

Phase7

Toronto Metropolitan University

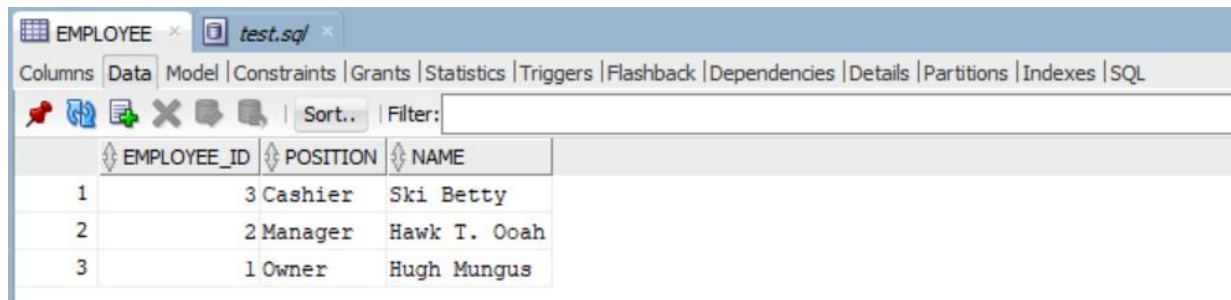
Simon Lin (501103322),  
Dylan Ha (501056670),  
Enes Polat (501061594)

CPS510 - Database Systems

Point of Sale System for Shopper Drug Marts

## Database Normalization:

Employee (Employee\_ID#, Position, Name)



The screenshot shows a database management interface with a tab for 'EMPLOYEE' and another for 'test.sql'. The 'EMPLOYEE' tab is active, displaying a table with three columns: 'EMPLOYEE\_ID', 'POSITION', and 'NAME'. The table contains three rows of data.

	EMPLOYEE_ID	POSITION	NAME
1	3	Cashier	Ski Betty
2	2	Manager	Hawk T. Ooah
3	1	Owner	Hugh Mungus

Functional Dependencies:

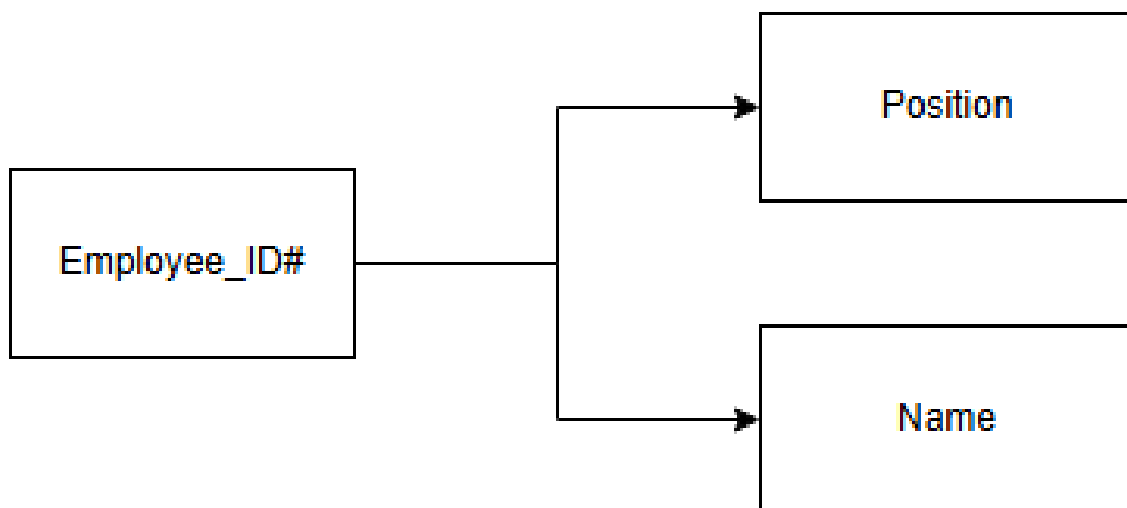
$\{\text{Employee\_ID}\} \rightarrow \text{Position}$

