
MySQL Constraints
365 DataScience

PRIMARY KEY Constraint

365 DataScience

PRIMARY KEY Constraint

So far:

PRIMARY KEY Constraint

So far:

- Theory of Relational Databases

PRIMARY KEY Constraint

So far:

- Theory of Relational Databases
- SQL Theory

PRIMARY KEY Constraint

So far:

- Theory of Relational Databases
- SQL Theory
- Creating a Database

PRIMARY KEY Constraint

So far:

- Theory of Relational Databases
- SQL Theory
- Creating a Database

In this section:

PRIMARY KEY Constraint

So far:

- Theory of Relational Databases
- SQL Theory
- Creating a Database

In this section:

- Constraints

PRIMARY KEY Constraint

So far:

- Theory of Relational Databases
- SQL Theory
- Creating a Database

In this section:

- Constraints

In this lesson:

PRIMARY KEY Constraint

So far:

- Theory of Relational Databases
- SQL Theory
- Creating a Database

In this section:

- Constraints

In this lesson:

- the PRIMARY KEY Constraint

PRIMARY KEY Constraint

constraints

specific rules, or limits, that we define in our tables

PRIMARY KEY Constraint

constraints

specific rules, or limits, that we define in our tables

- the role of constraints is to outline the existing relationships between different tables in our database

PRIMARY KEY Constraint

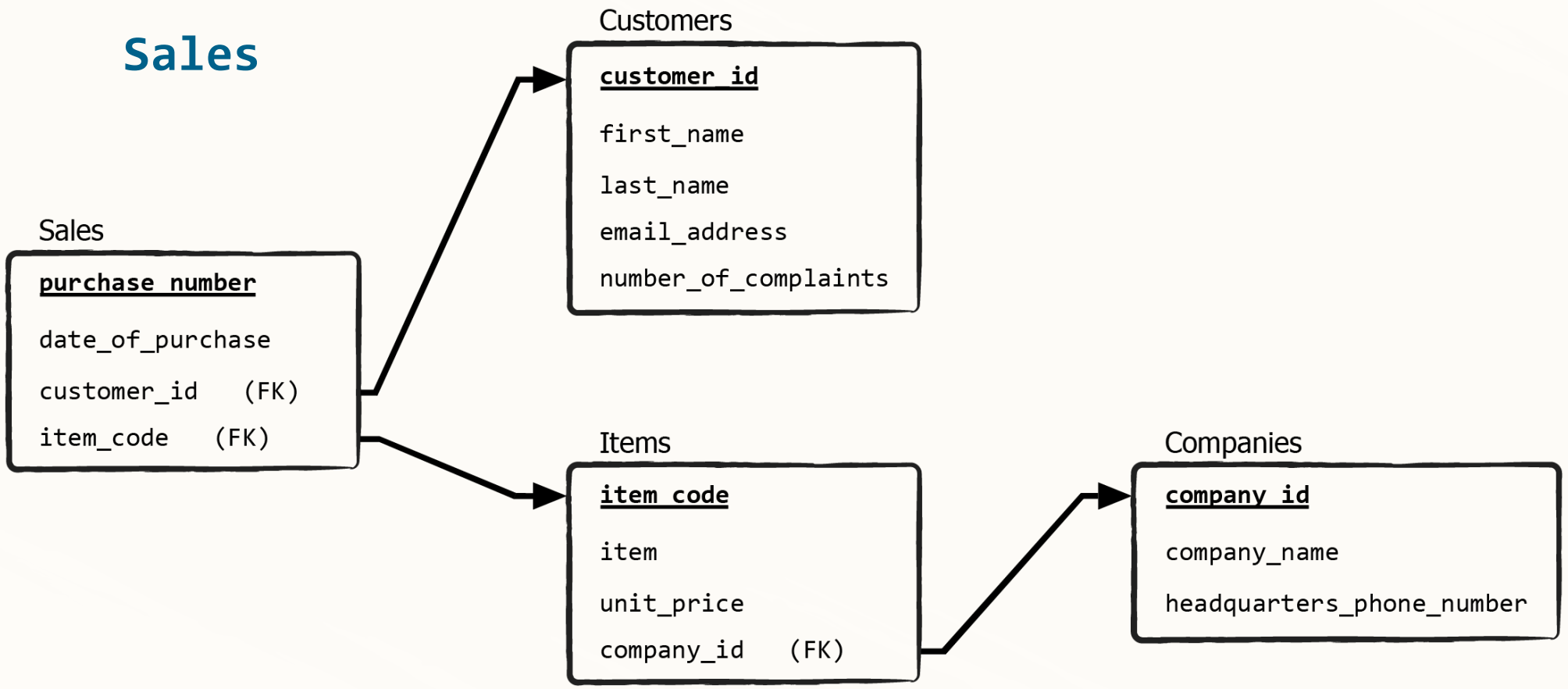
constraints

specific rules, or limits, that we define in our tables

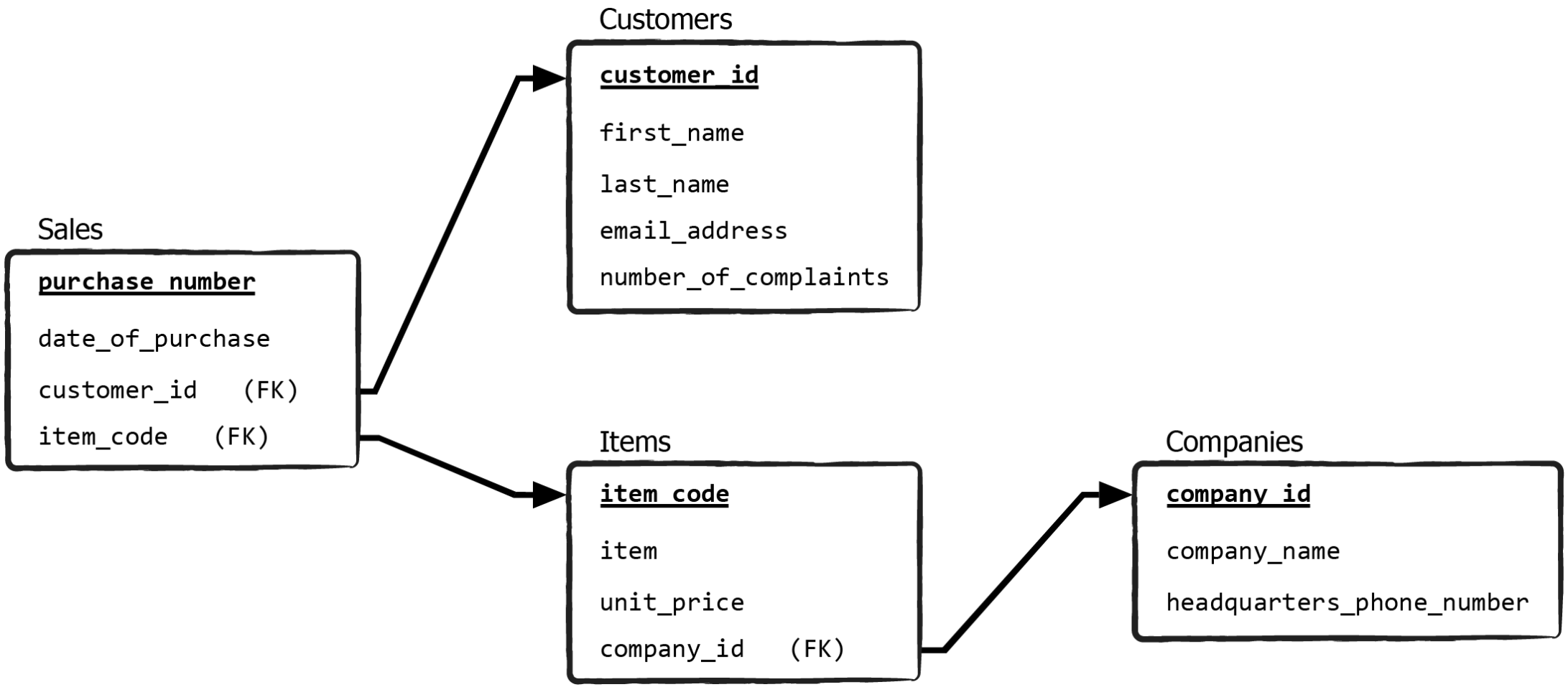
- the role of constraints is to outline the existing relationships between different tables in our database

