

# Introduction to Data Modeling

INTRODUCTION TO DATA MODELING IN SNOWFLAKE



**Nuno Rocha**  
Director of Engineering

# Your instructors

Nuno Rocha

Director of Engineering



Margarita Torres

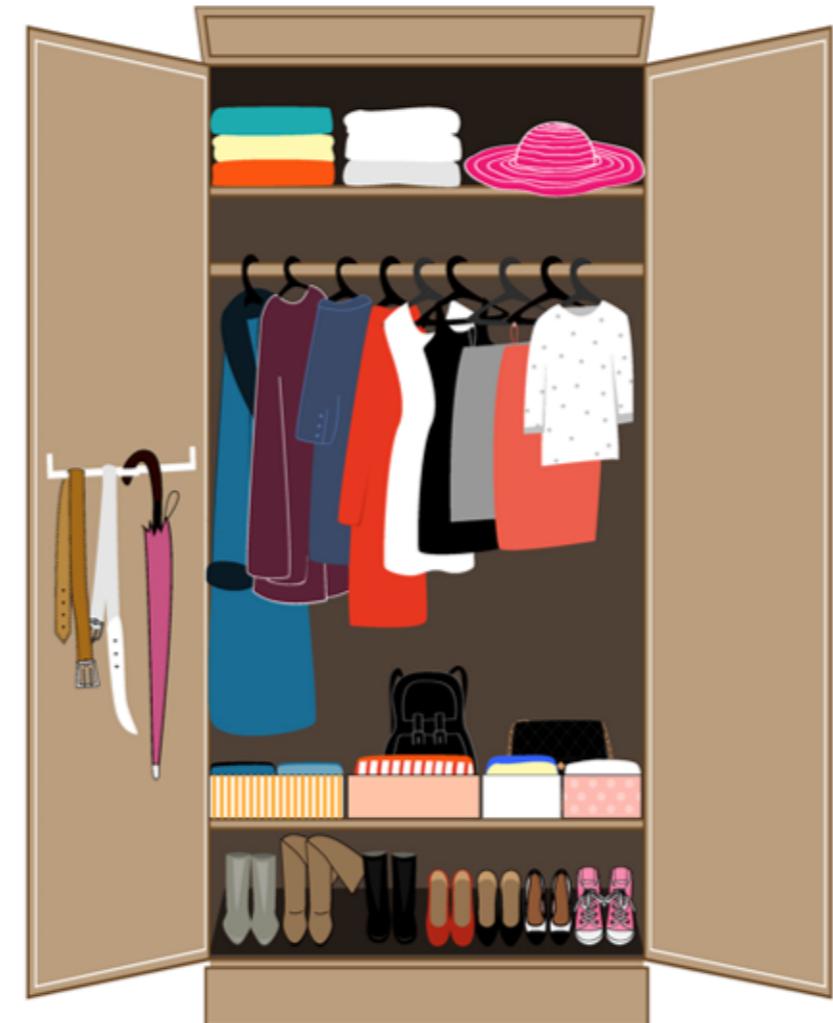
Senior Data Engineer



# What is data modeling?



# What is data modeling? (1)



# What is data modeling? (2)



# Components of data modeling



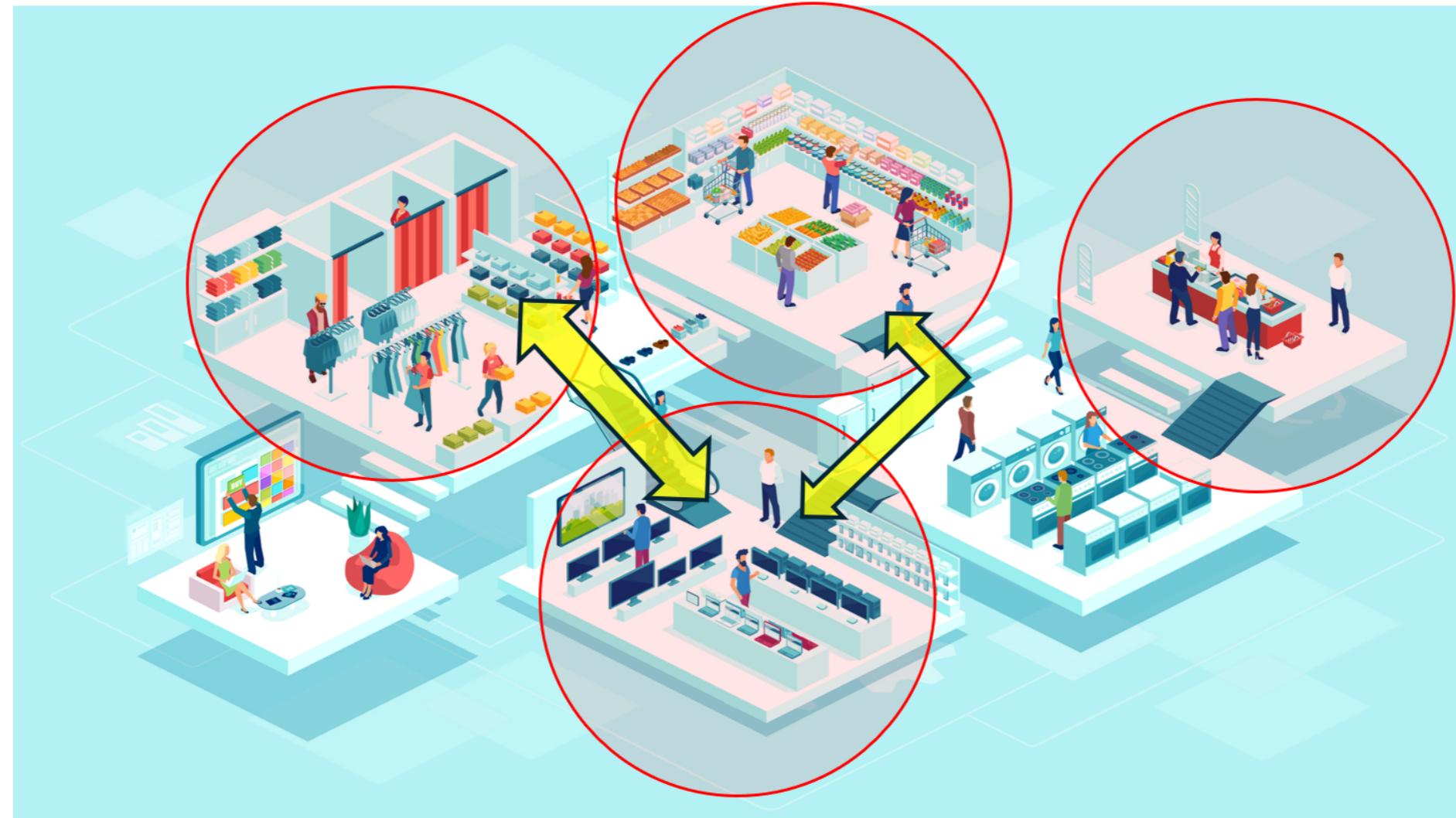
# Components of data modeling (1)



