Welcome to the CoGrammar

Tutorial: Relational Databases

The session will start shortly...

Questions? Drop them in the chat.

Software Engineering Session Housekeeping

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly. **(Fundamental British Values: Mutual Respect and Tolerance)**
- No question is daft or silly - **ask them!**
- There are **Q&A sessions** throughout this session, should you wish to ask any follow-up questions.
- If you have any questions outside of this lecture, or that are not answered during this lecture, please do submit these for upcoming Academic Sessions. You can submit these questions here: Questions

Software Engineering Session Housekeeping cont.

- For all **non-academic questions**, please submit a query: www.hyperiondev.com/support
- Report a **safeguarding** incident: www.hyperiondev.com/safeguardreporting
- We would love your **feedback** on lectures: Feedback on Lectures

Enhancing Accessibility: Activate Browser Captions

Why Enable Browser Captions?

- Captions provide **real-time text for spoken content**, ensuring inclusivity.
- Ideal for individuals in noisy or quiet environments or for those with **hearing impairments**.

How to Activate Captions:

1. **YouTube or Video Players:**
   - Look for the CC (Closed Captions) icon and click to enable.
2. **Browser Settings:**
   - Google Chrome: Go to *Settings > Accessibility > Live Captions* and toggle ON.
   - Edge: Enable captions in *Settings > Accessibility*.

Safeguarding & Welfare

We are committed to all our students and staff feeling safe and happy; we want to make sure there is always someone you can turn to if you are worried about anything.

If you are feeling upset or unsafe, are worried about a friend, student or family member, or you feel like something isn't right, speak to our safeguarding team:

Ian Wyles
Designated Safeguarding Lead

Simone Botes

Nurhaan Snyman

Rafiq Manan

Ronald Munodawafa

Tevin Pitts

Scan to report a safeguarding concern

or email the Designated Safeguarding Lead:
Ian Wyles
safeguarding@hyperiondev.com

CoGrammar HyperionDev

Polls

CoGrammar

Polls

- *Refer to the polls section to vote for you option.*

1. Which SQL command is used to delete data from a table in SQLite?

   A. REMOVE

   B. ERASE

   C. DELETE

   D. DROP

Polls

- *Refer to the polls section to vote for you option.*

2. What method is used to connect to a SQLite database in Python?

   A. connect()

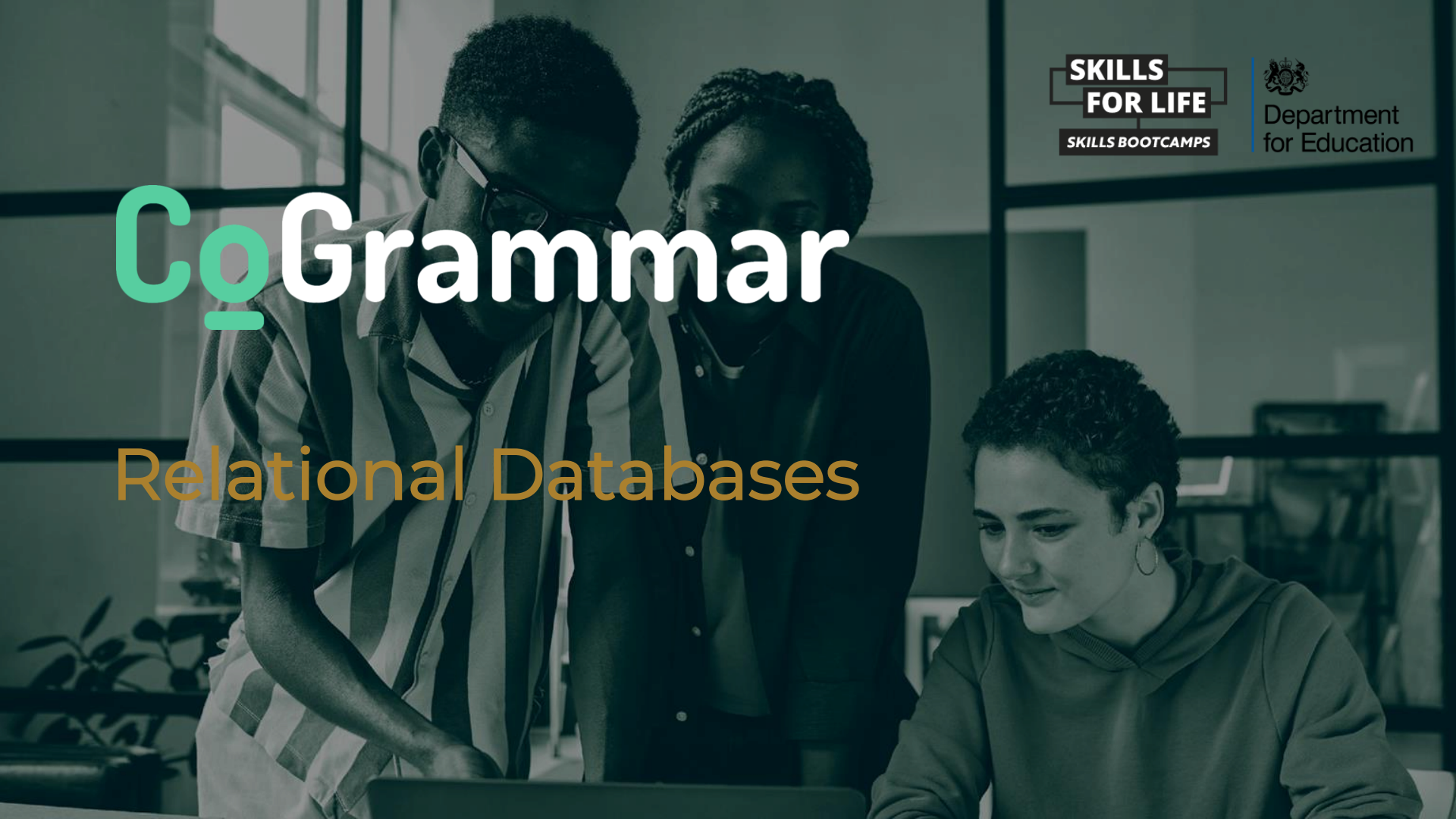
   B. open()