e.g. `NOT NULL`

PRIMARY KEY Constraint



Database: sales



FOREIGN KEY Constraint

365 DataScience

FOREIGN KEY Constraint

Sales

Sales

purchase_number
date_of_purchase
customer_id (FK)
item_code (FK)

Customers

customer_id
first_name
last_name
email_address
number_of_complaints

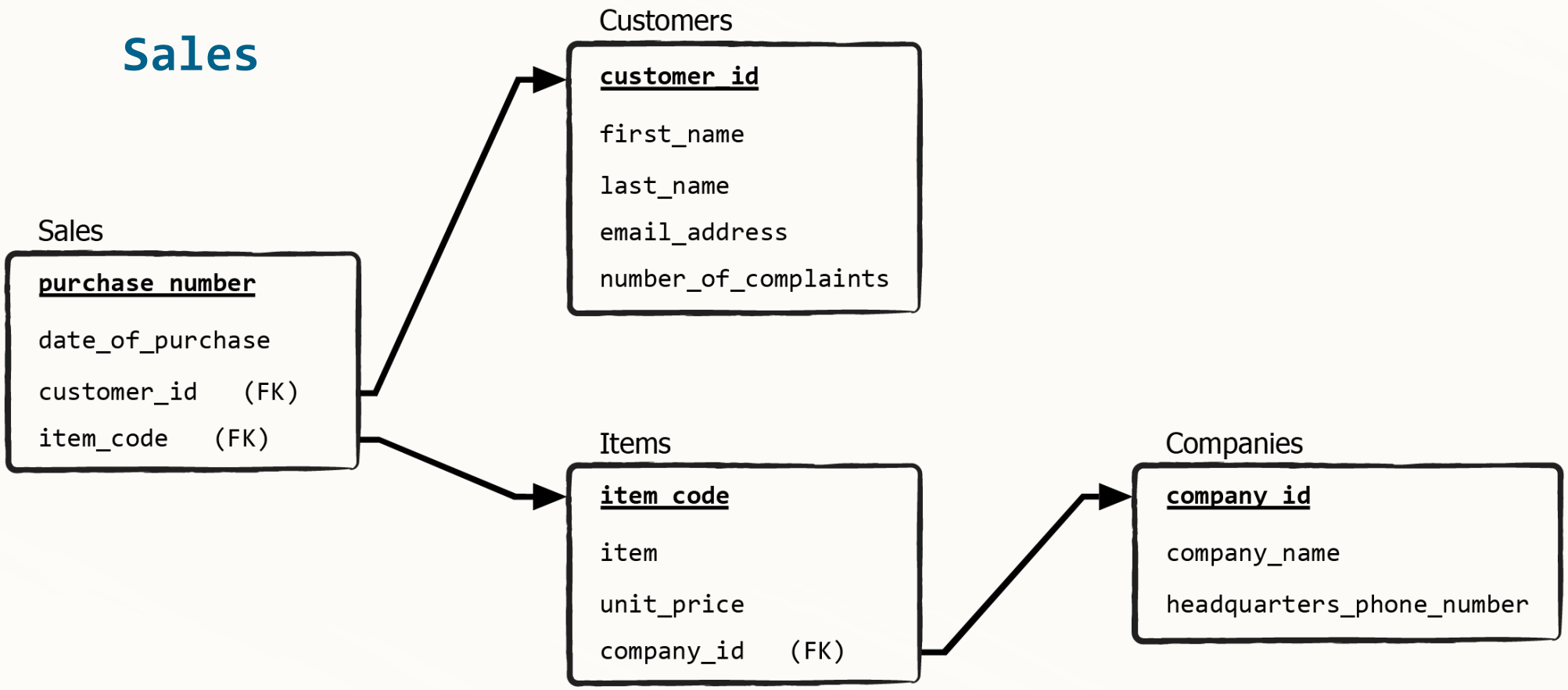
Items

item_code
item
unit_price
company_id (FK)

