Databases

INTRODUCTION TO SQL

Introducing databases

A database is an organized collection of structured information, or data, typically stored electronically in a computer system.

patrons

card_num	name	member_year	total_fine
54378	Izzy	2012	9.86
94722	Maham	2020	0
45783	Jasmin	2022	2.05
90123	James	1989	0

books

id	title	author	genre	pub_year
638	Being Mortal	Atul Gawande	Non-Fiction	2015
912	Educated	Tara Westover	Non-Fiction	2018
322	Night	Elie Wiesel	Non-Fiction	1956
156	Where the Wild Things Are	Maurice Sendak	Childrens	1963

checkouts

id	start_date	due_date	card_num	book_id
567	2022-05-13	2022-05-27	54378	638
568	2022-06-10	2022-06-24	54378	322
569	2022-06-27	2022-07-11	45783	156
570	2022-08-14	2022-08-28	90123	912

Introducing databases

patrons

card_num	name	member_year	total_fine
54378	Izzy	2012	9.86
94722	Maham	2020	0
45783	Jasmin	2022	2.05
90123	James	1989	0

Relational databases

- Define relationships between tables of data inside the database

RDBMS stands for Relational Database Management System.

RDBMS is a program used to maintain a relational database.

RDBMS is the basis for all modern database systems such as MySQL, Microsoft SQL Server, Oracle, and Microsoft Access.

patrons

card_num	name	member_year	total_fine
54378	Izzy	2012	9.86
94722	Maham	2020	0
45783	Jasmin	2022	2.05
90123	James	1989	0

books

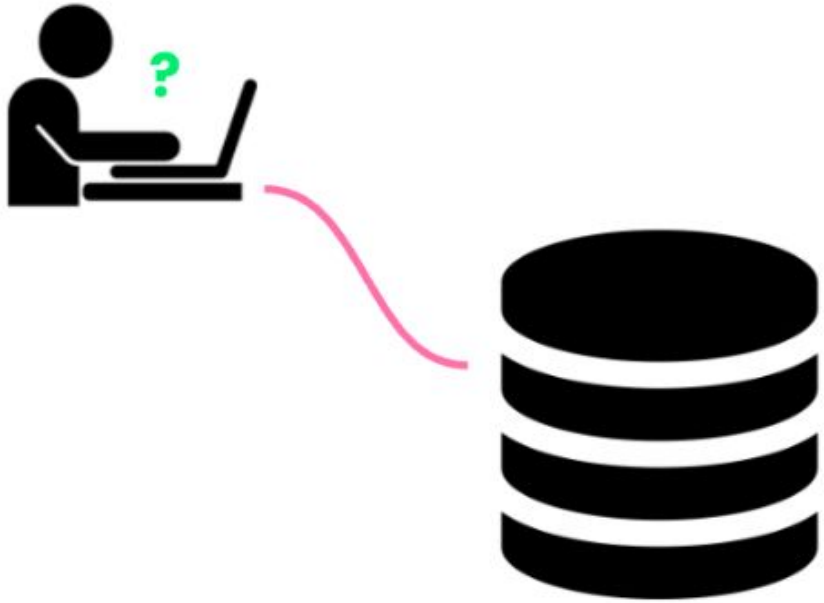
id	title	author	genre	pub_year
638	Being Mortal	Atul Gawande	Non-Fiction	2015
912	Educated	Tara Westover	Non-Fiction	2018
322	Night	Elie Wiesel	Non-Fiction	1956
156	Where the Wild Things Are	Maurice Sendak	Childrens	1963