$\{\text{Employee\_ID}\} \rightarrow \text{Name}$

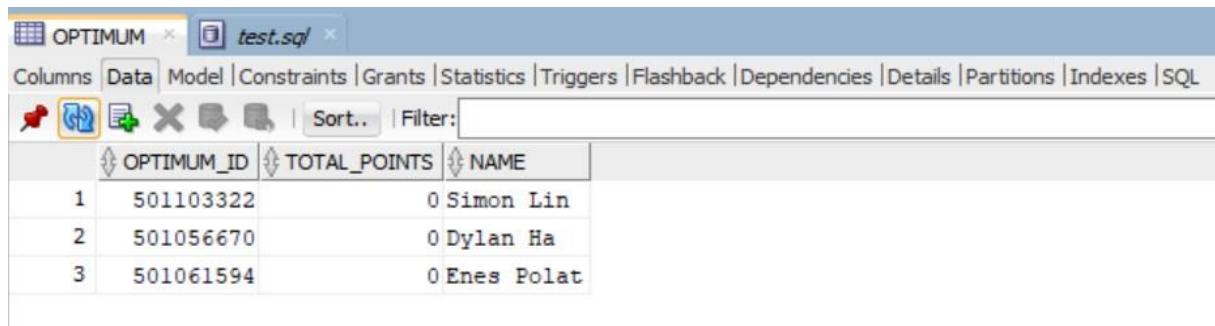
The Employee table relation is in **1NF**, all values contained in the table are atomic.

The Employee table relation is in **2NF**, all non-key values contained in the table are fully functionally dependent on the primary key, there are no partial dependencies.

The Employee table relation is in **3NF**, all non-key values contained in the table are non-transitive and dependent on only the primary key.



Optimum(Optimum\_ID#, Total\_Points, Name)



The screenshot shows a database management tool interface with a tab labeled 'test.sql'. Below the tab is a menu bar with options: Columns, Data, Model, Constraints, Grants, Statistics, Triggers, Flashback, Dependencies, Details, Partitions, Indexes, and SQL. Below the menu bar is a toolbar with icons for various database operations. Below the toolbar is a table with the following data:

	OPTIMUM_ID	TOTAL_POINTS	NAME
1	501103322	0	Simon Lin
2	501056670	0	Dylan Ha
3	501061594	0	Enes Polat

Functional Dependencies:

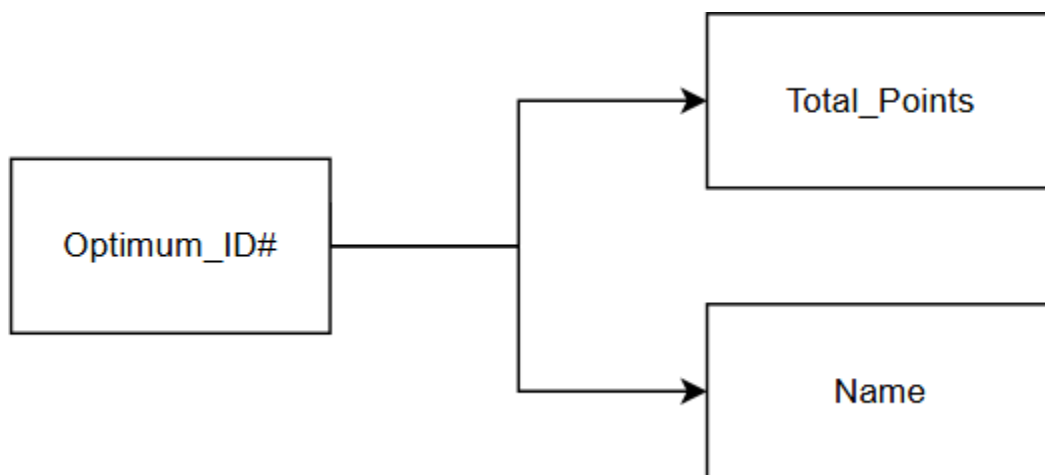
$\{\text{Optimum\_ID}\} \rightarrow \text{Total\_Points}$