   C. start()

   D. establish()

Learning Outcomes

- Define Normalisation
- Normalise a table to 1NF, 2NF and 3NF
- Create tables with their relationships using SQLite.
- Connect your SQLite database to a Python program


SKILLS
FOR LIFE
SKILLS BOOTCAMPS
Department
for Education
CoGrammar
Relational Databases

Relational Databases

- A relational database is a database based on the relational model of data
- Relational Model
  - A relational model organises data into one or more tables (or "relations") of columns and rows, with a unique key identifying each row.

Relational Databases

- Rows are also called records or tuples.
- Columns are also called attributes.
- Generally, each table/relation represents one "entity type" (such as customer or product).
- The rows represent instances of that type of entity and the columns represent values attributed to that instance.

Keys

- Each row in a table has its own unique key.
- Rows in a table can be linked to rows in other tables by adding a column for the unique key of the linked row
- Allows us to select or modify one and only one row in a table
- unique primary key (PK) for each row in a table
- Foreign and Primary keys

Relationships

- Relationships are a logical connection between different tables (entities), established based on interaction among these tables.
- These relationships can be modelled as an entity-relationship model.

Designing a Relational Database

1. Identify entities and their attributes.
   a. An entity is a real-world object or a concept. This can be a customer or an order.
   b. An attribute is a portion of an entity that's used to describe it, such as their name or the order ID.
2. Identify relationships between entities. An example would be customers place orders.
   a. These are typically identified and implemented using foreign keys
3. Database design should be built using normalisation form.
   a. This is the process of organising data in different tables (Customers table and Order table)