Companies

company_id
company_name
headquarters_phone_number

FOREIGN KEY Constraint

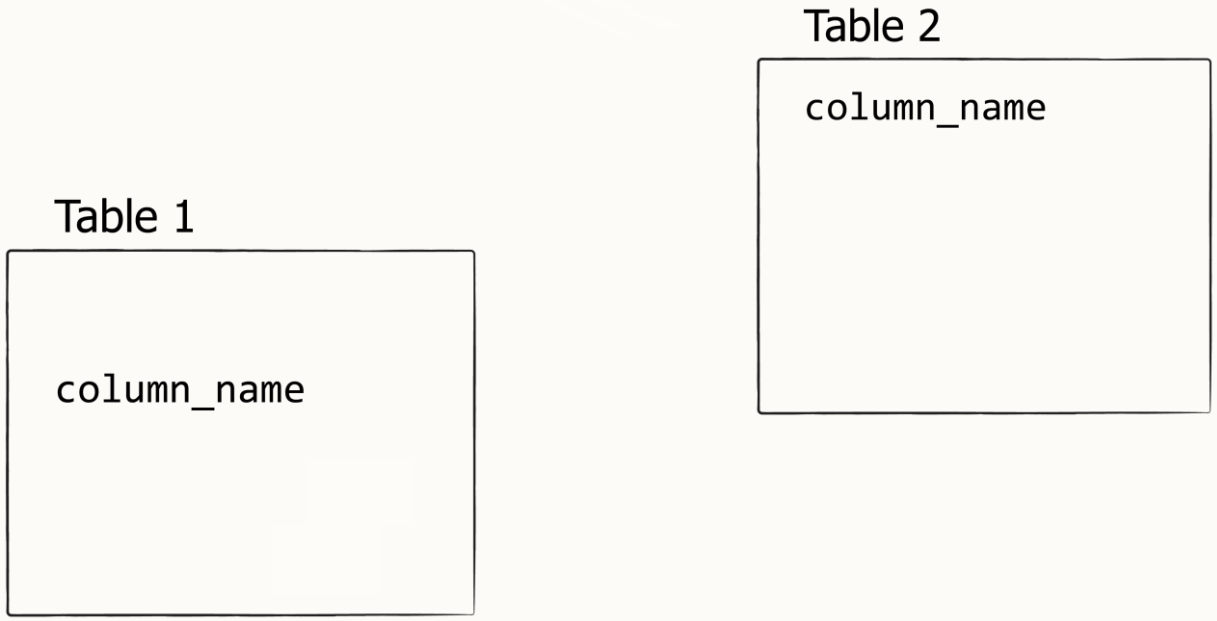


FOREIGN KEY Constraint

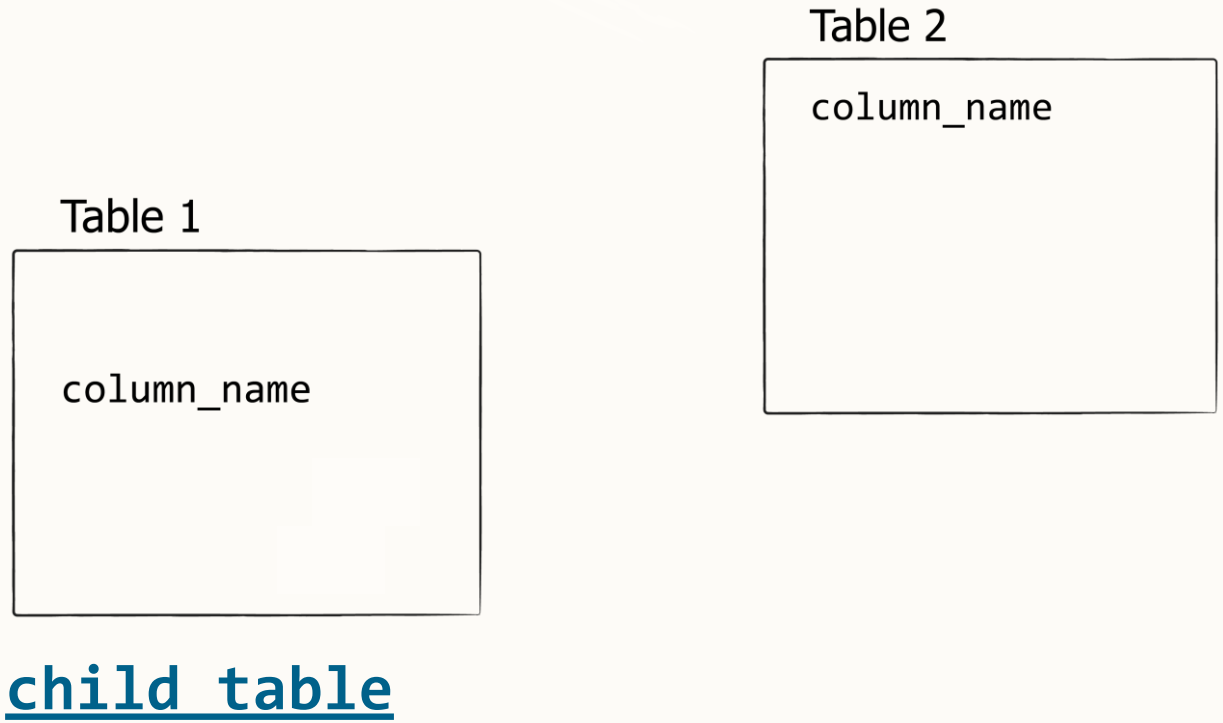
`foreign key`

points to a column of another table and, thus, links the two tables

FOREIGN KEY Constraint