$\{\text{Optimum\_ID}\} \rightarrow \text{Name}$

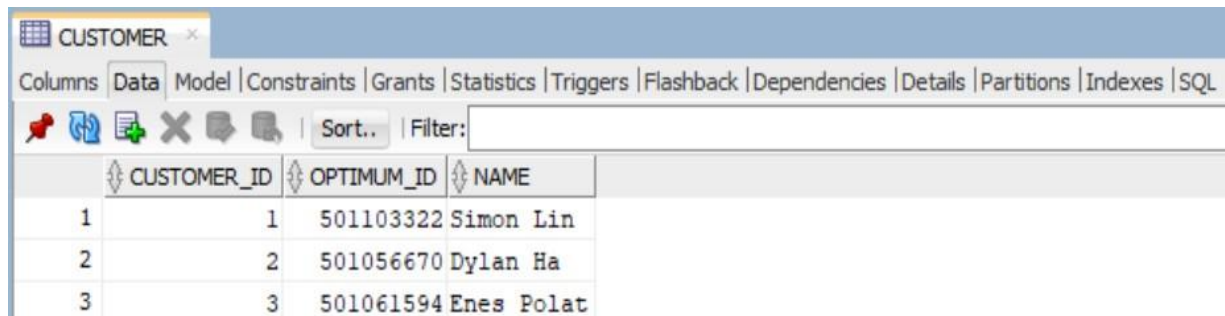
The Optimum table relation is in **1NF**, all values contained in the table are atomic.

The Optimum table relation is in **2NF**, all non-key values contained in the table are fully functionally dependent on the primary key, there are no partial dependencies.

The Optimum table relation is in **3NF**, all non-key values contained in the table are non-transitive and dependent on only the primary key.



Customer(Employee ID#, Position, Name)



A screenshot of a database management system window titled 'CUSTOMER'. The window has a menu bar with options: Columns, Data, Model, Constraints, Grants, Statistics, Triggers, Flashback, Dependencies, Details, Partitions, Indexes, and SQL. Below the menu bar is a toolbar with icons for various database operations. The main area displays a table with three columns: CUSTOMER\_ID, OPTIMUM\_ID, and NAME. The table contains three rows of data.

	CUSTOMER_ID	OPTIMUM_ID	NAME
1	1	501103322	Simon Lin
2	2	501056670	Dylan Ha
3	3	501061594	Enes Polat

Functional Dependencies:

$\{ \text{Customer\_ID} \} \rightarrow \text{Optimum\_ID}$

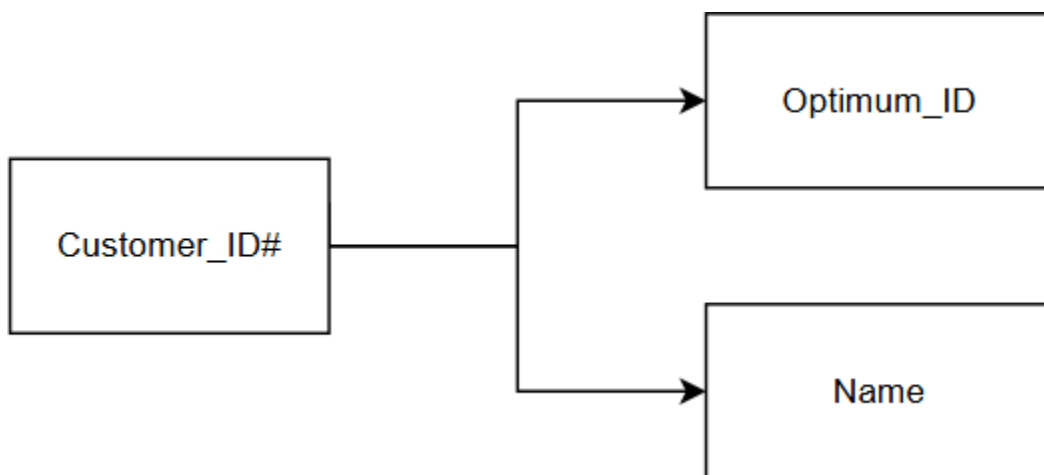
$\{ \text{Customer\_ID} \} \rightarrow \text{Name}$

The Customer table relation is in **1NF**, all values contained in the table are atomic.