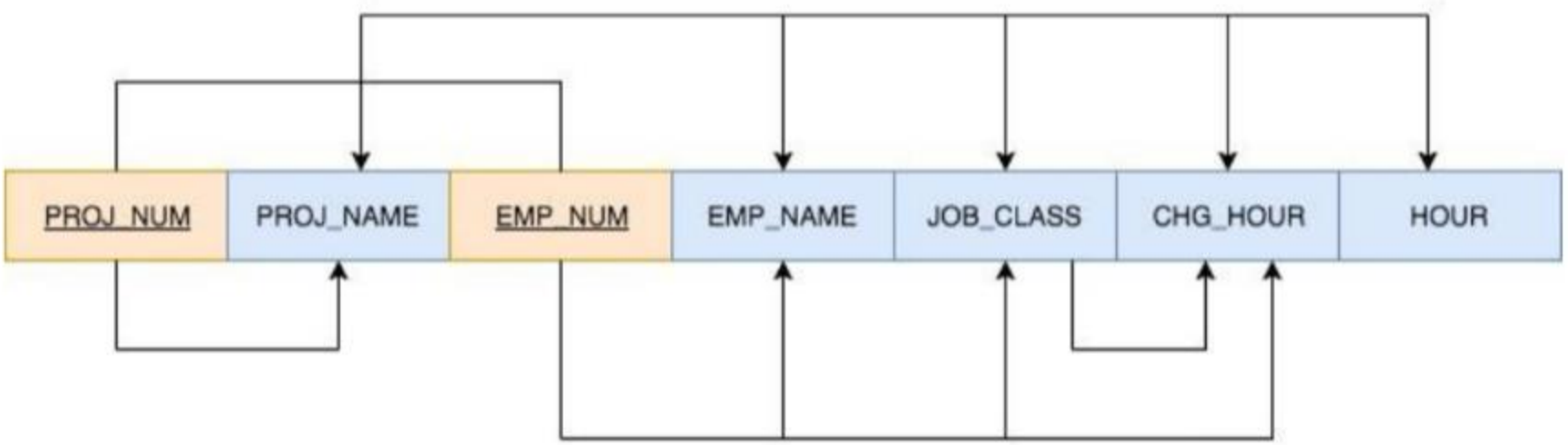
Normalisation

- Process of organising data in a database
- Creating tables and establishing relationships between those tables
- Eliminate redundancy and inconsistent dependency

Normalisation

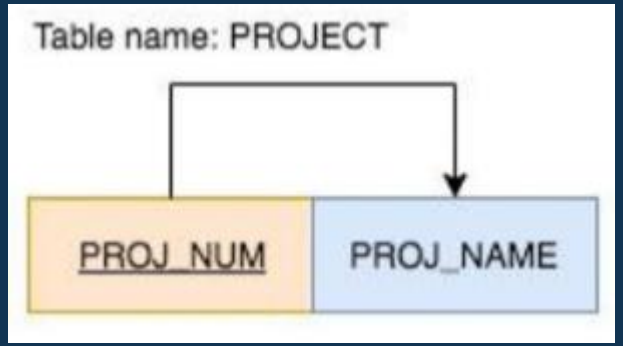
PROJ_NUM	PROJ_NAME	EMP_NUM	EMP_NAME	JOB_CLASS	CHG_HOUR	HOURS
15	Evergreen	103	June Arbaugh	Elect. Engineer	$67.55	23
15	Evergreen	101	John News	Database Designer	$82.00	19
15	Evergreen	105	Alice Johnson	Database Designer	$82.00	35
15	Evergreen	106	William Smithfield	Programmer	$26.66	12
15	Evergreen	102	David Senior	System Analyst	$76.43	12
18	Amberwave	114	Ann Jones	Applications Designer	$38.00	24
18	Amberwave	118	James Frommer	General Support	$14.50	45
18	Amberwave	104	Anne Remoras	System Analyst	$76.43	32
18	Amberwave	112	Darlene Smithson	DSS Analyst	$36.30	44

