

1. What is a **relation?**

- A relation is essentially a table within a database that holds data. It is made up of rows and columns

2. What is a **tuple?**

- Each row is a tuple, as it refers to all the columns pertaining to an entry or instance

3. What is an **attribute?**

- The columns within a relation are the attributes, as they each are used to identify or relate to the entry

4. What is a **domain? Give one example domain for a column in this lab.**

- A domain refers to the rules we have within our relation./ For instance, we can add a rule that assures we have the class year within a certain range, and we can also add a rule that limits the varchar range to two characters, etc..

5. Explain **schema vs **instance** with one example.**

- A schema is a way to specify the databases, containing all the rules, attributes, and domains as well. For instance, when we create a table, we have the column names, we have the rules, and we also decide if we want primary keys or not. An instance is a specific entry within our data. For instance, we enter a student that fits the domain and schema of the relation, and we can use the key to find the student along with the attributes referring to them.

6. What is a **key? Why do we use a **primary key**?**

- A key is what we use to uniquely identify a tuple within a relation. We use primary keys in order to identify the specific tuple we need in case tuples have identical data (like name, grade, age, etc)

Query Query History

```
1 SELECT * FROM Students
2 ORDER BY class_year
```

Data Output Messages Notifications

The screenshot shows a database interface with a toolbar at the top containing various icons for file operations like new, open, save, and export. Below the toolbar is a table with six rows of student information. The columns are labeled: student_id [PK] integer, name character varying (250), major character varying (2), and class_year integer.

	student_id [PK] integer	name character varying (250)	major character varying (2)	class_year integer
1	1	Amina Johnson	CS	2026
2	5	Nia Williams	PS	2026
3	2	David Chen	EE	2027
4	3	Maria Lopez	BI	2028
5	6	Olu Adeyemi	CS	2028
6	4	Jordan Smith	MA	2029

Query Query History

```
1 SELECT * FROM Enrollments  
2 WHERE grade IS NULL;
```

Data Output Messages Notifications



A screenshot of a database application interface showing the results of a SQL query. The top navigation bar includes tabs for 'Data Output', 'Messages', and 'Notifications'. Below the toolbar, there is a table structure with three rows of data. The columns are labeled 'student_id' (integer), 'course_id' (integer), and 'grade' (integer). The first row has values 6, 101, and [null]. The second row has values 1, 102, and [null]. The third row is currently selected, indicated by a grey background.

	student_id integer	course_id integer	grade integer
1	6	101	[null]
2	1	102	[null]

Query Query History

```
1 SELECT * FROM Students  
2 WHERE name LIKE 'A%';
```

Data Output Messages Notifications

The screenshot shows a database management interface with a toolbar at the top containing various icons for file operations, a search bar, and a SQL tab. Below the toolbar is a table with four columns: student_id, name, major, and class_year. The table has one row of data: student_id 1, name Amina Johnson, major CS, and class_year 2026.

	student_id [PK] integer	name character varying (250)	major character varying (2)	class_year integer
1	1	Amina Johnson	CS	2026

Query Query History

```
1 SELECT * FROM Courses  
2 WHERE credits >= 3;
```

Data Output Messages Notifications



	course_id [PK] integer	title character varying (250)	credits integer
1	101	Introduction to Computer Scien...	4
2	102	Calculus I	4
3	103	Database Systems	3

Query Query History

```
1 SELECT * FROM Students  
2 WHERE major='CS';
```

Query Query History

1 `SELECT name, student_id FROM Students`

Data Output Messages Notifications



	name character varying (250)	student_id [PK] integer
1	Amina Johnson	1
2	David Chen	2
3	Maria Lopez	3
4	Jordan Smith	4
5	Nia Williams	5
6	Olu Adeyemi	6

	student_id integer	course_id integer	grade integer
1	1	101	98
2	1	103	95
3	6	102	86
4	6	103	83
5	4	102	87
6	5	104	100
7	6	101	[null]
8	1	102	[null]

	student_id [PK] integer	name character varying (250)	major character varying (2)	class_year integer
1	1	Amina Johnson	CS	2026
2	2	David Chen	EE	2027
3	3	Maria Lopez	BI	2028
4	4	Jordan Smith	MA	2029
5	5	Nia Williams	PS	2026
6	6	Olu Adeyemi	CS	2028

	course_id [PK] integer	title character varying (250)	credits integer
1	101	Introduction to Computer Scien...	4
2	102	Calculus I	4
3	103	Database Systems	3
4	104	Public Speaking	2

```

1 CREATE TABLE Students(
2     student_id int PRIMARY KEY,
3     name varchar(255),
4     major varchar(2),
5     class_year int CHECK (class_year >= 2026 AND class_year <= 2029)
6 )
7
8 CREATE TABLE Courses(
9     course_id int PRIMARY KEY,
10    title varchar(250),
11    credits int CHECK (credits >= 1 AND credits <= 4)
12 )
13
14 CREATE TABLE Enrollments(
15     student_id int,
16     course_id int,
17     FOREIGN KEY (student_id) REFERENCES Students(student_id),
18     FOREIGN KEY (course_id) REFERENCES Courses(course_id),
19     title varchar(250),
20     credits int CHECK (credits >= 1 AND credits <= 4)
21 )

```

π student_id, name from students

σ major='cs'

Github:

<https://github.com/MrMir247/Miracle-s-DBM-Repo.git>