checkouts

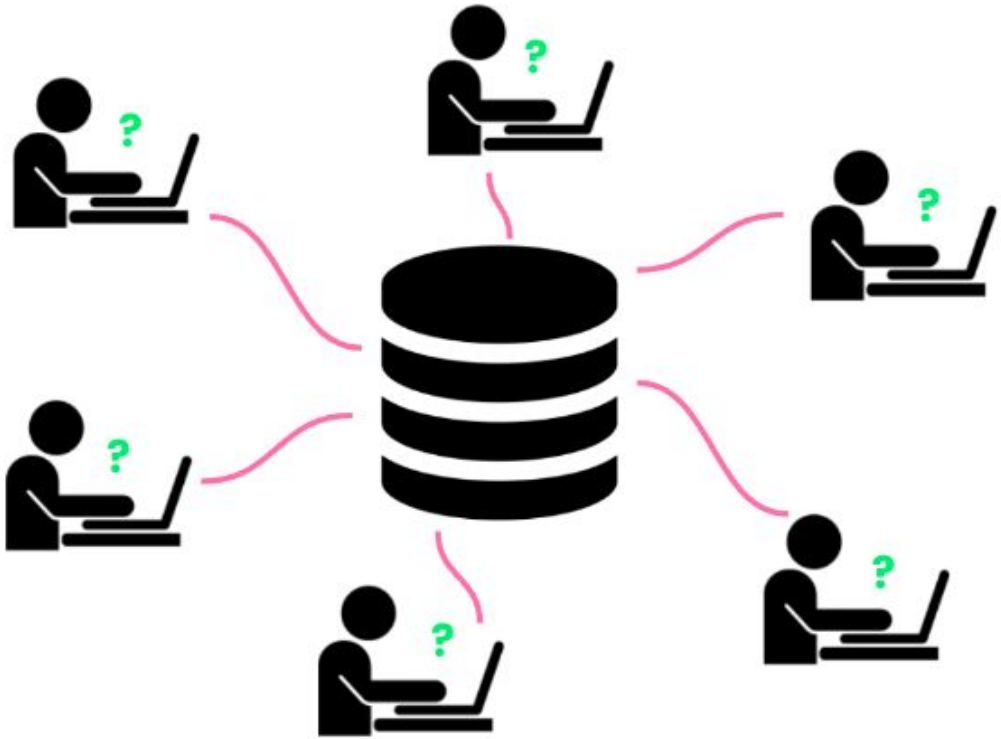
id	start_date	due_date	card_num	book_id
567	2022-05-13	2022-05-27	54378	638
568	2022-06-10	2022-06-24	54378	322
569	2022-06-27	2022-07-11	45783	156
570	2022-08-14	2022-08-28	90123	912

Database advantages

- More storage than spreadsheet applications
- Storage is more secure



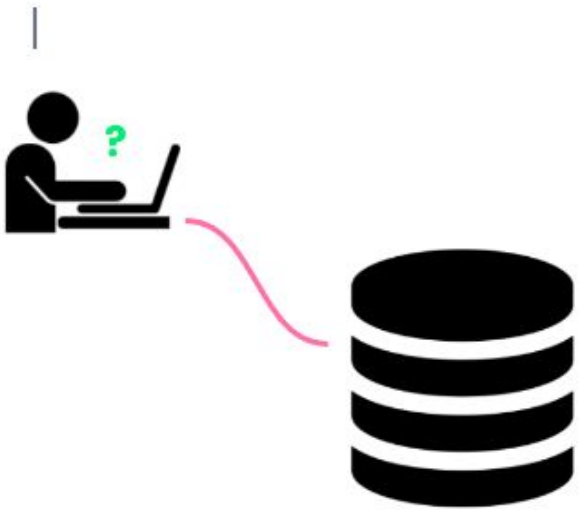
Database advantages



SQL

- Short for Structured Query Language
- The most widely used programming language for databases

```sql
SELECT *
FROM patrons
LIMIT 30
```



Tables

INTRODUCTION TO SQL

A seat at the table

- Table rows and columns are referred to as *records* and *fields*
- Fields are set at database creation; there is no limit to the number of records

patrons

card_num	name	member_year	total_fine
54378	Izzy	2012	9.86
94722	Maham	2020	0
45783	Jasmin	2022	2.05
90123	James	1989	0

books

id	title	author	genre	pub_year
638	Being Mortal	Atul Gawande	Non-Fiction	2015
912	Educated	Tara Westover	Non-Fiction	2018
322	Night	Elie Wiesel	Non-Fiction	1956
156	Where the Wild Things Are	Maurice Sendak	Childrens	1963

checkouts

id	start_date	due_date	card_num	book_id
567	2022-05-13	2022-05-27	54378	638
568	2022-06-10	2022-06-24	54378	322
569	2022-06-27	2022-07-11	45783	156
570	2022-08-14	2022-08-28	90123	912

Good table manners

Table names should...