# Components of data modeling (2)



# Components of data modeling (3)



# Describing entities

**Entity:** Object or concept represented in a data model



# Describing entities (1)

**DESC TABLE** : SQL command to display a table's structure

**DESC TABLE** groceries;

<b>name</b>	<b>type</b>	<b>kind</b>	<b>null?</b>	<b>default</b>	<b>primary key</b>	<b>unique key</b>	<b>check expression</b>	<b>comment</b>	<b>policy name</b>
productid	DECIMAL(10,2)	column	N	null	Y	N	null	null	null
productname	VARCHAR(255)	column	N	null	N	N	null	null	null
category	VARCHAR(255)	column	Y	null	N	N	null	null	null
price	DECIMAL(10,2)	column	N	0	N	N	null	null	null
quantityavailable	DECIMAL(10,2)	column	N	0	N	N	null	null	null
suppliername	VARCHAR(255)	column	Y	null	N	N	null	null	null
expirationdate	TIMESTAMP_NTZ(9)	column	Y	null	N	N	null	null	null

# Describing entities (2)

**Attribute:** Characteristics of an entity

<b>name</b>	<b>type</b>	<b>kind</b>	<b>null?</b>	<b>default</b>	<b>primary key</b>	<b>unique key</b>	<b>check expression</b>	<b>comment</b>	<b>policy</b>	<b>name</b>
productid	DECIMAL(10,2)	column	N	null	Y	N	null	null	null	null
productname	VARCHAR(255)	column	N	null	N	N	null	null	null	null
category	VARCHAR(255)	column	Y	null	N	N	null	null	null	null
price	DECIMAL(10,2)	column	N	0	N	N	null	null	null	null
quantityavailable	DECIMAL(10,2)	column	N	0	N	N	null	null	null	null
suppliername	VARCHAR(255)	column	Y	null	N	N	null	null	null	null
expirationdate	TIMESTAMP_NTZ(9)	column	Y	null	N	N	null	null	null	null

# Another industry

## E-Commerce online retail

invoiceno	stockcode	description	quantity	invoicedate	unitprice	customerid	country
545218	22386	JUMBO BAG PINK POLKADOT	10	2/28/11 5:01 PM	1.95	17462	United Kingdom
545218	21933	PINK VINTAGE PAISLEY PICNIC BAG	5	2/28/11 5:01 PM	2.95	17462	United Kingdom
545214	22796	PHOTO FRAME 3 CLASSIC HANGING	2	2/28/11 4:39 PM	9.95	17165	United Kingdom
545214	22699	ROSES REGENCY TEACUP AND SAUCER	6	2/28/11 4:39 PM	2.95	17165	United Kingdom
545214	22151	PLACE SETTING WHITE HEART	24	2/28/11 4:39 PM	0.42	17165	United Kingdom
545213	22321	BIRD DECORATION RED RETROSPOT	12	2/28/11 4:36 PM	0.85	16518	United Kingdom
545213	22148	EASTER CRAFT 4 CHICKS	12	2/28/11 4:36 PM	1.95	16518	United Kingdom
545213	21880	RED RETROSPOT TAPE	12	2/28/11 4:36 PM	0.65	16518	United Kingdom
545191	20725	LUNCH BAG RED RETROSPOT	10	2/28/11 3:39 PM	1.65	12811	Portugal
545191	22662	LUNCH BAG DOLLY GIRL DESIGN	10	2/28/11 3:39 PM	1.65	12811	Portugal

# Terminology and functions overview

- **Data model:** A representation that outlines the organization and storage of data
- **Entity:** Object or concept represented in a data model
- **Attribute:** Characteristics of an entity
- **Relationship:** Connections between entities
- **DESC TABLE** : SQL command to display a table's structure

```
DESC TABLE table_name;
```