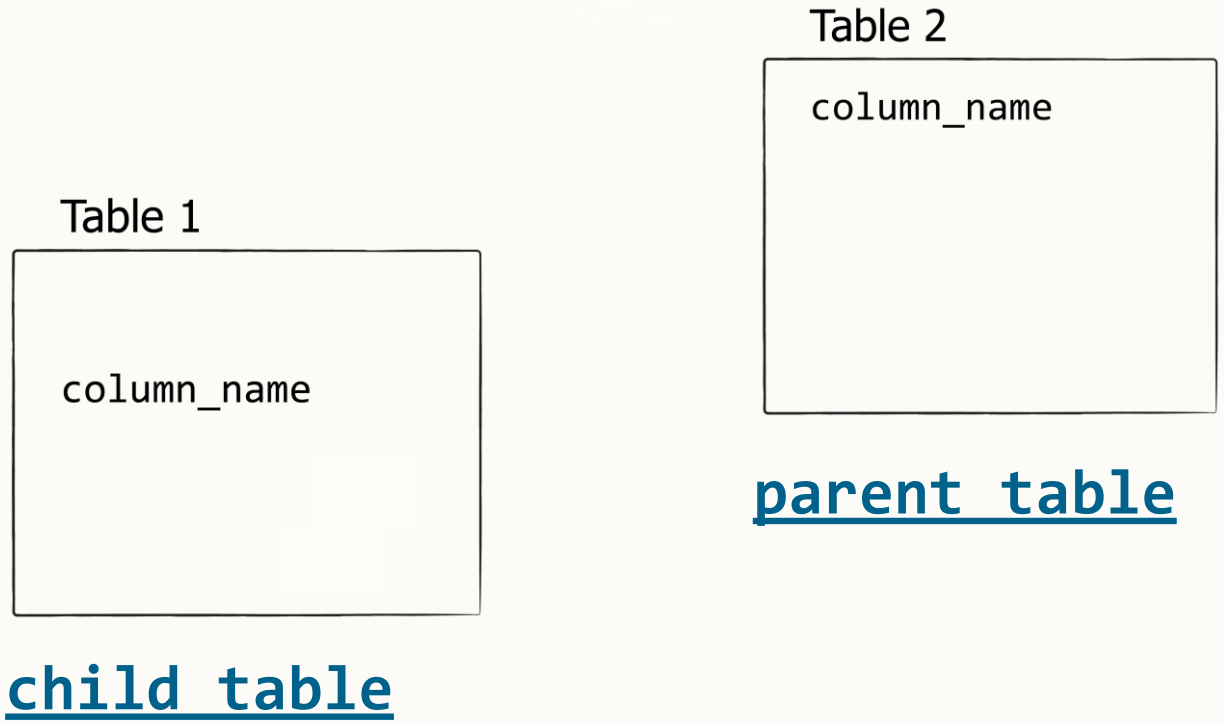


FOREIGN KEY Constraint

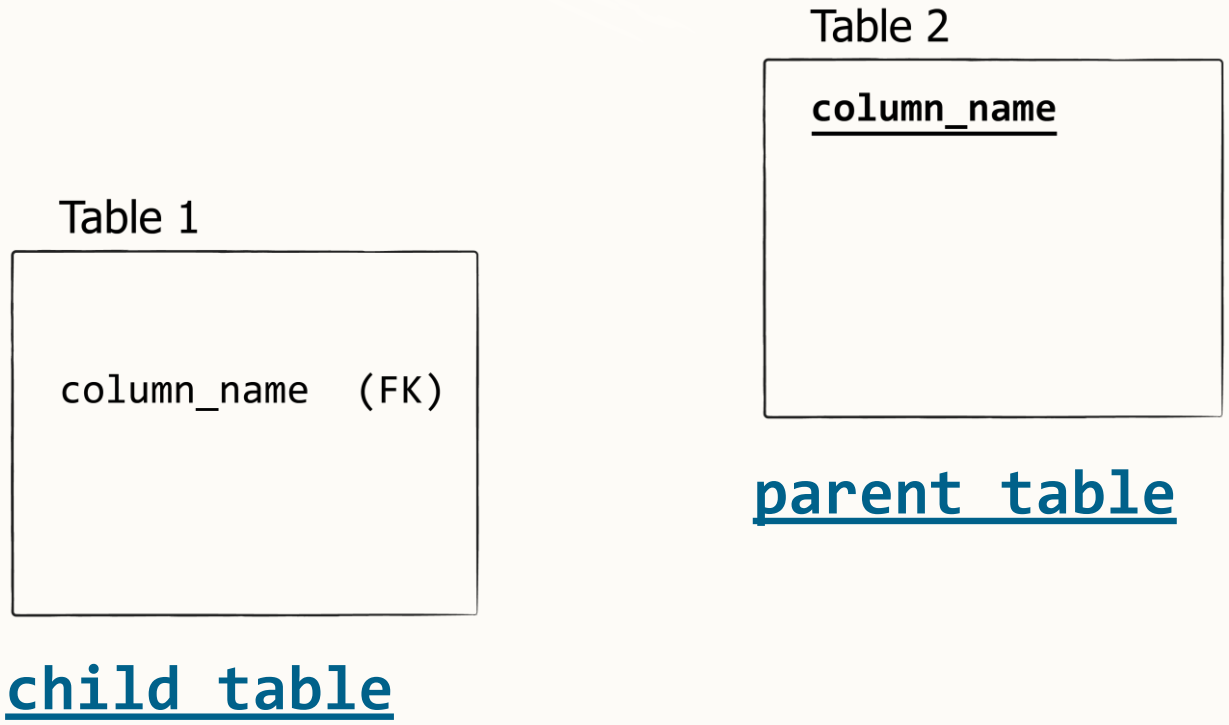


child table

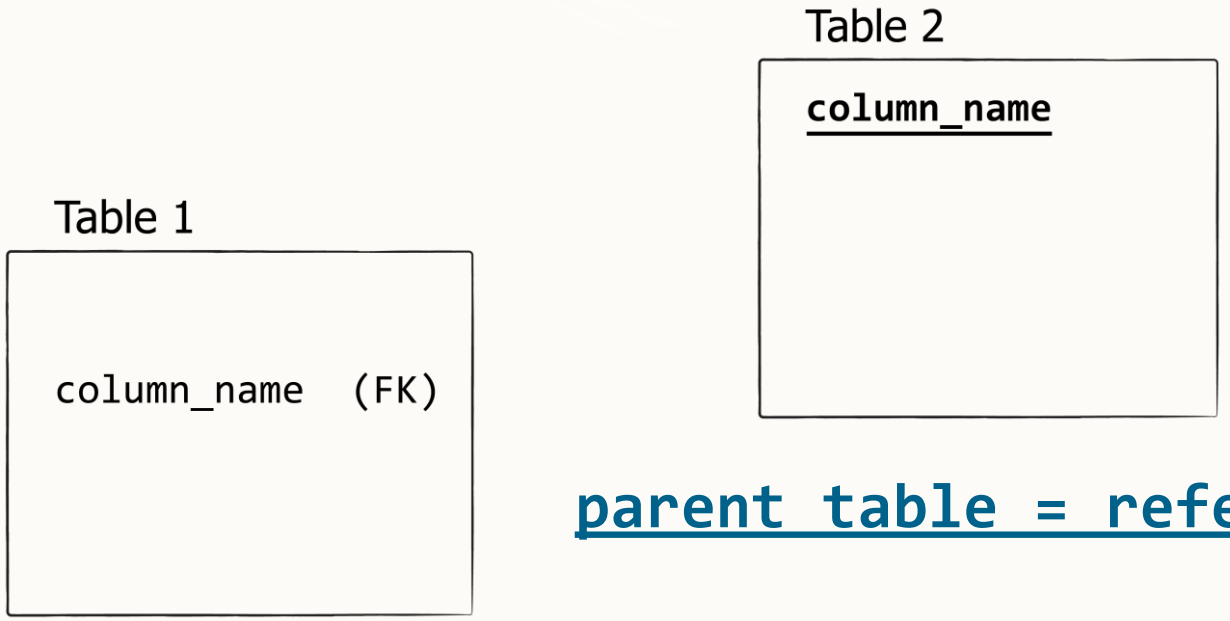
FOREIGN KEY Constraint



FOREIGN KEY Constraint



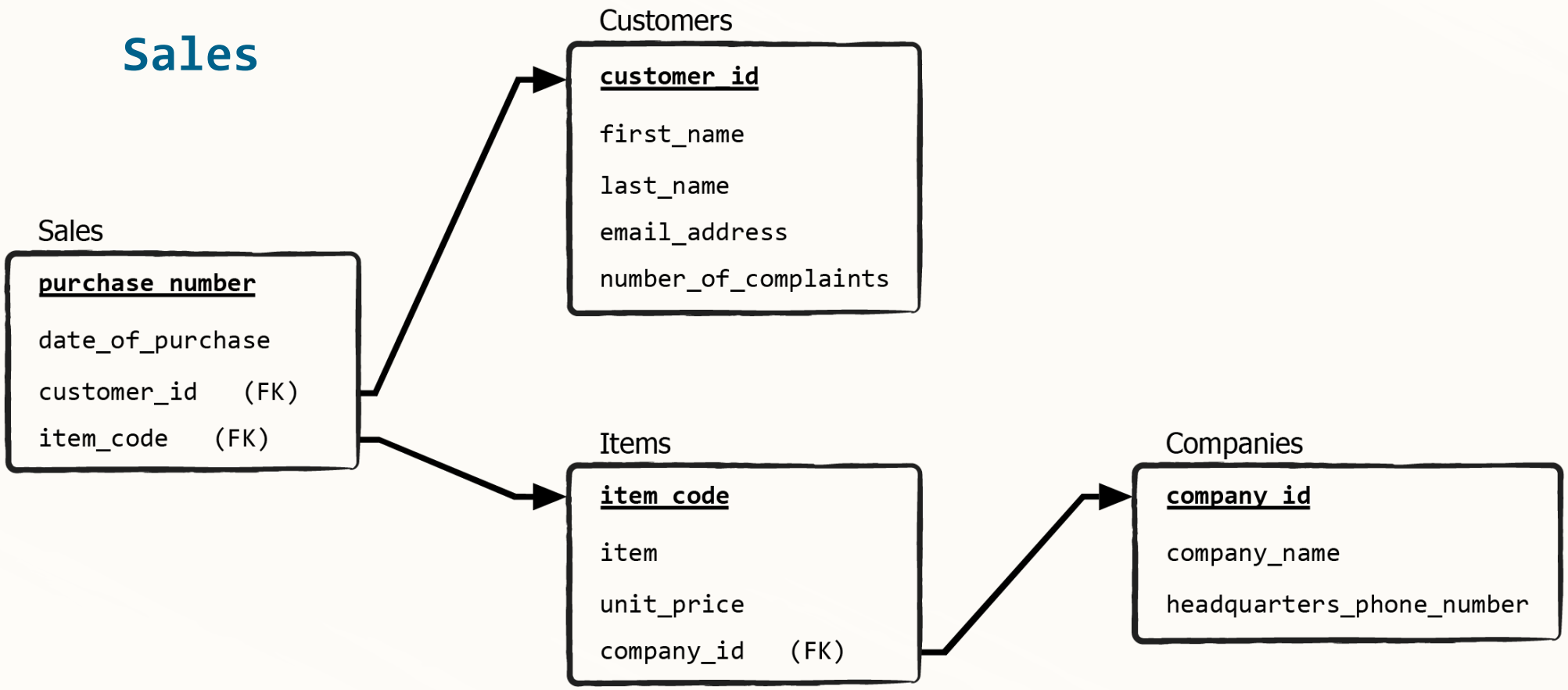
FOREIGN KEY Constraint



parent table = referenced table