- be lowercase
- have no spaces—use underscores instead
- refer to a collective group or be plural

patrons

card_num	name	member_year	total_fine
54378	Izzy	2012	9.86
94722	Maham	2020	0
45783	Jasmin	2022	2.05
90123	James	1989	0

Laying the table: records

A record is a row that holds data on an individual observation

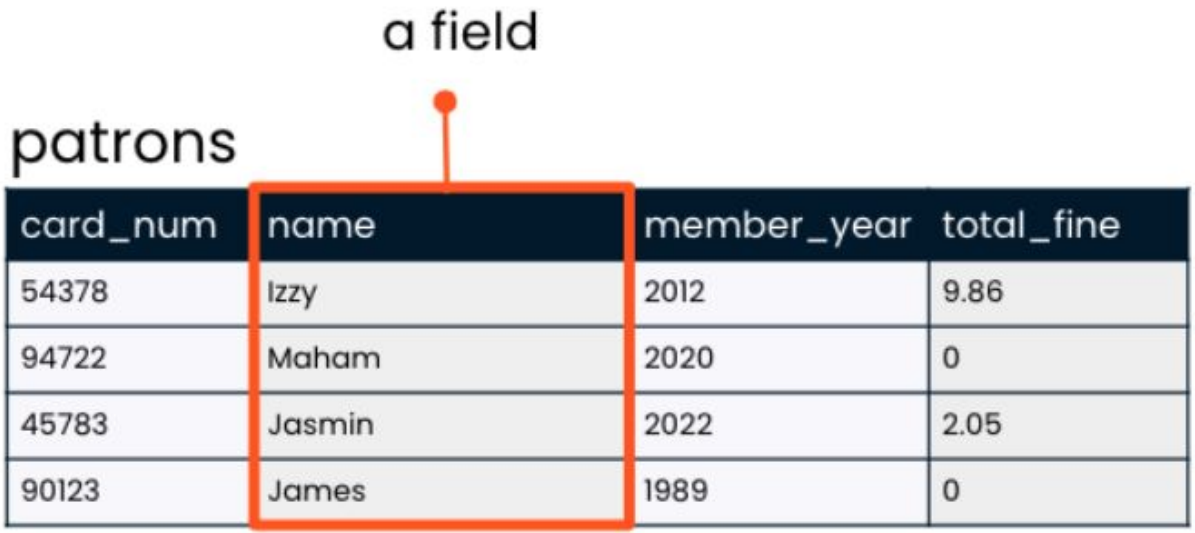
patrons

card_num	name	member_year	total_fine
54378	Izzy	2012	9.86
94722	Maham	2020	0
45783	Jasmin	2022	2.05
90123	James	1989	0

a record

Laying the table: fields

A field is a column that holds one piece of information about all records



More table manners

Field names should...

- be lowercase
- have no spaces
- be singular
- be different from other field names
- be different from the table name

Assigned seats

- *Unique identifiers* are used to identify records in a table
- They are unique and often numbers

patrons

card_num	name	member_year	total_fine
54378	Izzy	2012	9.86
94722	Maham	2020	0
45783	Jasmin	2022	2.05
90123	James	1989	0

unique identifier

The more the merrier

patrons

card_num	name	member_year	total_fine
54378	Izzy	2012	9.86
94722	Maham	2020	0
45783	Jasmin	2022	2.05
90123	James	1989	0

checkouts

id	start_date	due_date	card_num	book_id
567	2022-05-13	2022-05-27	54378	638
568	2022-06-10	2022-06-24	54378	322
569	2022-06-27	2022-07-11	45783	156
570	2022-08-14	2022-08-28	90123	912