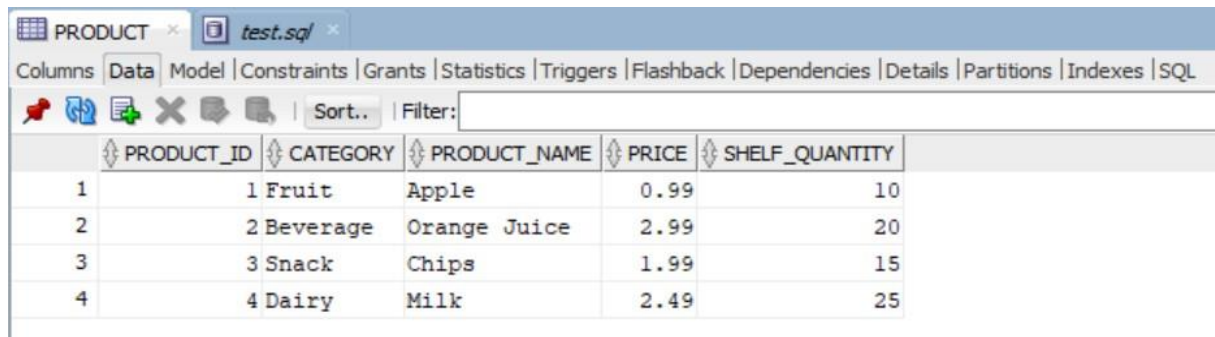
The Customer table relation is in **2NF**, all non-key values contained in the table are fully functionally dependent on the primary key, there are no partial dependencies.

The Customer table relation is in **3NF**, all non-key values contained in the table are non-transitive and dependent on only the primary key.

Note: Name in customer and employee table refers to different people, therefore the relationship is 3NF, i.e a Customer cannot be an Employee, vice versa.



Product(Product\_ID#, Category, Product\_Name, Price, Shelf\_Quantity)



The screenshot shows a database management interface with a tab labeled 'PRODUCT' and a file named 'test.sql'. Below the tabs are icons for various database operations and a 'Filter:' input field. The main area displays a table with the following data:

	PRODUCT_ID	CATEGORY	PRODUCT_NAME	PRICE	SHELF_QUANTITY
1	1	Fruit	Apple	0.99	10
2	2	Beverage	Orange Juice	2.99	20
3	3	Snack	Chips	1.99	15
4	4	Dairy	Milk	2.49	25

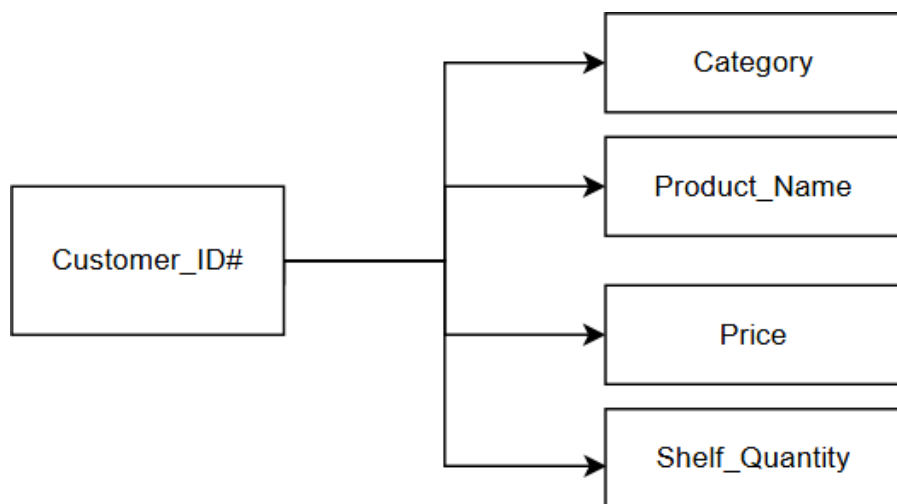
Functional Dependencies:

$\{Product\_ID\} \rightarrow Category$   
 $\{Product\_ID\} \rightarrow Product\_Name$   
 $\{Product\_ID\} \rightarrow Price$   
 $\{Product\_ID\} \rightarrow Shelf\_Quantity$

The Product table relation is in **1NF**, all values contained in the table are atomic.

The Product table relation is in **2NF**, all non-key values contained in the table are fully functionally dependent on the primary key, there are no partial dependencies.

The Product table relation is in **3NF**, all non-key values contained in the table are non-transitive and dependent on only the primary key.