# **Let's practice!**

**INTRODUCTION TO DATA MODELING IN SNOWFLAKE**

# Exploring Conceptual and Logical Data Models

INTRODUCTION TO DATA MODELING IN SNOWFLAKE

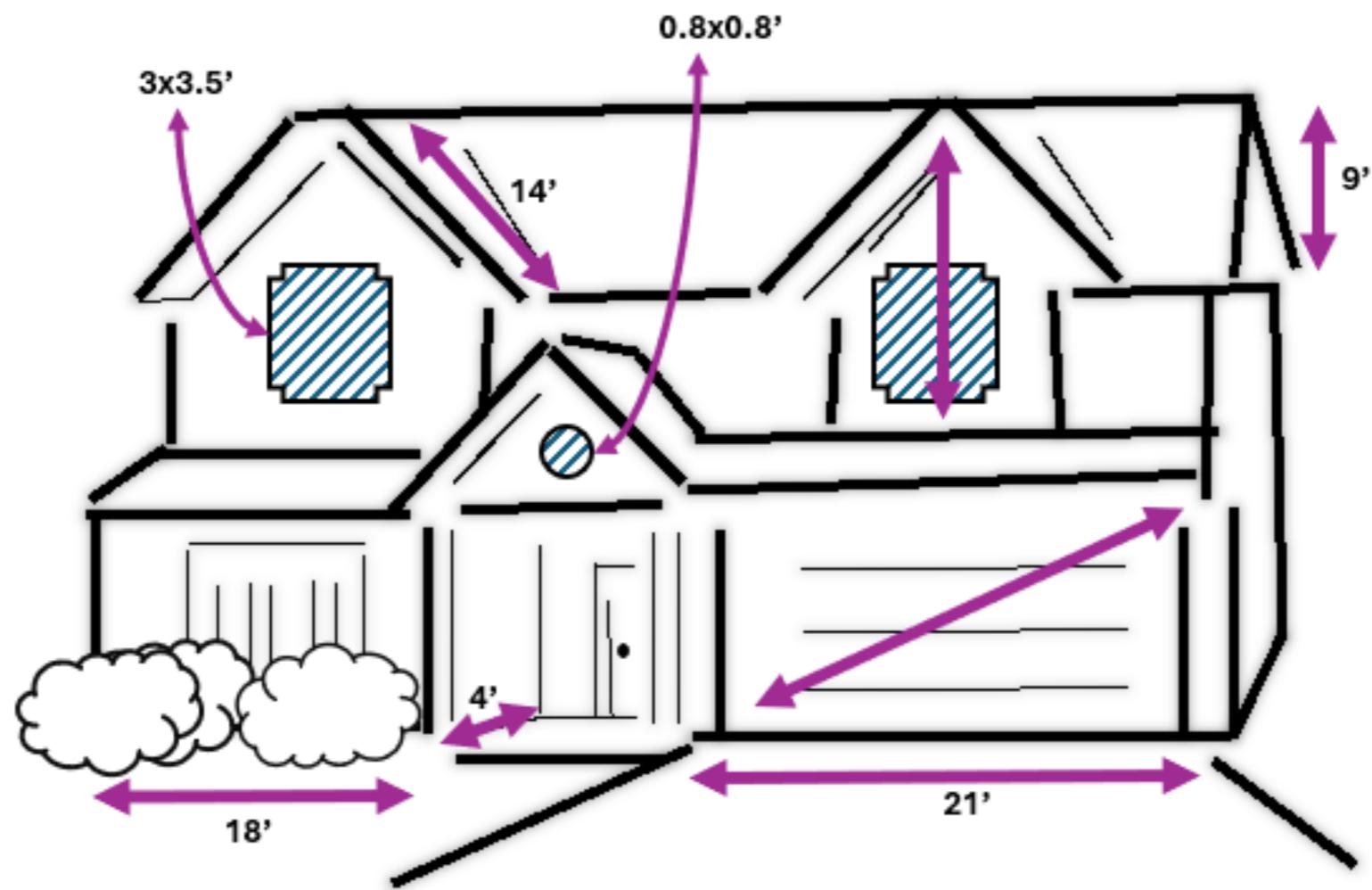


Nuno Rocha  
Director of Engineering

# What is a conceptual model?

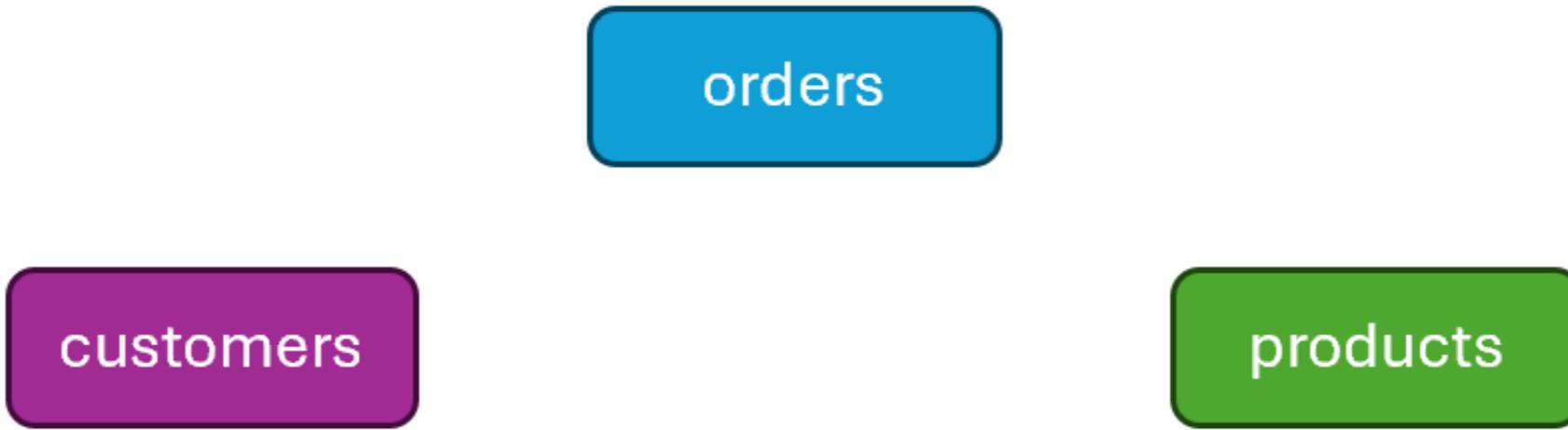