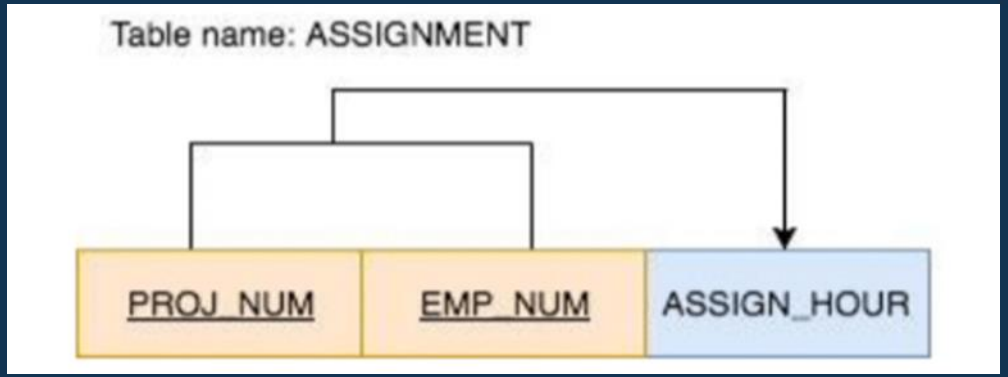
1NF

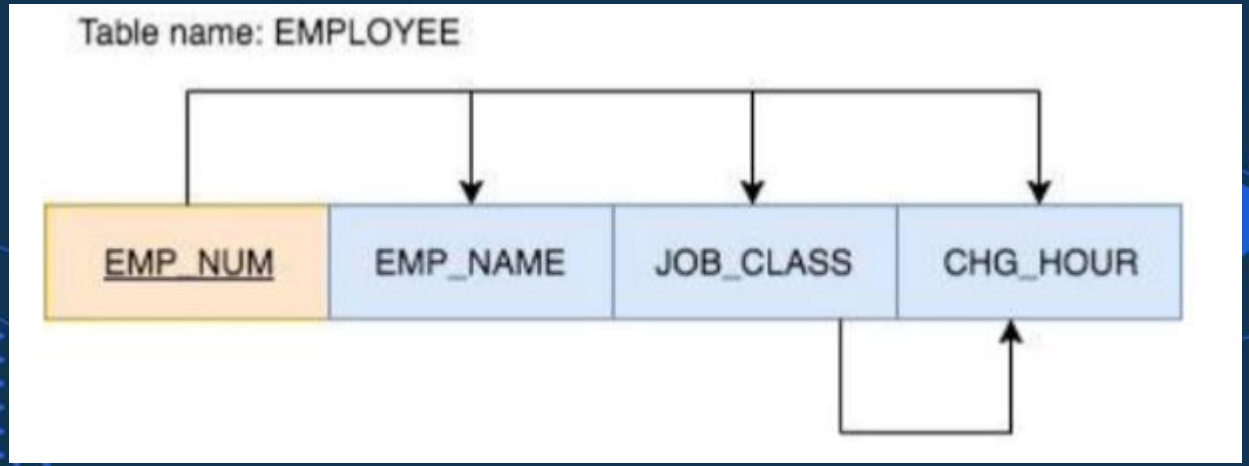


(Coronel & Morris, 2014, pg. 198)

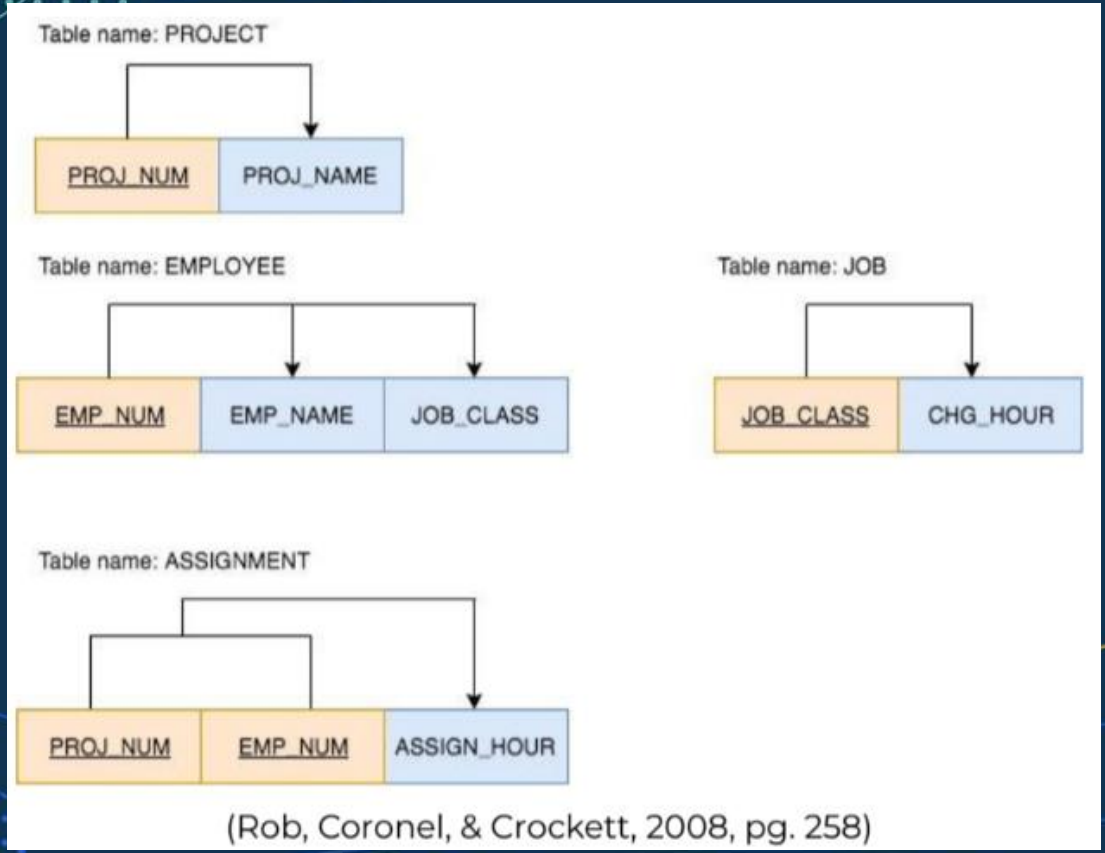
2NF







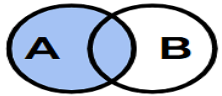
3NF



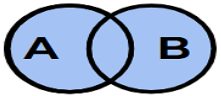
(Rob, Coronel, & Crockett, 2008, pg. 258)