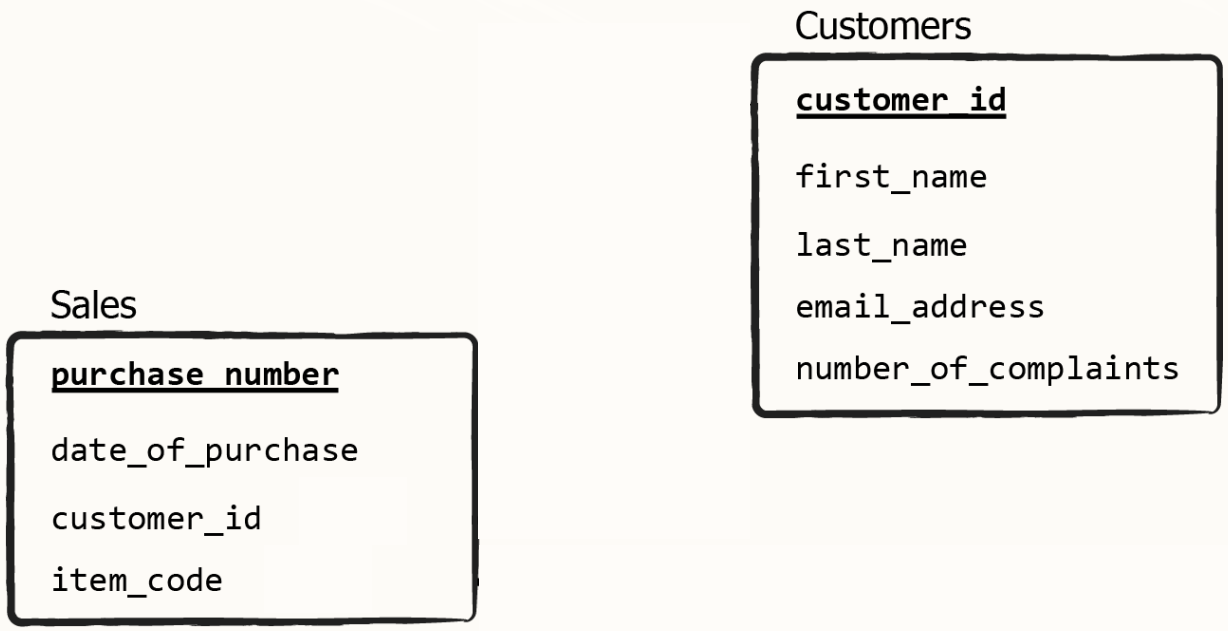
child table = referencing table

FOREIGN KEY Constraint



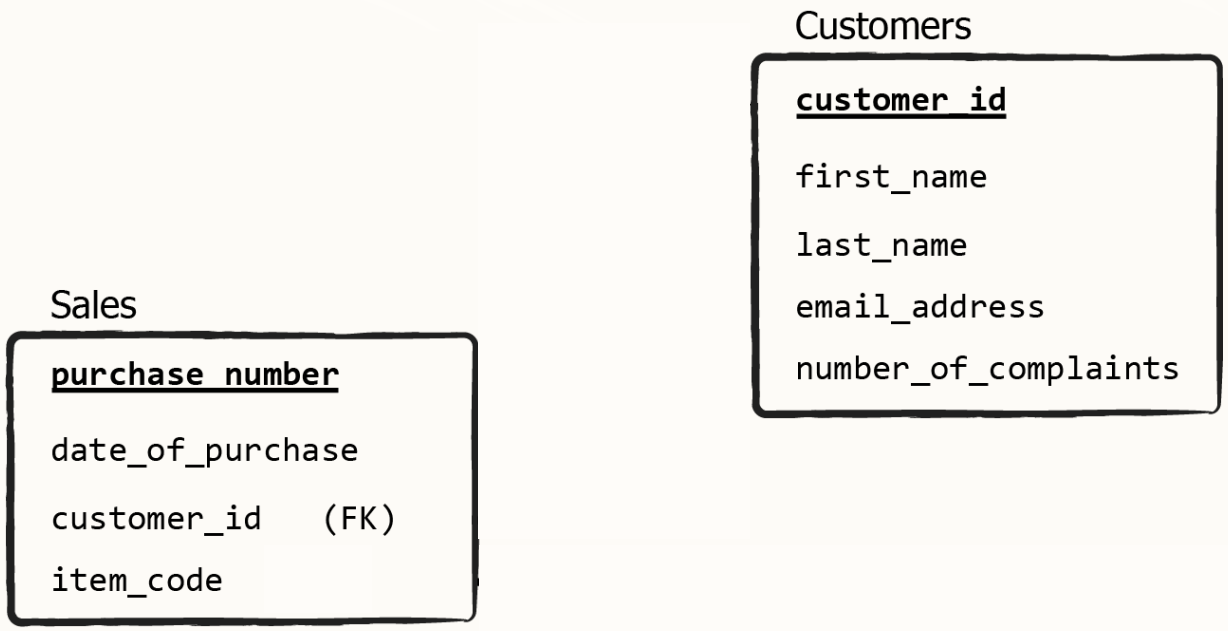
FOREIGN KEY Constraint

Sales



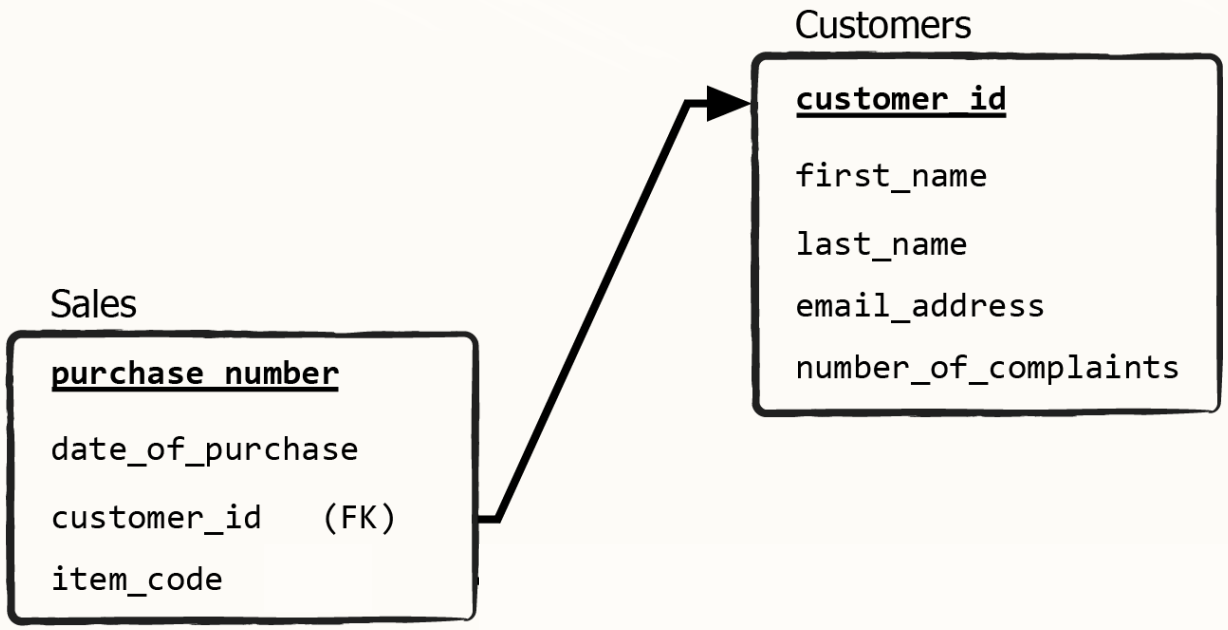
FOREIGN KEY Constraint

Sales



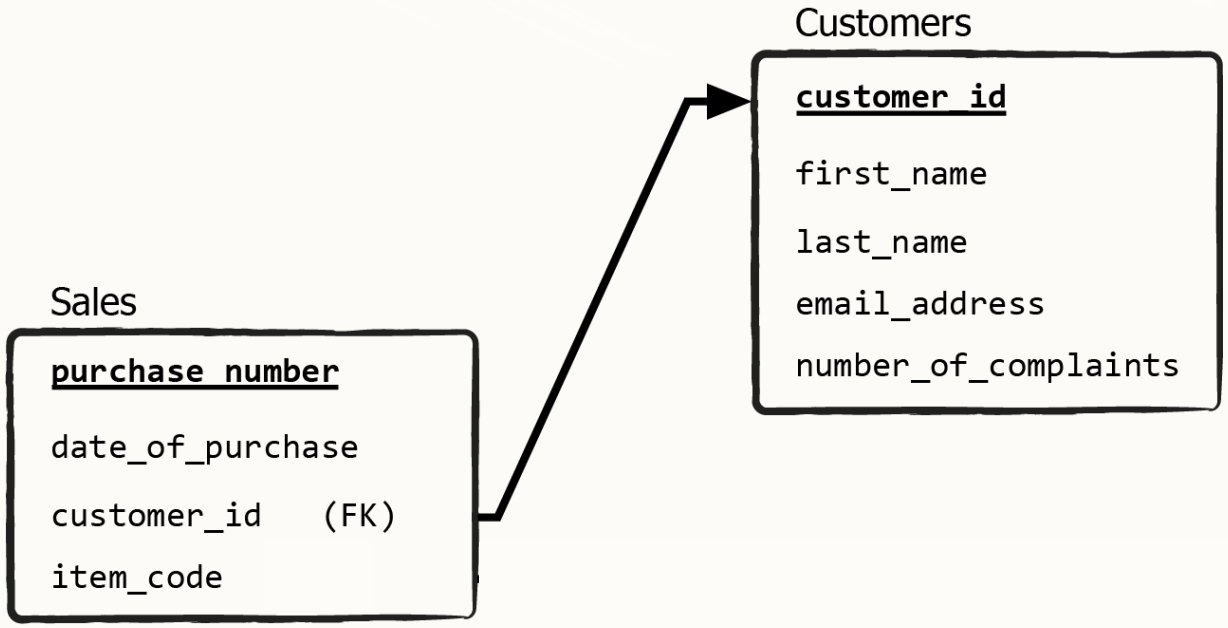
FOREIGN KEY Constraint

Sales



FOREIGN KEY Constraint