Transaction(Transaction ID#, Employee ID#, Total\_PointsSpent, Total\_Price, Payment\_Method, Transaction\_Date)

The screenshot shows a database management tool interface with a tab for 'TRANSACTION' and a file named 'test.sql'. The 'Data' tab is selected, displaying a table with columns: TRANSACTION\_ID, EMPLOYEE\_ID, TOTAL\_POINTS\_SPENT, TOTAL\_PRICE, PAYMENT\_METHOD, and TRANSACTION\_DATE. The first row of data is: 1, 1, 3, 500, 23.59 Cash, 24-SEP-24.

TRANSACTION_ID	EMPLOYEE_ID	TOTAL_POINTS_SPENT	TOTAL_PRICE	PAYMENT_METHOD	TRANSACTION_DATE
1	1	3	500	23.59 Cash	24-SEP-24

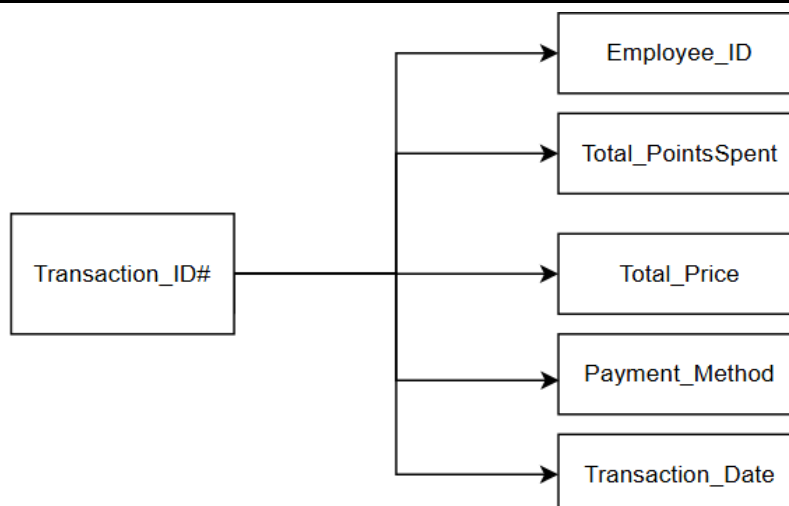
Functional Dependencies:

$\{ \text{Transaction\_ID} \} \rightarrow \text{Employee\_ID}$   
 $\{ \text{Transaction\_ID} \} \rightarrow \text{Total\_PointsSpent}$   
 $\{ \text{Transaction\_ID} \} \rightarrow \text{Total\_Price}$   
 $\{ \text{Transaction\_ID} \} \rightarrow \text{Payment\_Method}$   
 $\{ \text{Transaction\_ID} \} \rightarrow \text{Transaction\_Date}$

The Transaction table relation is in **1NF**, all values contained in the table are atomic.

The Transaction table relation is in **2NF**, all non-key values contained in the table are fully functionally dependent on the primary key, there are no partial dependencies.

The Transaction table relation is in **3NF**, all non-key values contained in the table are non-transitive and dependent on only the primary key.



Receipt(Transaction\_ID#, Product\_List, Points\_Earned, Total\_Price, Payment\_Method, Transaction\_Date)

TRANSAC...	PRODUCT_LIST	POINTS_E...	TOTAL_P...	PAYMENT...	TRANSACTION_DATE
1	1 x22 Apples	500	23.59	Cash	24-SEP-24

Functional Dependencies:

$\{ \text{Transaction\_ID} \} \rightarrow \text{Product\_List}$   
 $\{ \text{Transaction\_ID} \} \rightarrow \text{Points\_Earned}$   
 $\{ \text{Transaction\_ID} \} \rightarrow \text{Total\_Price}$   
 $\{ \text{Transaction\_ID} \} \rightarrow \text{Payment\_Method}$   
 $\{ \text{Transaction\_ID} \} \rightarrow \text{Transaction\_Date}$