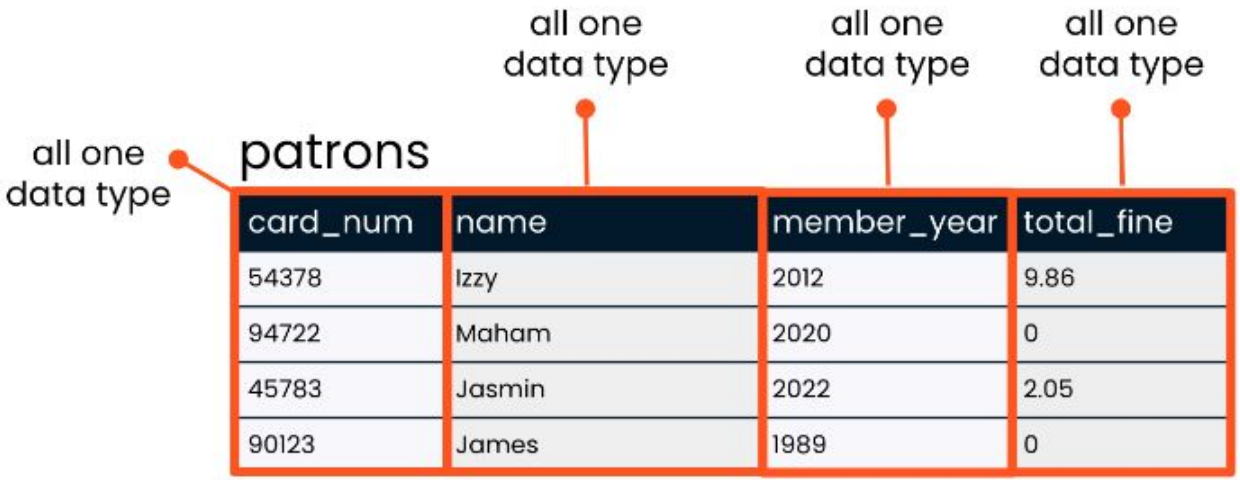
patron_checkouts

card_num	name	member_year	total_fine	checkout_id	start_date	due_date	book_id
54378	Izzy	2012	9.86	567	2022-05-13	2022-05-27	638
54378	Izzy	2012	9.86	568	2022-06-10	2022-06-24	322
94722	Maham	2020	0				
45783	Jasmin	2022	2.05	2022-06-27	2022-07-11	45783	156
90123	James	1989	0	570	2022-08-14	2022-08-28	912

Data

INTRODUCTION TO SQL

SQL data types



card_num	name	member_year	total_fine
54378	Izzy	2012	9.86
94722	Maham	2020	0
45783	Jasmin	2022	2.05
90123	James	1989	0

- Different types of data are stored differently and take up different space
- Some operations only apply to certain data types

Strings

- A string is a sequence of characters such as letters or punctuation
- `VARCHAR` is a flexible and popular string data type in SQL

