

A table is in 1NF if it contains *no repeating groups* and all values are *atomic*.

In the given table, the repeating group arises because one **OrderID** contains multiple items on separate lines. We therefore create one row per item.

OrderID	CustomerID	CustomerAddress	Item	Quantity
1	1001	123 Main St	Widget A	2
1	1001	123 Main St	Widget B	1
2	1002	456 Elm St	Widget C	5
3	1001	123 Main St	Widget A	3
3	1001	123 Main St	Widget D	2



(ii) Functional Dependencies and Primary Key

Exercise

Solution

Key idea: Functional dependencies describe which attributes determine others.

Identifiable FDs from the 1NF table:

- **OrderID** \rightarrow CustomerID, CustomerAddress (each order is placed by exactly one customer)
- **CustomerID** \rightarrow CustomerAddress (customer information is independent of the order)
- **{OrderID, Item}** \rightarrow Quantity (the amount refers to a specific item in a specific order)

Primary Key: {OrderID, Item}

This composite key uniquely identifies each record.



(iii) Conversion to Second Normal Form (2NF)

Definition (2NF):

A table is in 2NF if it is in 1NF and contains *no partial dependencies*.

In our 1NF table, CustomerID and CustomerAddress depend only on **OrderID**, not on the full primary key {OrderID, Item}. → Violates 2NF.

Therefore, we decompose into:

Order Table:

OrderID	CustomerID	CustomerAddress
1	1001	123 Main St
2	1002	456 Elm St
3	1001	123 Main St

OrderItem Table:

OrderID	Item	Quantity
1	Widget A	2
1	Widget B	1
2	Widget C	5
3	Widget A	3
3	Widget D	2



(iv) Conversion to Third Normal Form (3NF)

Definition (3NF):

A table is in 3NF if it is in 2NF and has *no transitive dependencies*.

In the **Order** table:

$\text{CustomerID} \rightarrow \text{CustomerAddress}$

Thus, CustomerAddress depends transitively on OrderID. \rightarrow Violates 3NF.

We therefore isolate customer information.

Customer Table:

CustomerID	CustomerAddress
1001	123 Main St
1002	456 Elm St

Updated **Order** Table:

OrderID	CustomerID
1	1001
2	1002
3	1001

Order Table:

OrderID	CustomerID
1	1001
2	1002
3	1001

OrderItem Table:

OrderID	Item	Quantity
1	Widget A	2
1	Widget B	1
2	Widget C	5
3	Widget A	3
3	Widget D	2

Customer Table:

CustomerID	CustomerAddress
1001	123 Main St
1002	456 Elm St