EXERCISE-12

Intro to Constraints: NOT NULL and UNIQUE Constraints

Name: Vedhasree S

Register Number: 240701580

Department: CSE

Global Fast Foods has been very successful this past year and has opened several new stores. They need to add a table to their database to store information about each of their store's locations. The owners want to make sure that all entries have an identification number, date opened, address, and city and that no other entry in the table can have the same email address. Based on this information, answer the following questions about the global_locations table. Use the table for your answers.

Global Fast Foods global_locations Table						
NAME	TYPE	LENGTH	PRECISION	SCALE	NULLABLE	DEFAULT
Id						
name						
date_opened						
address						
city						
zip/postal code						
phone						
email						
manager_id						
Emergency contact						

1. What is a "constraint" as it relates to data integrity?

   Answer:

   A constraint is a rule used to maintain data integrity in a database. It ensures that the data entered in a table is valid, accurate, and consistent according to defined rules. Constraints prevent invalid data and help maintain the correctness and reliability of the database.

2. What are the limitations of constraints that may be applied at the column level and at the table level?

   Answer:

   At the column level, constraints can be applied only to a single column. At the table level, constraints can be applied to one or more columns together. Some constraints like PRIMARY KEY and UNIQUE that involve multiple columns must be defined at the table level.

3. Why is it important to give meaningful names to constraints?

   Answer:

   It is important to give meaningful names to constraints so that they can be easily identified and understood. Meaningful names help in maintaining, modifying, or debugging the database when constraints need to be changed or referenced later.

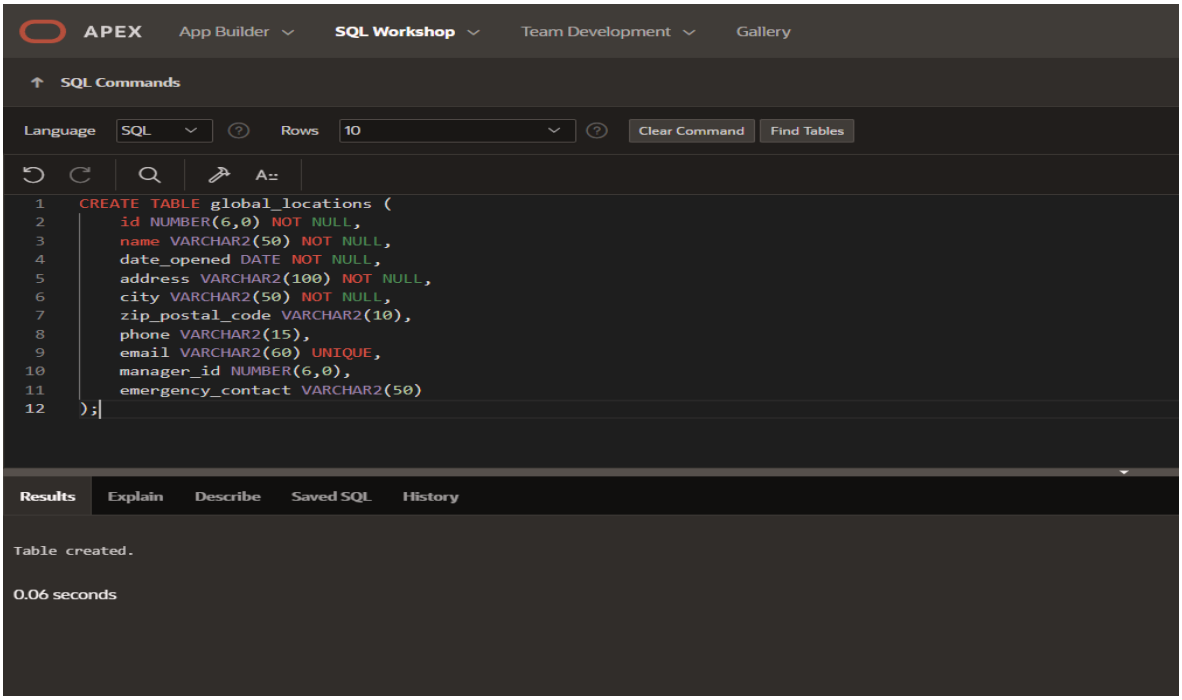
4. Based on the information provided by the owners, choose a datatype for each column. Indicate the length, precision, and scale for each NUMBER datatype.

