

Database Queries

SQL Editor

```
SELECT *  
FROM customers;
```

Output pane

	cid character(4)	name text	city text	discount numeric(5,2)
1	c001	Tiptop	Duluth	10.00
2	c002	Basics	Dallas	12.00
3	c003	Allied	Dallas	8.00
4	c004	ACME	Duluth	8.00
5	c005	Weyland-Yutani	Acheron	0.00
6	c006	ACME	Kyoto	0.00

Customer Query: displays all fields related to the customer

SQL Editor

```
SELECT *  
FROM agents;
```

Output pane

	aid character(3)	name text	city text	percent real
1	a01	Smith	New York	6
2	a02	Jones	Newark	6
3	a03	Brown	Tokyo	7
4	a04	Gray	New York	6
5	a05	Otasi	Duluth	5
6	a06	Smith	Dallas	5
7	a08	Bond	London	7

Agent Query: displays all fields related to agents

SQL Editor

```
SELECT *  
FROM products;
```

Output pane

	pid character(3)	name text	city text	quantity integer	priceusd numeric(10,2)
1	p01	comb	Dallas	111400	0.50
2	p02	brush	Newark	203000	0.50
3	p03	razor	Duluth	150600	1.00
4	p04	pen	Duluth	125300	1.00
5	p05	pencil	Dallas	221400	1.00
6	p06	folder	Dallas	123100	2.00
7	p07	case	Newark	100500	1.00
8	p08	clip	Newark	200600	1.25

Product Query: displays all fields related to products

SQL Editor

```
SELECT *  
FROM orders;
```

Output pane

	pid character(3)	name text	city text	quantity integer	priceusd numeric(10,2)
1	p01	comb	Dallas	111400	0.50
2	p02	brush	Newark	203000	0.50
3	p03	razor	Duluth	150600	1.00
4	p04	pen	Duluth	125300	1.00
5	p05	pencil	Dallas	221400	1.00
6	p06	folder	Dallas	123100	2.00
7	p07	case	Newark	100500	1.00
8	p08	clip	Newark	200600	1.25

Order Query: displays all fields related to orders

Definitions and Distinctions

Primary Key

A primary key is the singular column that causes a row to be completely unique. Primary keys uniquely identify tuples in order to reduce confusion and completely eliminate the possibility of duplicates in a database. Primary keys are necessary for all databases while being managed in a RDBMS, as there are far too many records the software will need to manage. Since primary keys are heavily relied upon, it is generally considered 'best practice' to assign a value that does not violate the "law of least astonishment".

Candidate Key

In relation to the primary key, candidate keys are columns or a set of data values that can be candidates for primary or uniquely identifiable keys, hence its name. However, unlike the primary key, candidate keys can be a combination of columns as well. There is no limit to how many may exist in any given table, although they must all qualify for the chance of being a primary key.

Super Key

Keeping the definitions of candidate and primary keys in mind, a super key also denotes columns that may uniquely identify rows. The only difference between a candidate key and a super key lies in the quantity; super keys minimize the amount of columns needed to uniquely identify a row.

Data Types

In order for a database to exist, it must possess many items. These objects include but are not limited to, tuples, columns, fields, tables or relations, schemas and much more. One of the required items, a data type, deals with the individual columns within a table. Typically used to better understand what kind of data is being presented, data types give numbers, words, and many other items categories. This technique not only helps developers to better organize tables, but also assists the RDBS in only allowing specific types of data into specific columns. Although it may seem confusing, the following example helps to shape the concept. Let us say we are a business that small educational business, specializing in offering educational courses for niche topics. In order to ensure each student has their own easily accessible record for grades, attendance, payment and graduation, we would need to organize many different forms of data into many different tables. There would first be a "main" table, one that would hold only students, their identification numbers, possibly their date of enrollment, their projected graduating date, and perhaps a few other peripherals. Each of the aforementioned would form their own respective columns, however, in order to ensure accuracy and efficiency, different data types would be assigned to each column. There would also need to be a decision made for each column's nullability, which determines if a column may be left blank. For the student's name, we would use the VARCHAR data type; VARCHAR accepts a variety of characters with a differentiable length rather than data types that specify fixed lengths (CHARACTER). Obviously, each student would need to have a name on record in order to be able to enroll into the system, which will prevent any errors in attendance or financial records. Next, identification numbers, depending on whether or not the developer wishes to use a combination of characters and integers, would utilize the INT data type. INT in

comparison to VARCHAR accepts only integers rather than characters and integers which prevents the user from accidentally inserting data into an inappropriate column. This column would also be unable to have any amount of nullability, as it is a high candidate for a primary key. Following ID's, students will also need to have a column for the date in which they enrolled. Since it would be theoretically possible for a student to apply for a course and not enroll, deciding not to attend for example, this column would be nullable, allowing the developer the opportunity of filling out the column at a later point in time. The expected graduation date would have similar properties to the enrollment date, as it falls under similar circumstances. Both the graduation date and enrollment date would also most likely share the same data type, depending on the preference of the developer. Should the developer choose to require the year, day, month, hour, minute, and second, they would choose the TIMESTAMP data type. However, if they were to only need the actual date, they may choose the DATE data type. Here is what the table would look like:

Students			
sid	name	enroll_date	graduation_date
<i>VARCHAR/INT</i>	<i>VARCHAR</i>	<i>DATE/TIMESTAMP</i>	<i>DATE/TIMESTAMP</i>
0001	SMITH, JOHN	10-2-2005	5-1-2006
0002	DOE, JANE	12-7-2005	7-6-2006
0002	JONES, MARY	2-14-2006	4-8-2007
0002	MENDEZ, ROBERT	8-12-2006	10-5-2007

Relational Rules

There are three types of relational rules:

1. **First Normal Form**
 - a. Data must be isolated
2. **Access Rows by Content Only**
 - a. The user may ask for the what the data is, but not the location
3. **All Rows Must be Unique**
 - a. No two rows may be the same