# What is a logical data model?



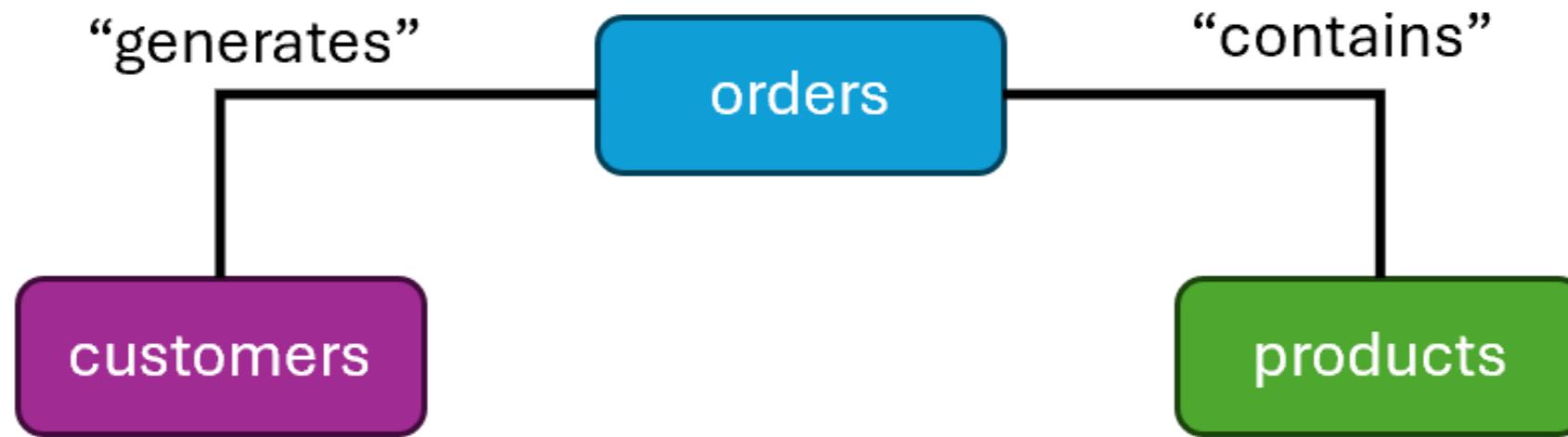
# Applying the conceptual model

Conceptual model: High-level overview of main data entities



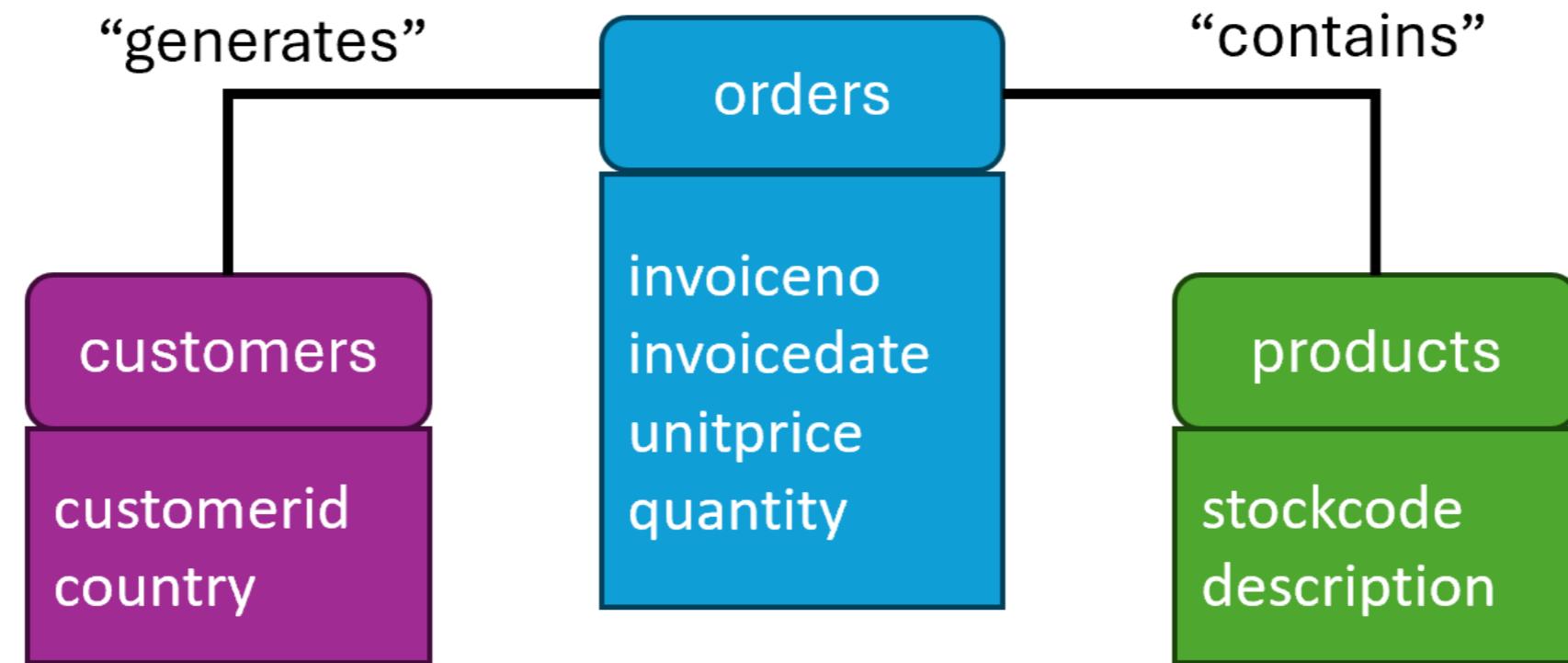
# Applying the conceptual model (1)