NAME	TYPE	LENGTH	PRECISION	SCALE
id	NUMBER	-	6	0
name	VARCHAR2	50	-	-
date_opened	DATE	-	-	-
address	VARCHAR2	100	-	-
city	VARCHAR2	50	-	-
zip/postal code	VARCHAR2	10	-	-
phone	VARCHAR2	15	-	-
email	VARCHAR2	60	-	-
manager_id	NUMBER	-	6	0
emergency_contact	VARCHAR2	50	-	-

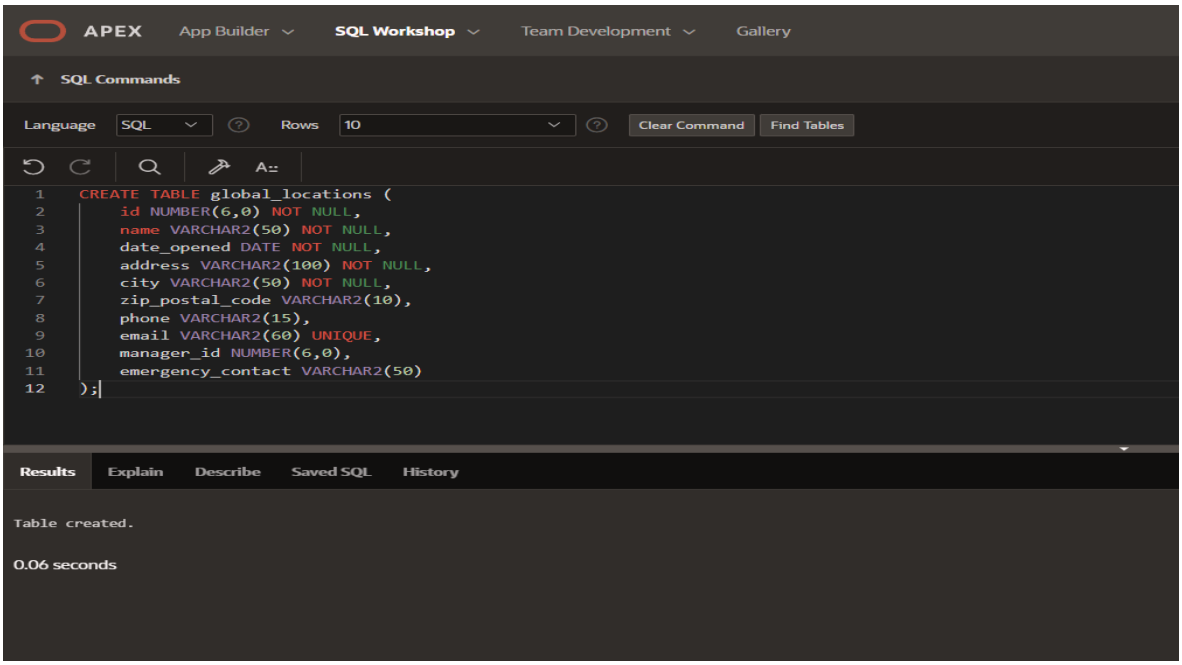
5. Use "(nullable)" to indicate those columns that can have null values.

NAME	TYPE	LENGTH	PRECISION	SCALE	NULLABLE
id	NUMBER	-	6	0	NOT NULL
name	VARCHAR2	50	-	-	NOT NULL
date_opened	DATE	-	-	-	NOT NULL
address	VARCHAR2	100	-	-	NOT NULL
city	VARCHAR2	50	-	-	NOT NULL
zip/postal code	VARCHAR2	10	-	-	NULLABLE
phone	VARCHAR2	15	-	-	NULLABLE
email	VARCHAR2	60	-	-	NULLABLE
manager_id	NUMBER	-	6	0	NULLABLE
emergency_contact	VARCHAR2	50	-	-	NULLABLE

6. Write the CREATE TABLE statement for the Global Fast Foods locations table to define the constraints at the column level.