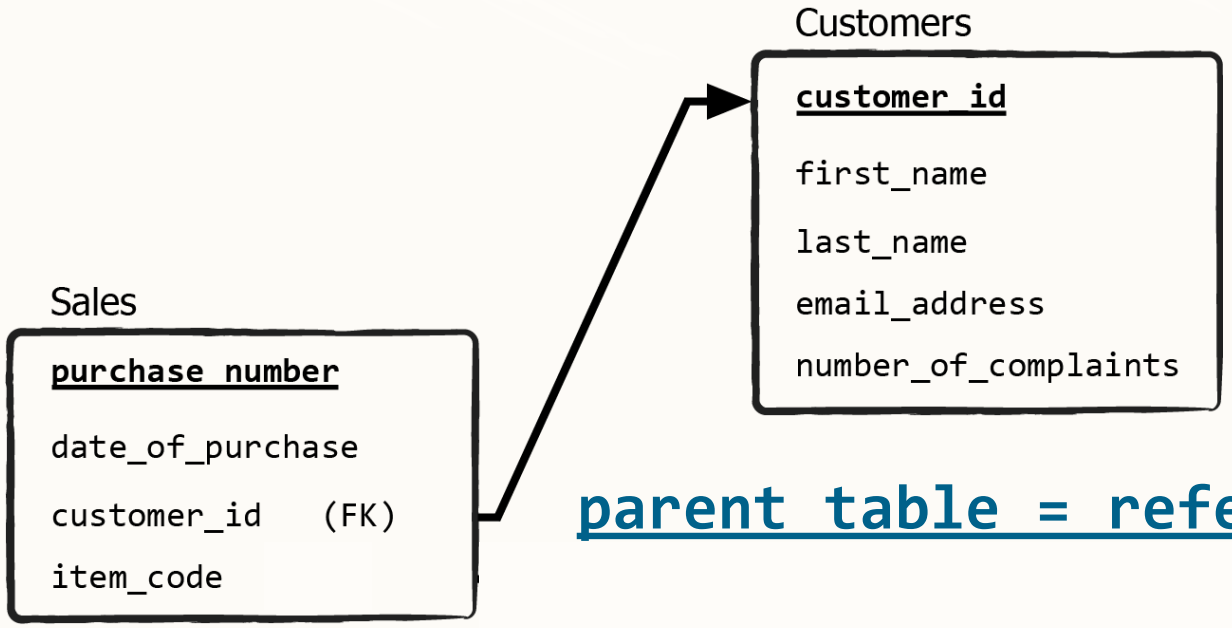
Sales



Remember, this is not an obligatory requirement – these two keys may have two completely different names. What's important is that the data types and the information match! It's just common practice to use, if not the same, then similar names for both keys.