Conceptual model: High-level overview of main data entities



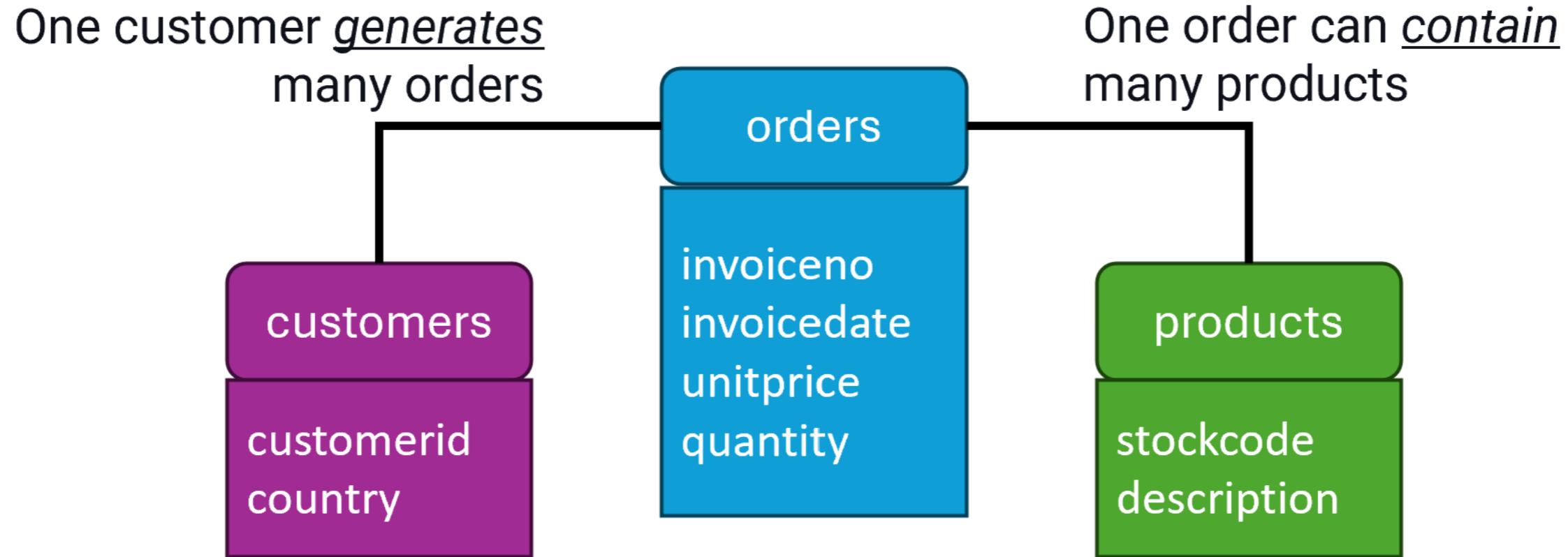
# Applying the logical model

Logical model: Detailed entities with attributes definition and their relationship



# Applying the logical model (1)

Relationship cardinality: Number of times entities are associated to each other



# Implementing the logical model

**SELECT FROM** : SQL command to fetch columns from a table

```
SELECT *  
FROM ecommerceonlineretail;
```

invoiceno	stockcode	description	quantity	invoicedate	unitprice	customerid	country
545218	22386	JUMBO BAG PINK POLKADOT	10	2/28/11 5:01 PM	1.95	17462	United Kingdom
545218	21933	PINK VINTAGE PAISLEY PICNIC BAG	5	2/28/11 5:01 PM	2.95	17462	United Kingdom
545214	22796	PHOTO FRAME 3 CLASSIC HANGING	2	2/28/11 4:39 PM	9.95	17165	United Kingdom
545214	22699	ROSES REGENCY TEACUP AND SAUCER	6	2/28/11 4:39 PM	2.95	17165	United Kingdom
545214	22151	PLACE SETTING WHITE HEART	24	2/28/11 4:39 PM	0.42	17165	United Kingdom
545213	22321	BIRD DECORATION RED RETROSPOT	12	2/28/11 4:36 PM	0.85	16518	United Kingdom
545213	22148	EASTER CRAFT 4 CHICKS	12	2/28/11 4:36 PM	1.95	16518	United Kingdom
545213	21880	RED RETROSPOT TAPE	12	2/28/11 4:36 PM	0.65	16518	United Kingdom
545191	20725	LUNCH BAG RED RETROSPOT	10	2/28/11 3:39 PM	1.65	12811	Portugal
545191	22662	LUNCH BAG DOLLY GIRL DESIGN	10	2/28/11 3:39 PM	1.65	12811	Portugal

# Implementing the logical model (1)

```
DESC TABLE ecommerceonlineretail;
```

<b>name</b>	<b>type</b>	<b>kind</b>	<b>null?</b>	<b>default</b>	<b>primary key</b>	<b>unique key</b>	<b>check expression</b>	<b>comment</b>	<b>policy name</b>
invoiceno	VARCHAR(10)	column	N	null	Y	N	null	null	null
stockcode	VARCHAR(255)	column	N	null	N	N	null	null	null
description	VARCHAR(255)	column	Y	null	N	N	null	null	null
quantity	NUMBER(38,0)	column	Y	0	N	N	null	null	null
invoicedate	TIMESTAMP_NTZ(9)	column	Y	null	N	N	null	null	null
unitprice	NUMBER(10,2)	column	Y	null	N	N	null	null	null
customerid	NUMBER(38,0)	column	N	0	N	N	null	null	null
country	VARCHAR(255)	column	Y	null	N	N	null	null	null

# Implementing the logical model (2)

**CREATE TABLE** : SQL command to define a new table structure

```
CREATE OR REPLACE TABLE customers (
    customerid NUMBER(38,0),
    country VARCHAR(255)
);
```

# Terminology and functions overview

- **Conceptual model:** High-level overview of main data entities
- **Logical model:** Detailed entities with attribute definitions and their relationship
- **Relationship cardinality:** Number of times entities are associated with each other
- **SELECT FROM :** SQL command to fetch columns from a table
- **CREATE OR REPLACE TABLE :** SQL command to create or replace a table structure

```
SELECT * FROM table_name;  
CREATE OR REPLACE TABLE table_name (  
    column_name column_datatype,  
    another_column column_datatype  
);
```