7. Execute the CREATE TABLE statement in Oracle Application Express.



8. Execute a DESCRIBE command to view the Table Summary information.

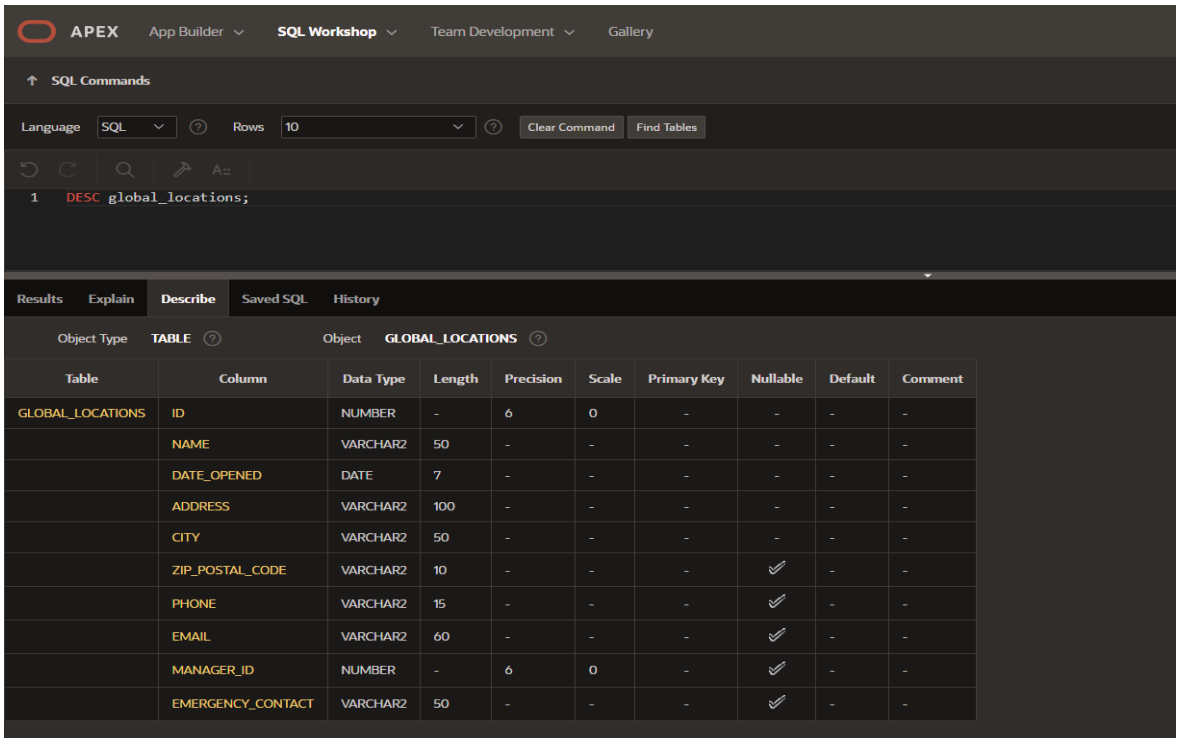
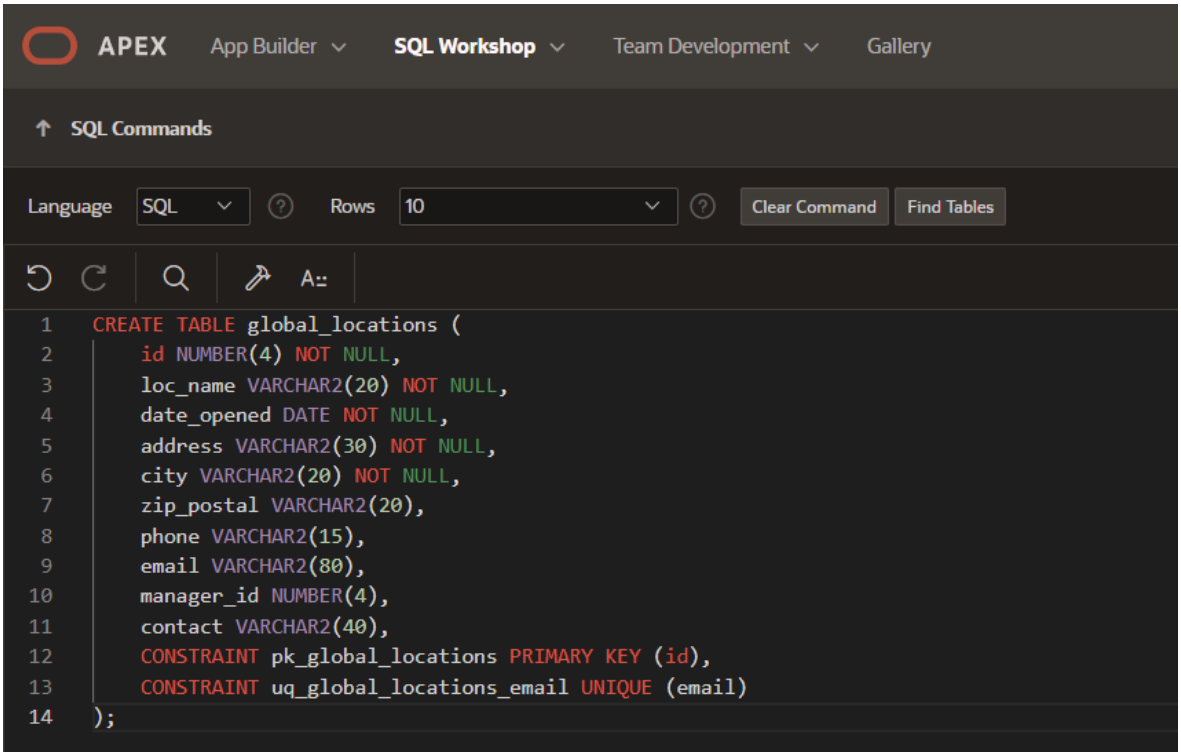


Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
GLOBAL_LOCATIONS	ID	NUMBER	-	6	0	-	-	-	-
	NAME	VARCHAR2	50	-	-	-	-	-	-
	DATE_OPENED	DATE	7	-	-	-	-	-	-
	ADDRESS	VARCHAR2	100	-	-	-	-	-	-
	CITY	VARCHAR2	50	-	-	-	-	-	-
	ZIP_POSTAL_CODE	VARCHAR2	10	-	-	-	✓	-	-
	PHONE	VARCHAR2	15	-	-	-	✓	-	-
	EMAIL	VARCHAR2	60	-	-	-	✓	-	-
	MANAGER_ID	NUMBER	-	6	0	-	✓	-	-
	EMERGENCY_CONTACT	VARCHAR2	50	-	-	-	✓	-	-

9. Rewrite the CREATE TABLE statement for the Global Fast Foods locations table to define the UNIQUE constraints at the table level. Do not execute this statement.

NAME	TYPE	LENGTH	PRECISION	SCALE	NULLABLE	DEFAULT
id	number	4				
loc_name	varchar2	20			X	
	date					
address	varchar2	30				
city	varchar2	20				
zip_postal	varchar2	20			X	
phone	varchar2	15			X	
email	varchar2	80			X	
manager_id	number	4			X	
contact	varchar2	40			X	



Primary Key, Foreign Key and Check Constraints

1. What is the purpose of a

- PRIMARY KEY
- FOREIGN KEY
- CHECK CONSTRAINT

Answer:

- PRIMARY KEY: The PRIMARY KEY uniquely identifies each record (row) in a table. It ensures that no two rows have the same value in that column (or set of columns) and that the value is not NULL.
- FOREIGN KEY: The FOREIGN KEY creates a link between two tables. It ensures referential integrity, meaning the value in one table must match a value in another table's PRIMARY KEY.
- CHECK CONSTRAINT: The CHECK constraint ensures that values in a column meet a specific condition. It helps maintain the validity and accuracy of data.