Integers

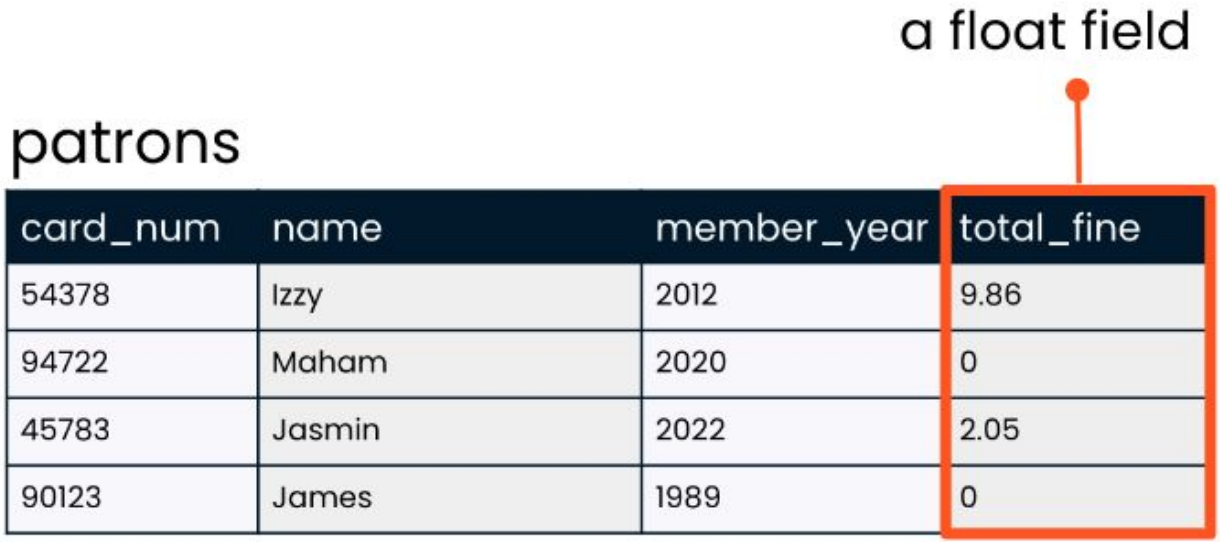
an integer field

patrons

card_num	name	member_year	total_fine
54378	Izzy	2012	9.86
94722	Maham	2020	0
45783	Jasmin	2022	2.05
90123	James	1989	0

- Integers store whole numbers
- `INT` is a flexible and popular integer data type in SQL

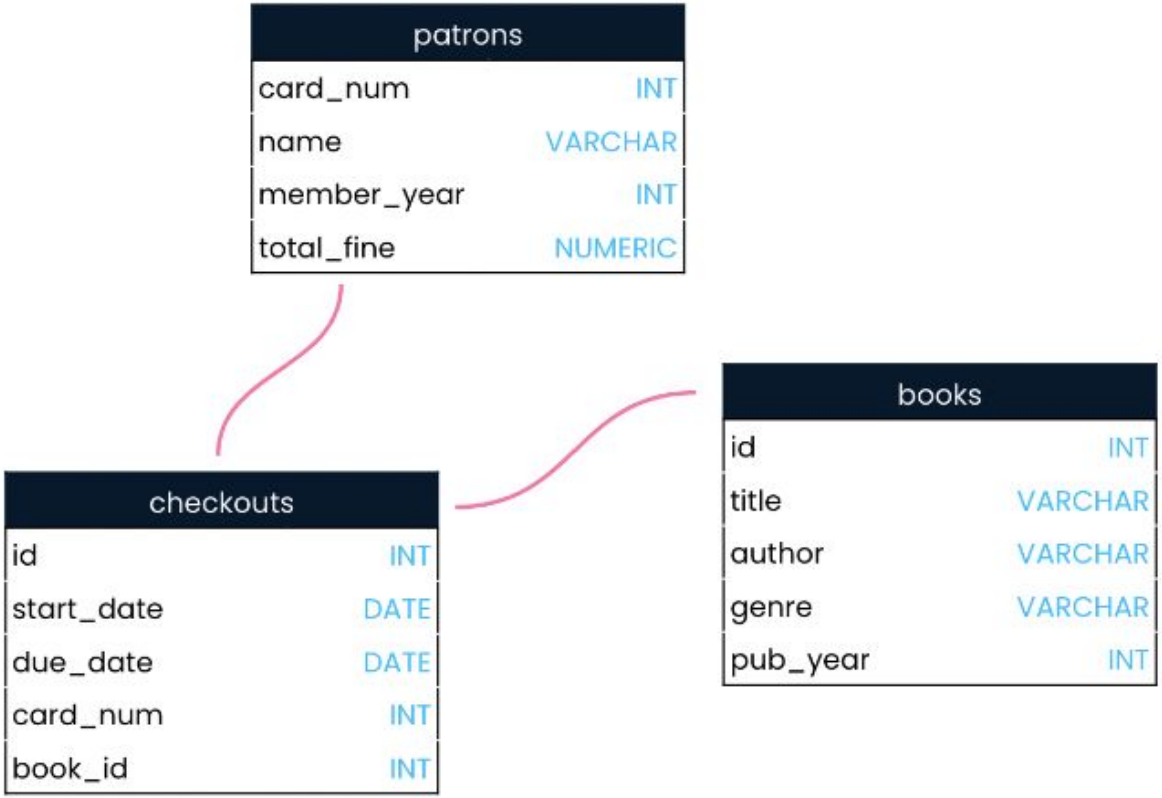
Floats



- Floats store numbers that include a fractional part
- `NUMERIC` is a flexible and popular float data type in SQL

Schemas

A schema is a collection of database objects like tables, triggers, stored procedures, etc.