# **Let's practice!**

**INTRODUCTION TO DATA MODELING IN SNOWFLAKE**

# Exploring Physical Data Models

INTRODUCTION TO DATA MODELING IN SNOWFLAKE

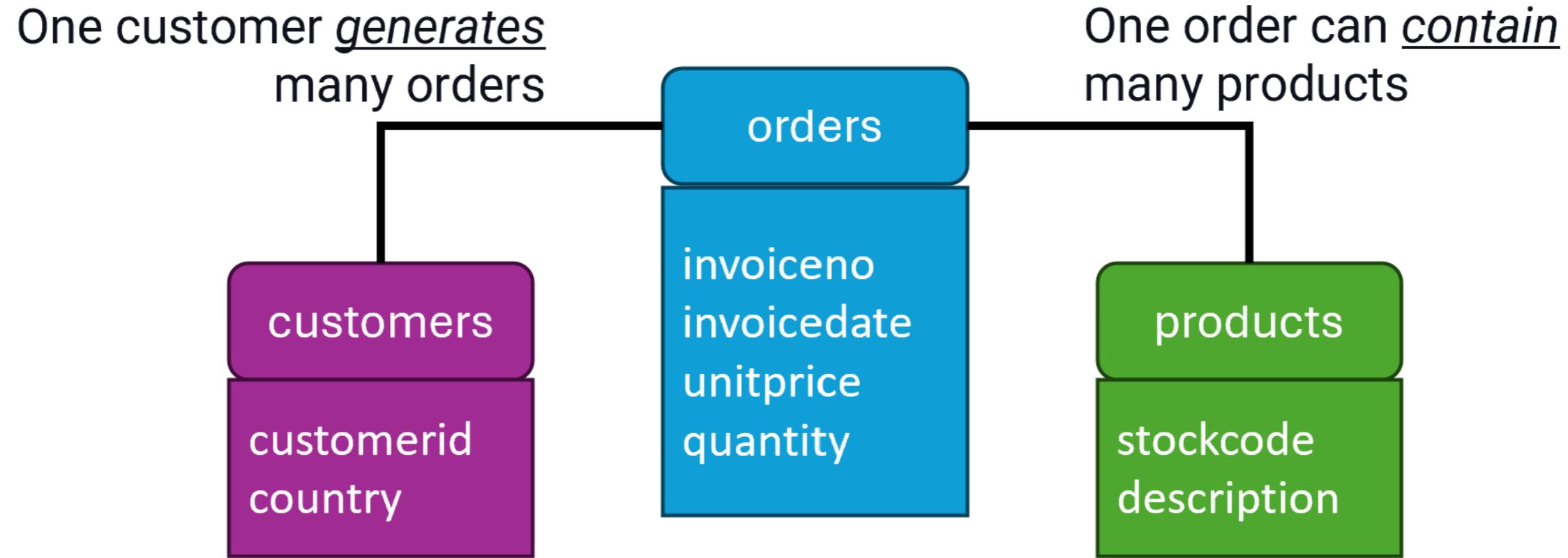


**Nuno Rocha**  
Director of Engineering

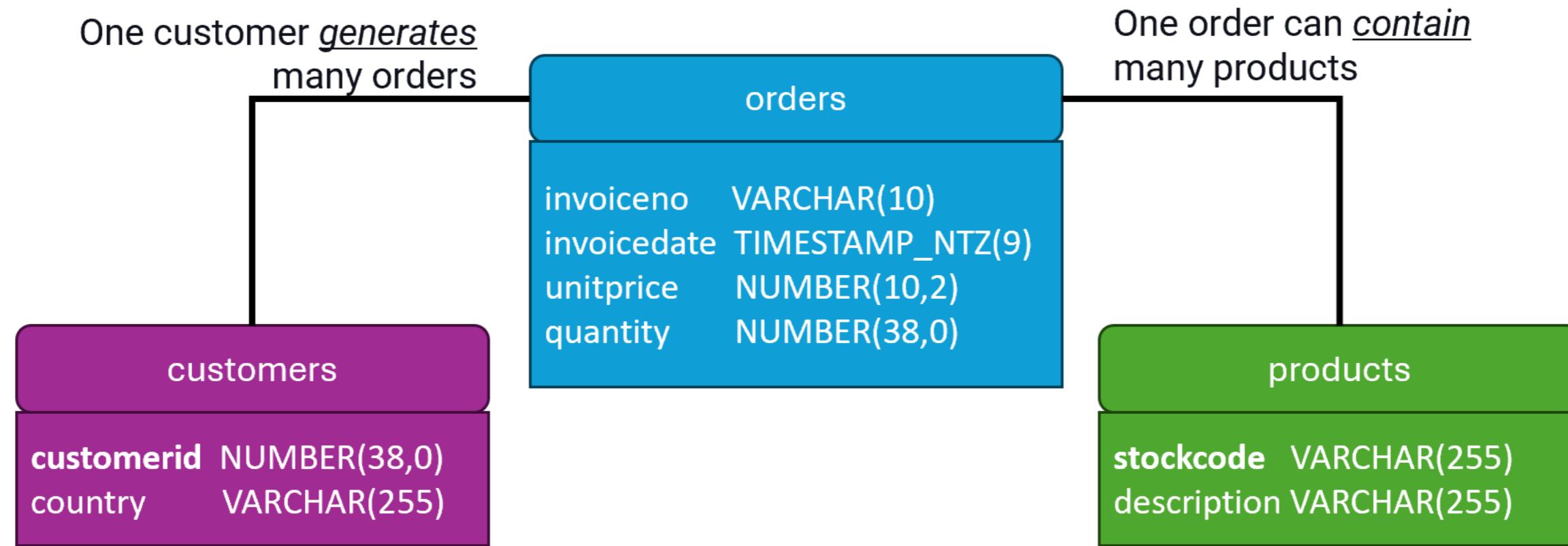
# The role of the physical model



# The physical model's details

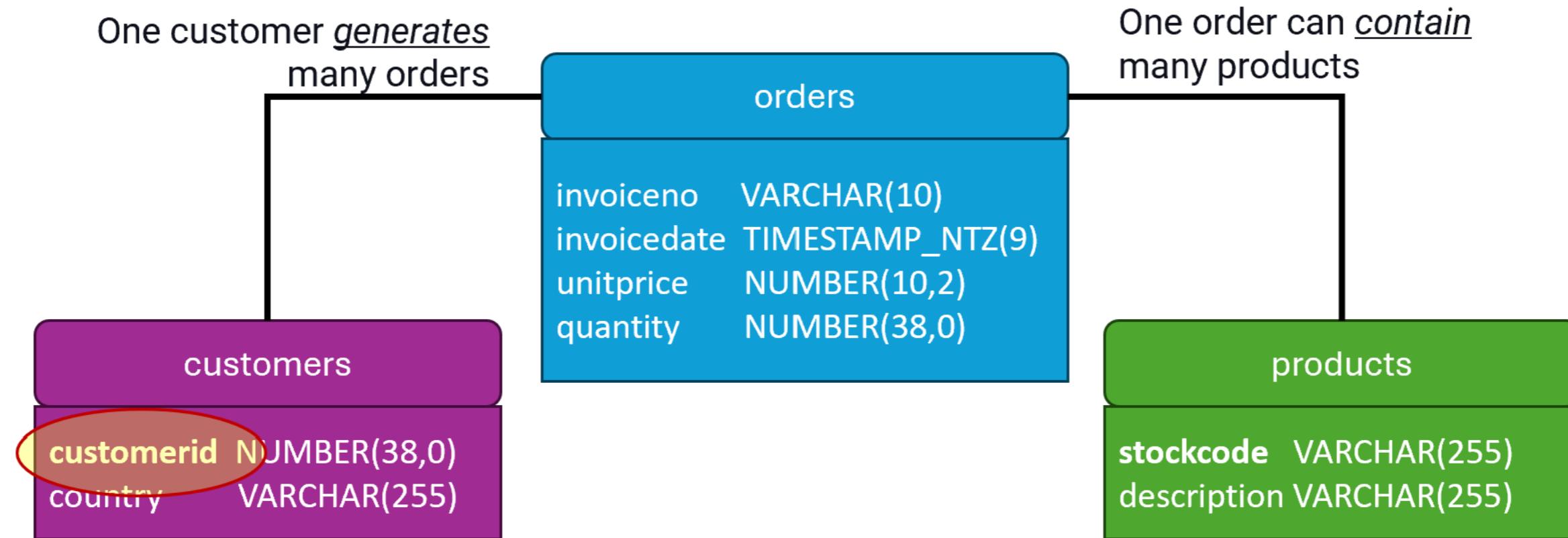


# The physical model's details (1)



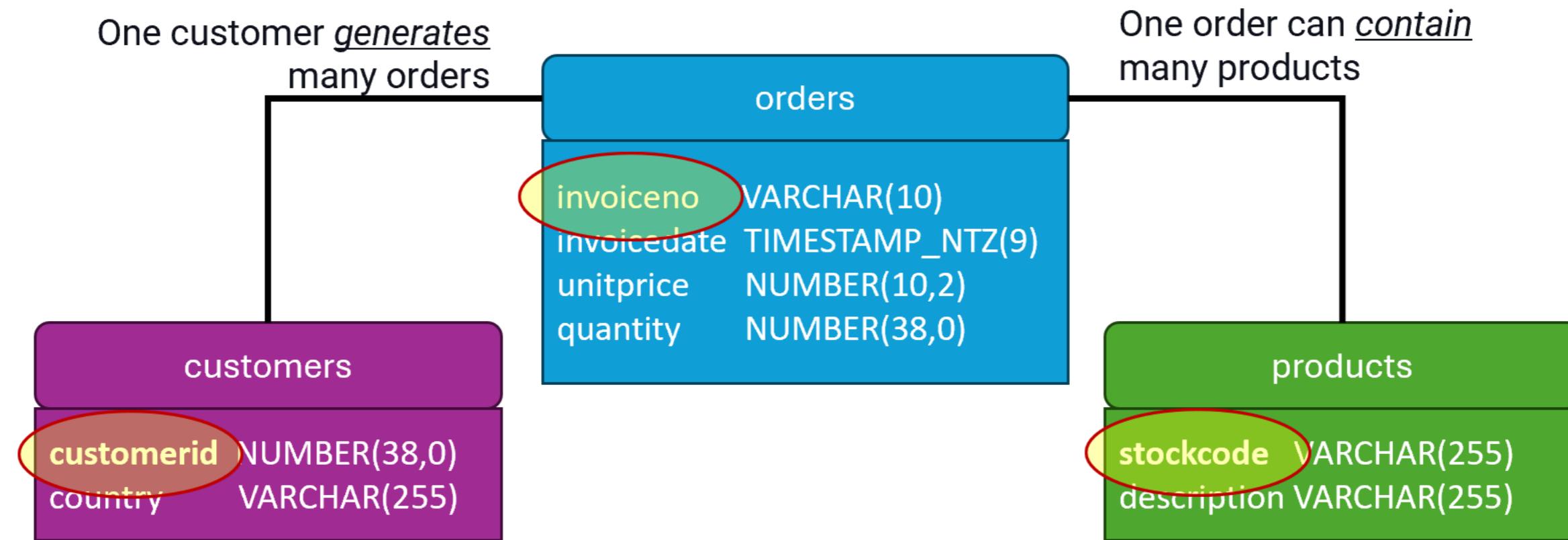
# Establishing primary keys

**Primary key (PK):** Ensures every record in a table has a unique identifier



# Establishing primary keys (1)

**Primary key (PK):** Ensures every record in a table has a unique identifier



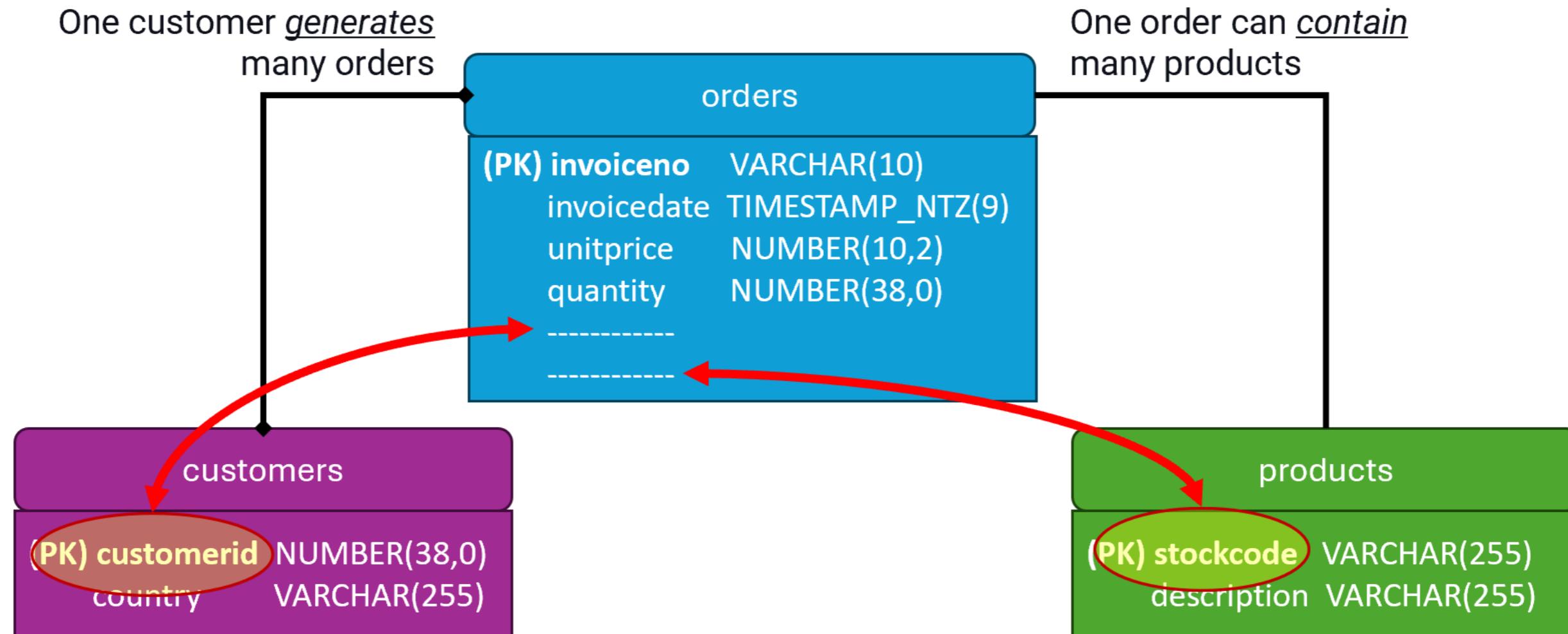