2. Using the column information for the animals table below, name constraints where applicable at the table level, otherwise name them at the column level. Define the primary key (animal_id). The license_tag_number must be unique. The admit_date and vaccination_date columns cannot contain null values.

```
animal_id          NUMBER(6),
name               VARCHAR2(25),
license_tag_number NUMBER(10),
admit_date         DATE,
adoption_id        NUMBER(5),
vaccination_date   DATE
```

Answer:

Primary Key:

- Column: animal_id
- Purpose: Uniquely identifies each animal.
- Constraint name (example): animals_animal_id_pk

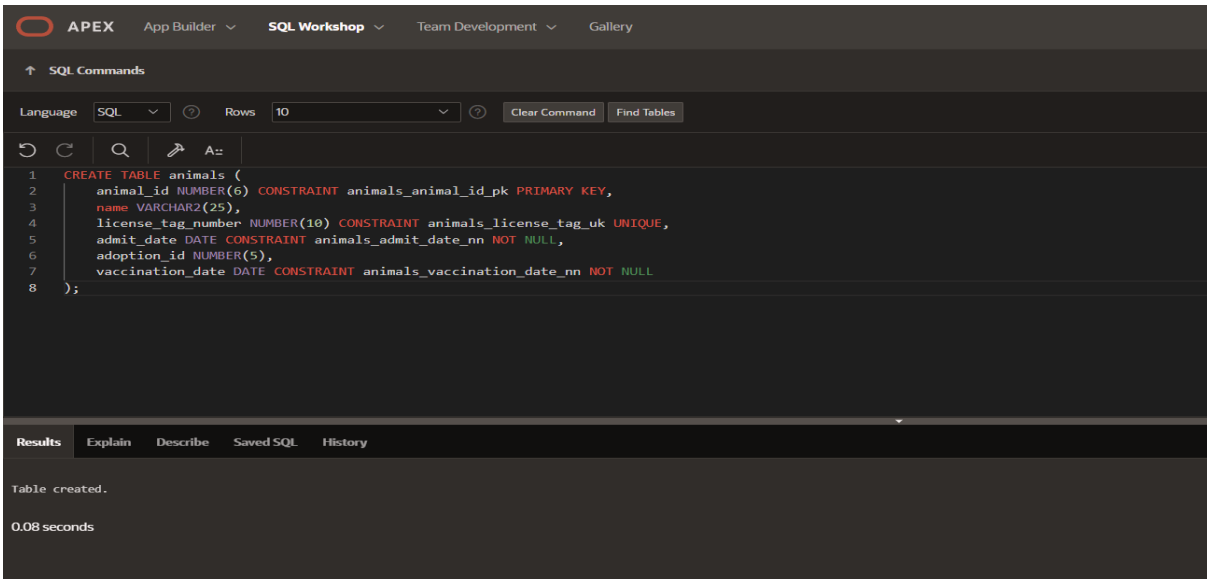
Unique Constraint:

- Column: license_tag_number
- Purpose: Ensures no two animals have the same license tag number.
- Constraint name (example): animals_license_tag_uk

Not Null Constraints:

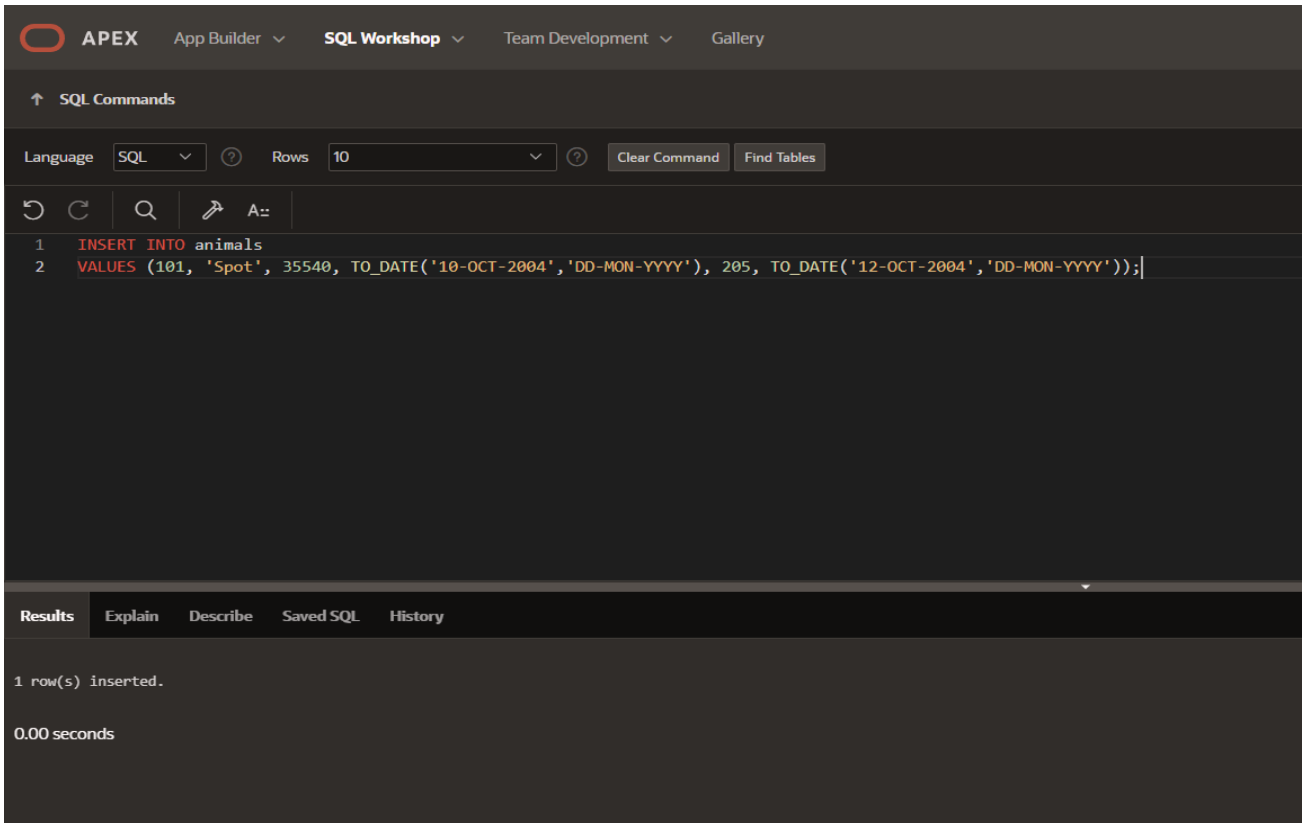
- Columns: admit_date, vaccination_date
- Purpose: These columns cannot have empty (NULL) values.
- Constraint names (examples): animals_admit_date_nn, animals_vaccination_date_nn