Introducing queries

What is SQL useful for?

patrons

card_num	name	member_year	total_fine
54378	Izzy	2012	9.86
94722	Maham	2020	0
45783	Jasmin	2022	2.05
90123	James	1989	0

SQL is used to communicate with a database and it is used to store, retrieve, and manipulate data in relational databases

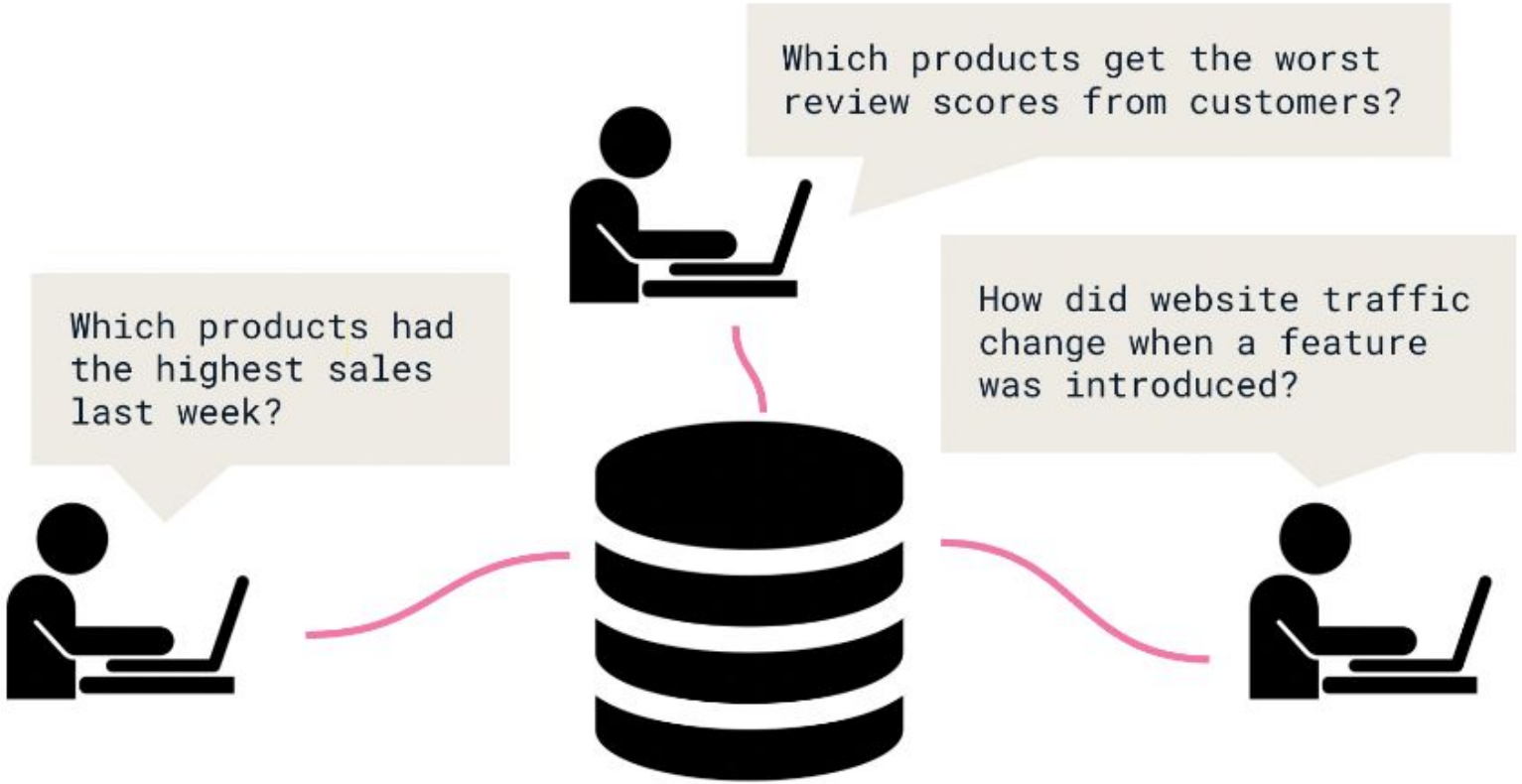
books

id	title	author	genre	pub_year
638	Being Mortal	Atul Gawande	Non-Fiction	2015
912	Educated	Tara Westover	Non-Fiction	2018
322	Night	Elie Wiesel	Non-Fiction	1956
156	Where the Wild Things Are	Maurice Sendak	Childrens	1963

checkouts

id	start_date	due_date	card_num	book_id
567	2022-05-13	2022-05-27	54378	638
568	2022-06-10	2022-06-24	54378	322
569	2022-06-27	2022-07-11	45783	156
570	2022-08-14	2022-08-28	90123	912

Best for large datasets



Keywords

Keywords are reserved words for operations

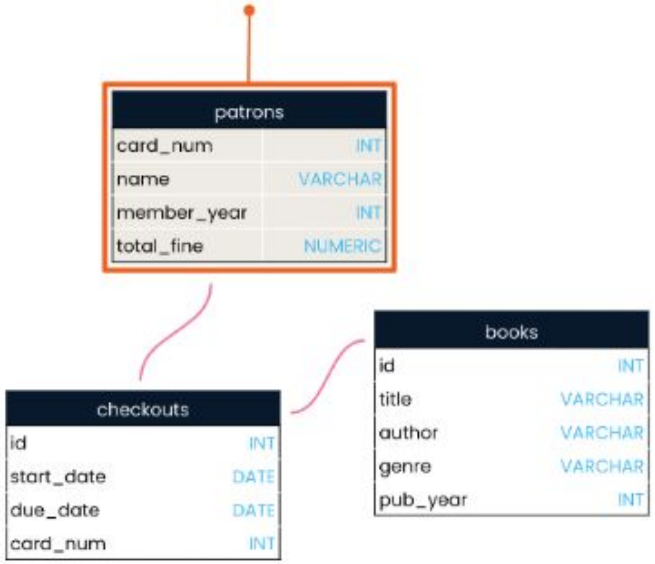
Common keywords: `SELECT`, `FROM`

```
SELECT name
```

```
FROM patrons
```

patrons

card_num	name	member_year	total_fine
54378	Izzy	2012	9.86
94722	Maham	2020	0
45783	Jasmin	2022	2.05
90123	James	1989	0



Our first query

```
SELECT name
FROM patrons;
```

```
| name   |
|--------|
| Izzy   |
| Maham  |
| Jasmin |
| James  |
```

patrons

card_num	name	member_year	total_fine
54378	Izzy	2012	9.86
94722	Maham	2020	0
45783	Jasmin	2022	2.05
90123	James	1989	0

- Query results often called *result set*

Selecting multiple fields

```
SELECT card_num, name
FROM patrons;
```

```
| card_num | name   |
|----------|--------|
| 54378    | Izzy   |
| 94722    | Maham  |
| 45783    | Jasmin |
| 90123    | James  |
```

```
SELECT name, card_num
FROM patrons;
```

```
| name   | card_num |
|--------|----------|
| Izzy   | 54378    |
| Maham  | 94722    |
| Jasmin | 45783    |
| James  | 90123    |
```

Selecting multiple fields

```
SELECT name, card_num, total_fine
FROM patrons;
```

```
| card_num | name   | total_fine |
|----------|--------|------------|
| 54378    | Izzy   | 9.86       |
| 94722    | Maham  | 0          |
| 45783    | Jasmin | 2.05       |
| 90123    | James  | 0          |
```