SQL: Accessing Multiple Tables

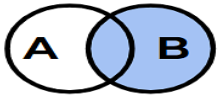
- INNER JOIN - Records match in both tables
- LEFT JOIN - All values in A, and matching values in B
- FULL OUTER JOIN - All values in both tables



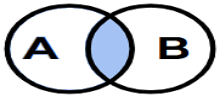

```
SELECT *
FROM A
LEFT JOIN B
ON A.id = B.id
```



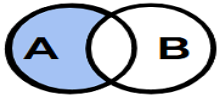

```
SELECT *
FROM A
FULL OUTER JOIN B
ON A.id = B.id
```



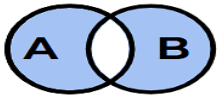

```
SELECT *
FROM A
RIGHT JOIN B
ON A.id = B.id
```



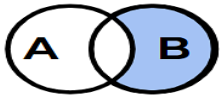

```
SELECT *
FROM A
INNER JOIN B
ON A.id = B.id
```




```
SELECT *
FROM A
LEFT JOIN B
ON A.id = B.id
WHERE B.id IS NULL
```




```
SELECT *
FROM A
FULL OUTER JOIN B
ON A.id = B.id
WHERE A.id IS NULL
OR B.id IS NULL
```




```
SELECT *
FROM A
RIGHT JOIN B
ON A.id = B.id
WHERE A.id IS NULL
```

CoGrammar

DBMS

- Database management system (DBMS)
  - Collection of programs that manages the database structure and access to the data
  - Acts as intermediary between user and database
  - Hides internal complexity from the user

DBMS Advantages

- Better data sharing
  - End users have more efficient access to better managed data due to the DBMS managing the data and access to the data
- Improved data integration
  - The DBMS helps provide a clearer and more integrated view of the organisation's operations to the end-users

DBMS Advantages

- Minimised data inconsistency
  - Occurs when different versions of the same data appear in different places. Properly designed databases greatly reduces the probability of data inconsistency.
- Improved data access
  - A query is a specific request for data manipulation sent to the DBMS. The DBMS makes it possible to produce quick answers to spur-of-the-moment queries.

DBMS Advantages

- Improved decision making
  - Better quality information (on which decisions are made) is generated due to better managed data and improved data access.
- Increased end-user productivity
  - The availability of data and the ability to transform data in to usable information encourages end-users to make quicker and more informed decisions.

Connecting SQL to Python

CoGrammar

SQLite Syntax

```
import sqlite3

db = sqlite3.connect('data/student_db')
cursor = db.cursor()

cursor.execute('''
    CREATE TABLE student(id INTEGER PRIMARY KEY, name TEXT,
                grade INTEGER)
''')

db.commit()
```

Basic SQLite Syntax

```
cursor.execute('''INSERT INTO student(name, grade)
                  VALUES(?,?)''', (name1,grade1))
db.commit()

students_ = [(name1,grade1),(name2,grade2),(name3,grade3)]
cursor.executemany(''' INSERT INTO student(name, grade) VALUES(?,?)''',
students_)
db.commit()
```

Let’s take a short break

CoGrammar

Let’s get coding!

CoGrammar

Polls

CoGrammar

Polls

- *Refer to the polls section to vote for you option.*

1. **Which of the following statements best describes a relational database?**

   A. Stores data in a single table

   B. Organises data into structured tables related by common fields

   C. Stores data in a hierarchical structure

   D. Stores data in unstructured format

Polls

- *Refer to the polls section to vote for you option.*

2. Which of the following best describes a foreign key in a relational database?

   A. A key used for securing database connections

   B. A key that uniquely identifies each record in a table

   C. A key that establishes a relationship between two tables

   D. A key used for sorting data in a table

Lesson Conclusion

- Relational Databases: Based on the relational model of data that organises data into one or more tables (or "relations") of columns and rows, with a unique key identifying each row.
- Normalisation: Process of organising data in a database.
- DBMS: Collection of programs that manages the database structure and access to the data
- SQLite: Lightweight, self-contained SQL database engine.

Follow-up Activities

CoGrammar

Follow-up Activities

Objective: Normalise the table provided on GitHub and create it using SQLite.

Connect it to a Python program and perform a few queries.

Questions and Answers

CoGrammar

Thank you for attending

SKILLS FOR LIFE
SKILLS BOOTCAMPS

Department for Education

CoGrammar