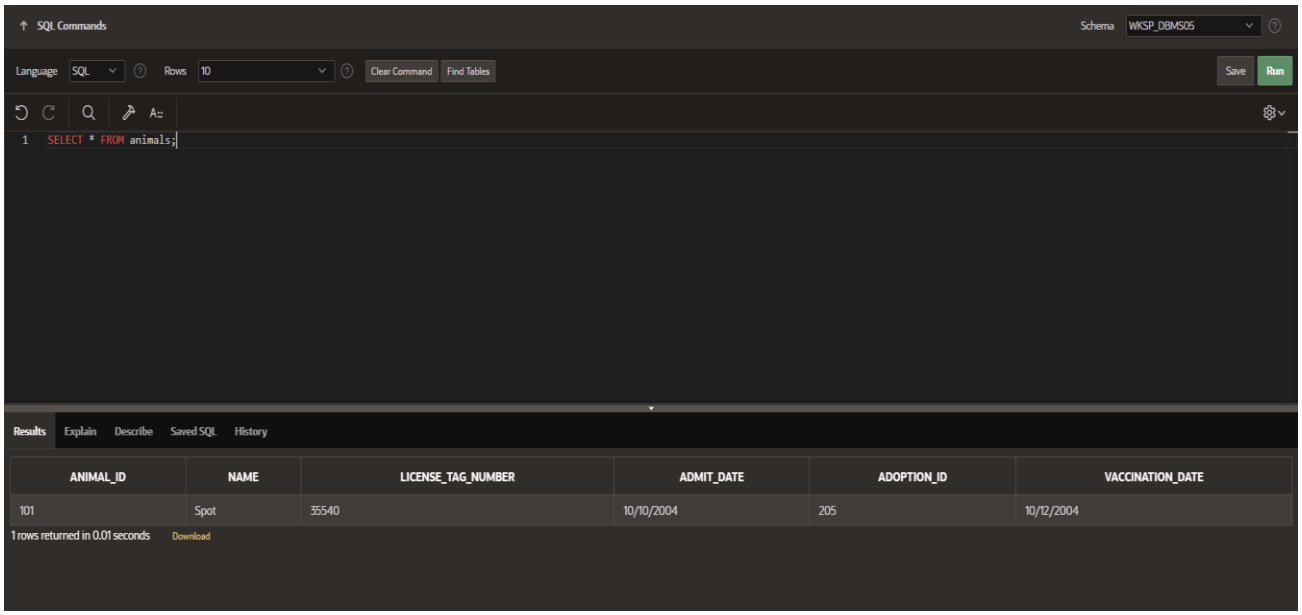
3. Create the animals table. Write the syntax you will use to create the table.



4. Enter one row into the table. Execute a SELECT * statement to verify your input. Refer to the graphic below for input.

ANIMAL_ID	NAME	LICENSE_TAG_NUMBER	ADMIT_DATE	ADOPTION_ID	VACCINATION_DATE
101	Spot	35540	10-Oct-2004	205	12-Oct-2004





5. Write the syntax to create a foreign key (adoption_id) in the animals table that has a corresponding primary-key reference in the adoptions table. Show both the column-level and table-level syntax. Note that because you have not actually created an adoptions table, no adoption_id primary key exists, so the foreign key cannot be added to the animals table.