Selecting all fields

```
SELECT *
FROM patrons;
```

```
| card_num | name   | member_year | total_fine |
|----------|--------|-------------|------------|
| 54378    | Izzy   | 2012        | 9.86       |
| 94722    | Maham  | 2020        | 0          |
| 45783    | Jasmin | 2022        | 2.05       |
| 90123    | James  | 1989        | 0          |
```

Writing queries

INTRODUCTION TO SQL

Aliasing

Use *aliasing* to rename columns

```sql
SELECT name AS first_name, year_hired
FROM employees;
```

```
| first_name | year_hired |
|------------|------------|
| Darius     | 2020       |
| Raven      | 2017       |
| Eduardo    | 2022       |
| Maggie     | 2021       |
| Amy        | 2020       |
| Meehir     | 2021       |
```

Selecting distinct records

JO DISTINCT(UNIQUE) ko hi select krta h

```
SELECT year_hired
FROM employees;
```

```
| year_hired |
|------------|
| 2020       |
| 2017       |
| 2022       |
| 2021       |
| 2020       |
| 2021       |
```

```
SELECT DISTINCT year_hired
FROM employees;
```

```
| year_hired |
|------------|
| 2020       |
| 2017       |
| 2022       |
| 2021       |
```

DISTINCT with multiple fields

employees

id	name	dept_id	job_level_id	year_hired
54378	Darius	1	3	2020
94722	Raven	2	3	2017
45783	Eduardo	2	1	2022
90123	Maggie	3	2	2011
67284	Amy	2	2	2009
26148	Meehir	3	3	2021

```
SELECT dept_id, year_hired
FROM employees;
```

```
| dept_id | year_hired |
|---------|------------|
| 1       | 2020       |
| 2       | 2017       |
| 2       | 2022       |
| 3       | 2021       |
| 2       | 2020       |
| 3       | 2021       |
```

DISTINCT with multiple fields

```
SELECT DISTINCT dept_id, year_hired
FROM employees;
```

```
| dept_id | year_hired |
|---------|------------|
| 1       | 2020       |
| 2       | 2017       |
| 2       | 2022       |
| 3       | 2021       |
| 2       | 2020       |
```

Views

- A *view* is a virtual table that is the result of a saved SQL `SELECT` statement
- When accessed, views automatically update in response to updates in the underlying data

```
CREATE VIEW employee_hire_years AS
SELECT id, name, year_hired
FROM employees;
```

if the update you made to underlying tables is adding or deleting Data, then the view is auto updated with the new data. If you add or delete the columns form the underlying tables (basically the definition of the View), then you need to run sp_RefreshView stored procedure to reflect the new schema in your view.

Using views

```
SELECT id, name
FROM employee_hire_years;
```

```
| id    | name    |
|-------|---------|
| 54378 | Darius  |
| 94722 | Raven   |
| 45783 | Eduardo |
| 90123 | Maggie  |
| 67284 | Amy     |
| 26148 | Meehir  |
```

SQL flavors

INTRODUCTION TO SQL

SQL flavors

- Both free and paid
- All used with relational databases
- Vast majority of keywords are the same
- All must follow universal standards
- Only the additions on top of these standards make flavors different

Two popular SQL flavors

PostgreSQL

- Free and open-source relational database system
- Created at the University of California, Berkeley
- "PostgreSQL" refers to both the PostgreSQL database system and its associated SQL flavor

SQL Server

- Has free and paid versions
- Created by Microsoft
- T-SQL is Microsoft's SQL flavor, used with SQL Server databases

Comparing PostgreSQL and SQL Server

- Like dialects of the same language

PostgreSQL:

```
SELECT id, name
FROM employees
LIMIT 2;
```

```
| id    | name   |
|-------|--------|
| 54378 | Darius |
| 94722 | Raven  |
```

- Example: limiting number of results

SQL Server:

```
SELECT TOP(2) id, name
FROM employees;
```

```
| id    | name   |
|-------|--------|
| 54378 | Darius |
| 94722 | Raven  |
```

Choosing a flavor

Just like with ice cream, any flavor is probably a good choice!