FOREIGN KEY Constraint

Sales



parent table = referenced table

child table = referencing table

FOREIGN KEY Constraint

- a *foreign key* in SQL is defined through a foreign key constraint

FOREIGN KEY Constraint

- a *foreign key* in SQL is defined through a foreign key constraint

  the foreign key maintains the *referential integrity* within the database

FOREIGN KEY Constraint

ON DELETE CASCADE

if a specific value *from the parent table's primary key* has been deleted, all the records *from the child table* referring to this value will be removed as well

FOREIGN KEY Constraint

Customers				
customer_id	first_name	last_name	email_address	number_of_complaints
1	John	McKinley	john.mackinley@365careers.com	0
2	Elizabeth	McFarlane	e.mcfarlane@365careers.com	2
3	Kevin	Lawrence	kevin.lawrence@365careers.com	1
4	Catherine	Winnfield	c.winnfield@365careers.com	0

Sales			
purchase_number	date_of_purchase	customer_id	item_code
1	9/3/2016	1	A_1
2	12/2/2016	2	C_1
3	4/15/2017	3	D_1
4	5/24/2017	1	B_2
5	5/25/2017	4	B_2
6	6/6/2017	2	B_1
7	6/10/2017	4	A_2
8	6/13/2017	3	C_1
9	7/20/2017	1	A_1
10	8/11/2017	2	B_1

FOREIGN KEY Constraint

Customers				
customer_id	first_name	last_name	email_address	number_of_complaints
1	John	McKinley	john.mackinley@365careers.com	0
2	Elizabeth	McFarlane	e.mcfarlane@365careers.com	2
3	Kevin	Lawrence	kevin.lawrence@365careers.com	1
4	Catherine	Winnfield	c.winnfield@365careers.com	0

Sales			
purchase_number	date_of_purchase	customer_id	item_code
1	9/3/2016	1	A_1
2	12/2/2016	2	C_1
3	4/15/2017	3	D_1
4	5/24/2017	1	B_2
5	5/25/2017		B_2
6	6/6/2017	2	B_1
7	6/10/2017	4	A_2
8	6/13/2017	3	C_1
9	7/20/2017	1	A_1
10	8/11/2017	2	B_1

FOREIGN KEY Constraint

Customers

customer_id	first_name	last_name	email_address	number_of_complaints
1	John	McKinley	john.mackinley@365careers.com	0
2	Elizabeth	McFarlane	e.mcfarlane@365careers.com	2
3	Kevin	Lawrence	kevin.lawrence@365careers.com	1

Sales

purchase_number	date_of_purchase	customer_id	item_code
1	9/3/2016	1	A_1
2	12/2/2016	2	C_1
3	4/15/2017	3	D_1
4	5/24/2017	1	B_2
5	5/25/2017	4	B_2
6	6/6/2017	2	B_1
7	6/10/2017	4	A_2
8	6/13/2017	3	C_1
9	7/20/2017	1	A_1
10	8/11/2017	2	B_1

FOREIGN KEY Constraint

Customers				
customer_id	first_name	last_name	email_address	number_of_complaints
1	John	McKinley	john.mackinley@365careers.com	0
2	Elizabeth	McFarlane	e.mcfarlane@365careers.com	2
3	Kevin	Lawrence	kevin.lawrence@365careers.com	1

ON DELETE CASCADE

Sales			
purchase_number	date_of_purchase	customer_id	item_code
1	9/3/2016	1	A_1
2	12/2/2016	2	C_1
3	4/15/2017	3	D_1
4	5/24/2017	1	B_2
6	6/6/2017	2	B_1
8	6/10/2017	3	C_1
9	7/20/2017	1	A_1
10	8/11/2017	2	B_1

UNIQUE Constraint

UNIQUE Constraint

`unique key`

used whenever you would like to specify that you don’t want to see duplicate data in a given field

UNIQUE Constraint

unique key

used whenever you would like to specify that you don't want to see duplicate data in a given field

- ensures that all values in a column (or a set of columns) are different

UNIQUE Constraint

- *unique keys* are implemented in SQL through a constraint – the UNIQUE Constraint

UNIQUE Constraint

- *unique keys* are implemented in SQL through a constraint – the UNIQUE Constraint

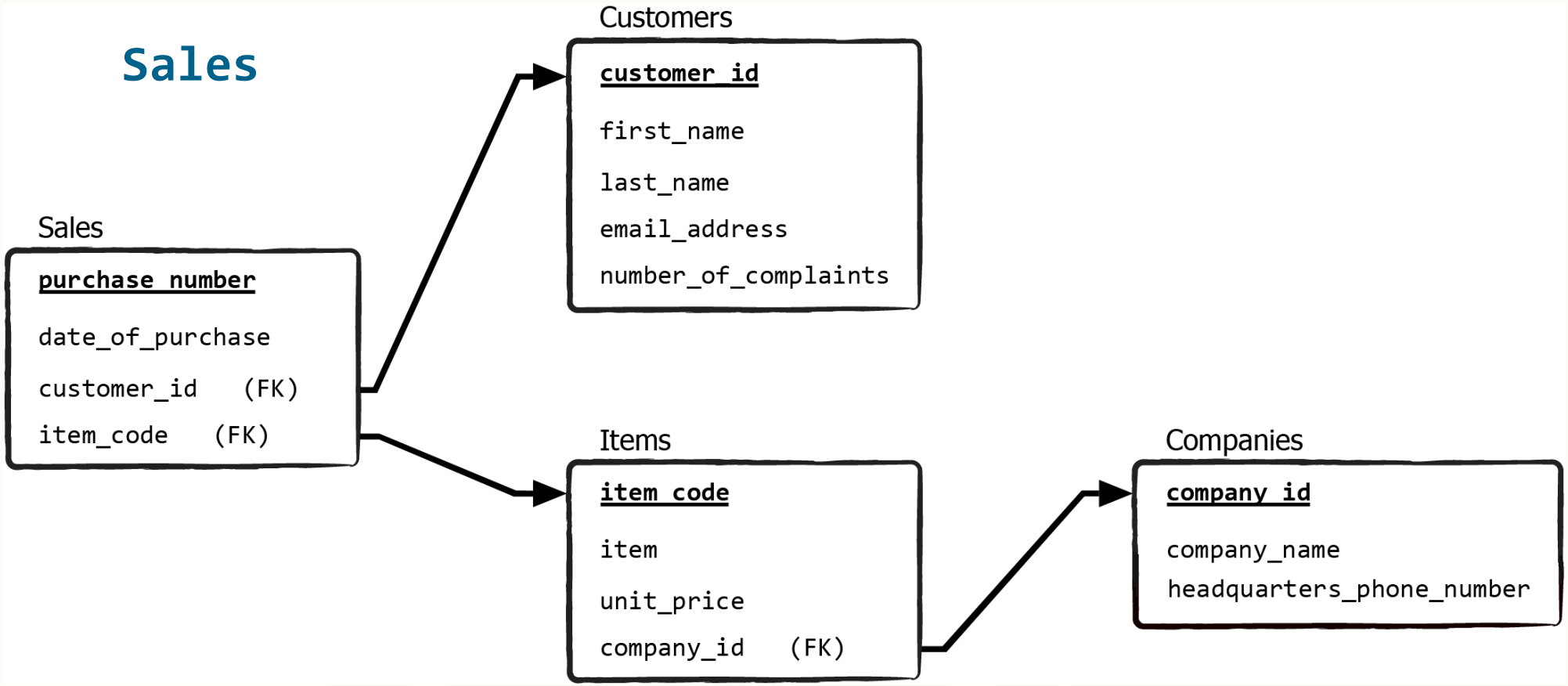
if you attempt to insert an already existing, duplicate value in the unique column, SQL will display an error

UNIQUE Constraint

- *unique keys* are implemented in SQL through a constraint – the UNIQUE Constraint

if you attempt to insert an already existing, duplicate value in the unique column, SQL will display an error

UNIQUE Constraint



UNIQUE Constraint

Customers				
customer_id	first_name	last_name	email_address	number_of_complaints
1	John	McKinley	john.mackinley@365careers.com	0
2	Elizabeth	McFarlane	e.mcfarlane@365careers.com	2
3	Kevin	Lawrence	kevin.lawrence@365careers.com	1
4	Catherine	Winnfield	c.winnfield@365careers.com	0

UNIQUE Constraint

- Indexes

UNIQUE Constraint

Indexes

unique keys in MySQL have the same role as *indexes*

UNIQUE Constraint

Indexes

unique keys in MySQL have the same role as *indexes*

the reverse isn’t true !!!