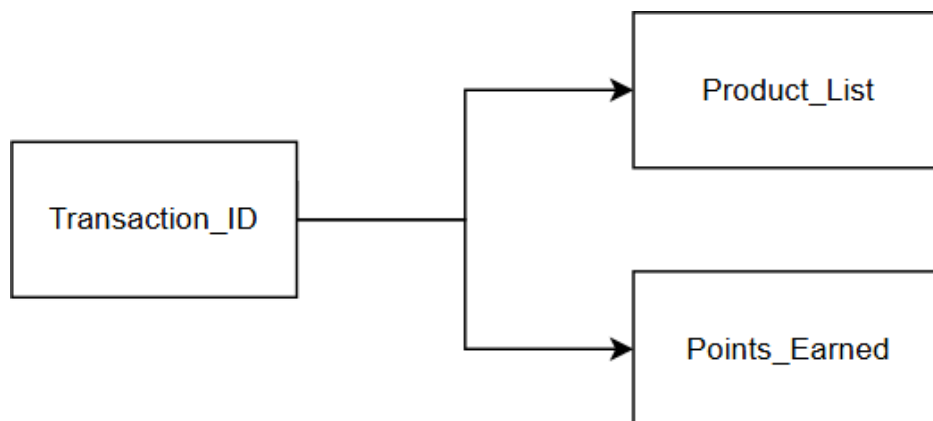
Function Dependencies  
Removed Redundancy:

$\{ \text{Transaction\_ID} \} \rightarrow \text{Product\_List}$   
 $\{ \text{Transaction\_ID} \} \rightarrow \text{Points\_Earned}$

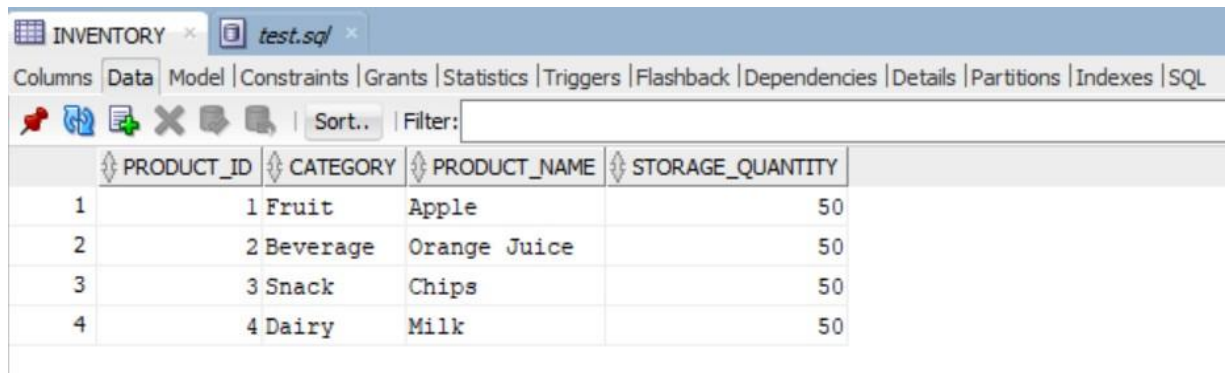
The Receipt table relation is in **1NF**, all values contained in the table are atomic.

The Receipt table relation is in **2NF**, all non-key values contained in the table are fully functionally dependent on the primary key, there are no partial dependencies.

The Receipt table relation is in **3NF**, all non-key values contained in the table are non-transitive and dependent on only the primary key.



## Inventory(Product\_ID#, Category, Product\_Name, Storage\_Quantity)



The screenshot shows a database management interface with a tab labeled 'test.sql'. Below the tab is a menu bar with options: Columns, Data, Model, Constraints, Grants, Statistics, Triggers, Flashback, Dependencies, Details, Partitions, Indexes, and SQL. Below the menu bar is a toolbar with icons for various database operations. Below the toolbar is a table with the following data:

	PRODUCT_ID	CATEGORY	PRODUCT_NAME	STORAGE_QUANTITY
1	1	Fruit	Apple	50
2	2	Beverage	Orange Juice	50
3	3	Snack	Chips	50
4	4	Dairy	Milk	50

Functional Dependencies:

$\{Product\_ID\} \rightarrow Category$   
 $\{Product\_ID\} \rightarrow Product\_Name$   
 $\{Product\_ID\} \rightarrow Storage\_Quantity$

Function Dependencies Removed  
Redundancy:

$\{Product\_ID\} \rightarrow Storage\_Quantity$

The Inventory table relation is in **1NF**, all values contained in the table are atomic.

The Inventory table relation is in **2NF**, all non-key values contained in the table are fully functionally dependent on the primary key, there are no partial dependencies.

The Inventory table relation is in **3NF**, all non-key values contained in the table are non-transitive and dependent on only the primary key.

