Answers 3.3

* **Step 1**

A screenshot of a computer

Description automatically generated

* **Step 2**

**A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated**

1. category\_id integer NOT NULL DEFAULT nextval('category\_category\_id\_seq'::regclass)

* **NOT NULL**: The category\_id column cannot have NULL values, ensuring that every record must have a value in this column.
* **DEFAULT nextval(...)**: Automatically assigns unique values to category\_id using the category\_category\_id\_seq sequence. This eliminates the need to manually assign unique identifiers.

1. name text COLLATE pg\_catalog."default" NOT NULL

* **NOT NULL**: The name column must be filled and cannot be empty.
* **COLLATE pg\_catalog."default"**: Specifies the collation for sorting and comparing text, which is especially useful in multilingual databases.

1. last\_update timestamp with time zone NOT NULL DEFAULT now()

* **NOT NULL**: The last\_update column cannot be left empty.
* **DEFAULT now()**: Automatically sets the current timestamp when a new record is added. This helps track the time of the most recent update.

1. CONSTRAINT category\_pkey PRIMARY KEY (category\_id)

* Defines the primary key for the table, applied to the category\_id column.
* **PRIMARY KEY**: Ensures the uniqueness of values in the category\_id column and creates an index for faster lookups.
* **Why are these constraints important?**
* **Data consistency**: Constraints like NOT NULL and PRIMARY KEY prevent the insertion of records with missing or duplicate key information.
* **Automation**: Default values like nextval(...) and now() simplify data management and reduce the need for manual input of essential values.
* **Referential integrity**: The primary key allows this table to establish relationships with other tables in the database.
* **Efficient querying**: The primary key index improves the performance of data lookups.
* **Step 3**

**A screenshot of a computer

Description automatically generated** **A screenshot of a computer

Description automatically generated**

* **Step 4**

**A close up of a text

Description automatically generated** **A screenshot of a computer

Description automatically generated**

* **Step 5**

Using Excel for steps 1 to 4 is easier for beginners due to its visual, user-friendly interface. However, SQL is far better for handling large datasets, offering scalability, precision, and tools like constraints and keys to ensure data integrity. Excel requires more manual effort for complex tasks, increasing the risk of errors, making SQL the preferred choice for large-scale data management.

* **Bonus Task**

**A screenshot of a computer code

Description automatically generated** **A screenshot of a computer

Description automatically generated**