UNIQUE Constraint

Indexes

unique keys in MySQL have the same role as *indexes*

the reverse isn't true !!!

index of a table

an organizational unit that helps retrieve data more easily

UNIQUE Constraint

Indexes

unique keys in MySQL have the same role as *indexes*

the reverse isn't true !!!

index of a table

an organizational unit that helps retrieve data more easily

- it takes more time to update a table because indexes must be updated, too, and that's time consuming

UNIQUE Constraint

Customers				
customer_id	first_name	last_name	email_address	number_of_complaints
1	John	McKinley	john.mackinley@365careers.com	0
2	Elizabeth	McFarlane	e.mcfarlane@365careers.com	2
3	Kevin	Lawrence	kevin.lawrence@365careers.com	1
4	Catherine	Winnfield	c.winnfield@365careers.com	0

UNIQUE Constraint

SQL

```
ALTER TABLE table_name
DROP INDEX unique_key_field;
```

DEFAULT Constraint

365 DataScience

DEFAULT Constraint

- DEFAULT Constraint

DEFAULT Constraint

DEFAULT Constraint

helps us assign a particular default value to every row of a column

DEFAULT Constraint

DEFAULT Constraint

helps us assign a particular default value to every row of a column

- a value different from the default can be stored in a field where the DEFAULT constraint has been applied, *only if specifically indicated*.

DEFAULT Constraint

Customers				
customer_id	first_name	last_name	email_address	number_of_complaints
1	John	McKinley	john.mackinley@365careers.com	0
2	Elizabeth	McFarlane	e.mcfarlane@365careers.com	2
3	Kevin	Lawrence	kevin.lawrence@365careers.com	1
4	Catherine	Winnfield	c.winnfield@365careers.com	0

DEFAULT Constraint


```sql
CREATE TABLE customers
(
    customer_id INT,
    first_name VARCHAR(255),
    last_name VARCHAR(255),
    email_address VARCHAR(255),
    number_of_complaints INT DEFAULT 0,
PRIMARY KEY (customer_id)
);
```

DEFAULT Constraint

Customers					
customer_id	first_name	last_name	gender	email_address	number_of_complaints
1	John	McKinley	M	john.mackinley@365careers.com	0

DEFAULT Constraint

Customers					
customer_id	first_name	last_name	gender	email_address	number_of_complaints
1	John	McKinley	M	john.mackinley@365careers.com	0

2

DEFAULT Constraint

Customers					
customer_id	first_name	last_name	gender	email_address	number_of_complaints
1	John	McKinley	M	john.mackinley@365careers.com	0
2	Peter	Figaro	M		

Data Definition Language

Data Definition Language (DDL)

- CREATE
- ALTER
- DROP

NOT NULL Constraint

365 DataScience

NOT NULL Constraint

	primary key	unique key
NULL VALUES	no	yes

NOT NULL Constraint

	primary key	unique key
NULL VALUES	no	yes

the “not null” restriction is applied through the NOT NULL Constraint

NOT NULL Constraint

	primary key	unique key
NULL VALUES	no	yes

the “not null” restriction is applied through the NOT NULL Constraint

- when you insert values in the table, you cannot leave the respective field *empty*

NOT NULL Constraint

	primary key	unique key
NULL VALUES	no	yes

the “not null” restriction is applied through the NOT NULL Constraint

- when you insert values in the table, you cannot leave the respective field *empty*

- if you leave it *empty*, MySQL will signal an error

NOT NULL Constraint

Companies		
company_id	headquarters_phone_number	company_name
1	+1 (202) 555-0196	Company A
2	+1 (202) 555-0152	Company B
3	+1 (229) 853-9913	Company C
4	+1 (618) 369-7392	Company D

NOT NULL Constraint

Companies		
company_id	headquarters_phone_number	company_name
1	+1 (202) 555-0196	Company A
2	+1 (202) 555-0152	Company B
3	+1 (229) 853-9913	Company C
4	+1 (618) 369-7392	Company D

NOT NULL

NOT NULL Constraint


```sql
CREATE TABLE companies
(
    company_id INT AUTO_INCREMENT,
    headquarters_phone_number VARCHAR(255),
    company_name VARCHAR(255),
PRIMARY KEY (company_id)
);
```

NOT NULL Constraint


```sql
CREATE TABLE companies
(
    company_id INT AUTO_INCREMENT,
    headquarters_phone_number VARCHAR(255),
    company_name VARCHAR(255) NOT NULL,
PRIMARY KEY (company_id)
);
```

NOT NULL Constraint

- Don’t confuse a NULL value with the value of 0 or with a “NONE” response!

Think of a null value as a missing value.

0 NONE	NULL
assigned by the user	assigned by the computer

NOT NULL Constraint

Customers					
customer_id	first_name	last_name	gender	email_address	number_of_complaints
1	John	McKinley	M	john.mackinley@365careers.com	0
2	Elizabeth	McFarlane	F	e.mcfarlane@365careers.com	2
3	Kevin	Lawrence	M	kevin.lawrence@365careers.com	1
4	Catherine	Winnfield	F	c.winnfield@365careers.com	0

the value of 0, NOT NULL

NOT NULL Constraint

Customers					
customer_id	first_name	last_name	gender	email_address	number_of_complaints
1	John	McKinley	M	john.mackinley@365careers.com	
2	Elizabeth	McFarlane	F	e.mcfarlane@365careers.com	2
3	Kevin	Lawrence	M	kevin.lawrence@365careers.com	1
4	Catherine	Winnfield	F	c.winnfield@365careers.com	0

NULL

NOT NULL Constraint

Customers						
customer_id	first_name	last_name	gender	email_address	number_of_complaints	feedback
1	John	McKinley	M	john.mackinley@365careers.com	0	*I think…*
2	Elizabeth	McFarlane	F	e.mcfarlane@365careers.com	2	*Great service!*
3	Kevin	Lawrence	M	kevin.lawrence@365careers.com	1	*Great service!*
4	Catherine	Winnfield	F	c.winnfield@365careers.com	0	NONE

NOT NULL Constraint

Customers						
customer_id	first_name	last_name	gender	email_address	number_of_complaints	feedback
1	John	McKinley	M	john.mackinley@365careers.com	0	*I think…*
2	Elizabeth	McFarlane	F	e.mcfarlane@365careers.com	2	*Great service!*
3	Kevin	Lawrence	M	kevin.lawrence@365careers.com	1	*Great service!*
4	Catherine	Winnfield	F	c.winnfield@365careers.com	0	

NULL

Next:

Coding Techniques and Best Practices

Data Manipulation