# Creating Primary Keys

**PRIMARY KEY** : SQL clause to define a column as the unique identifier

```
CREATE OR REPLACE TABLE products (
    stockcode VARCHAR(255) PRIMARY KEY,
    description VARCHAR(255)
);
```

# Establishing foreign keys

Foreign Key (FK): Connects records in different tables to keep the data related

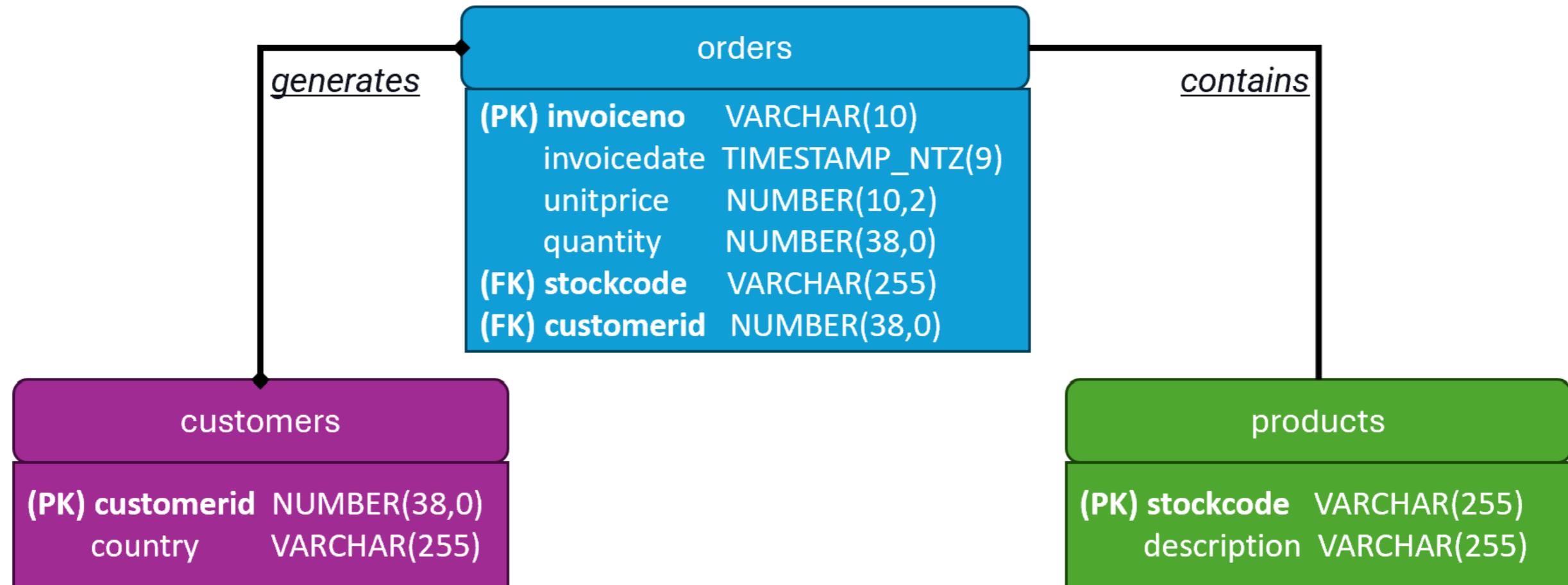


# Creating foreign keys

**FOREIGN KEY () REFERENCES ()** : SQL clause to define a column that references the primary key of another table

```
CREATE OR REPLACE TABLE orders (
    invoiceno INT,
    customerid INT,
    invoicedate DATE,
    unitprice DECIMAL(10, 2),
    quantity INT,
    stockcode VARCHAR(255),
    FOREIGN KEY (stockcode) REFERENCES products(stockcode)
);
```

# Finalizing the physical data model



# Terminology and functions overview

- **Physical data model:** Defines how data is stored and accessed, including table structures, data types, and primary and foreign keys
- **PRIMARY KEY :** SQL clause to define a column as the unique identifier
- **FOREIGN KEY (...) REFERENCES (...):** SQL clause to create a link between two tables

```
CREATE OR REPLACE TABLE table_name (
    unique_column column_datatype PRIMARY KEY,
    other_columns column_datatype,
    foreign_column column_datatype,
    FOREIGN KEY (foreign_column) REFERENCES foreign_table(PK_from_foreign_table)
);
```

# **Let's practice!**

**INTRODUCTION TO DATA MODELING IN SNOWFLAKE**