   Column-Level Foreign Key Syntax

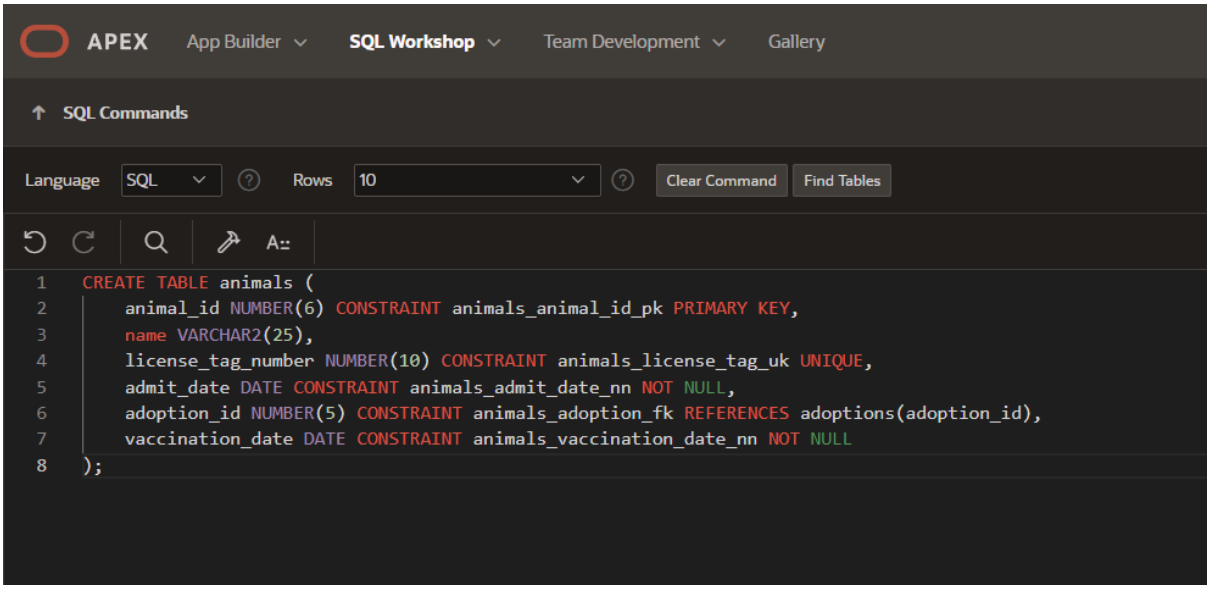
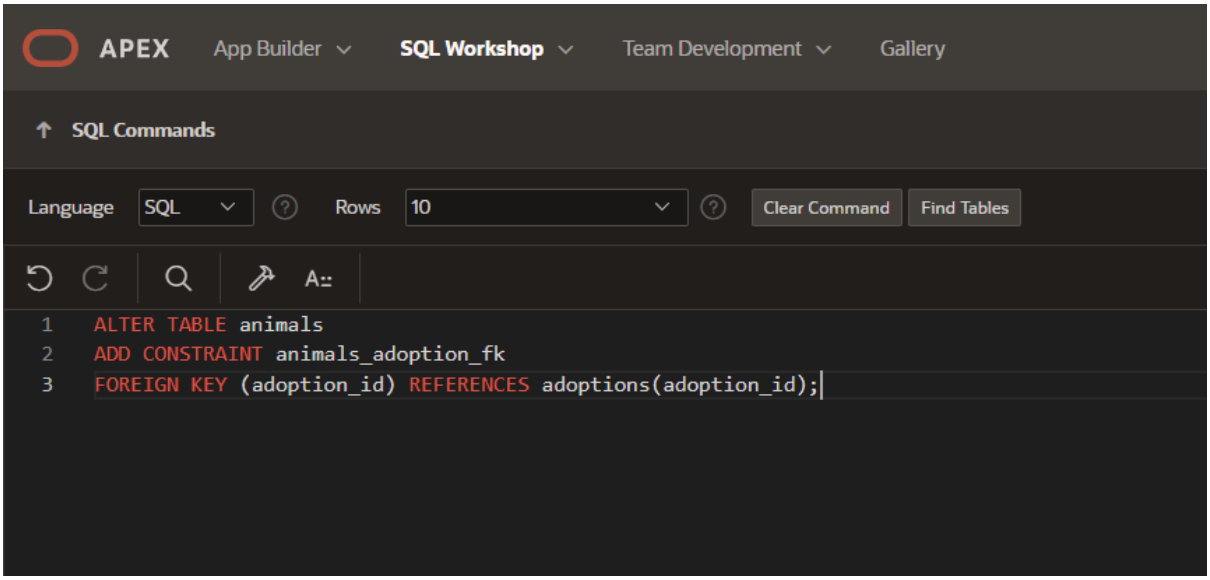


   Table-Level Foreign Key Syntax



6. What is the effect of setting the foreign key in the ANIMAL table as:
   a. ON DELETE CASCADE
   b. ON DELETE SET NULL

   Answer:
   ON DELETE CASCADE: If a row in the parent table (adoptions) is deleted, all related rows in the animals table with the same adoption_id are automatically deleted.
   ON DELETE SET NULL: If a row in the parent table (adoptions) is deleted, the adoption_id in the related rows of the animals table is set to NULL, but the rows themselves remain in the table.

7. What are the restrictions on defining a CHECK constraint?

   Answer:
   A CHECK constraint can only reference columns within the same table and must be a logical condition that evaluates to TRUE or FALSE. It cannot include subqueries or use non-deterministic functions like SYSDATE. At the column level, it can reference only that column, while at the table level, it can reference multiple columns in the same table.