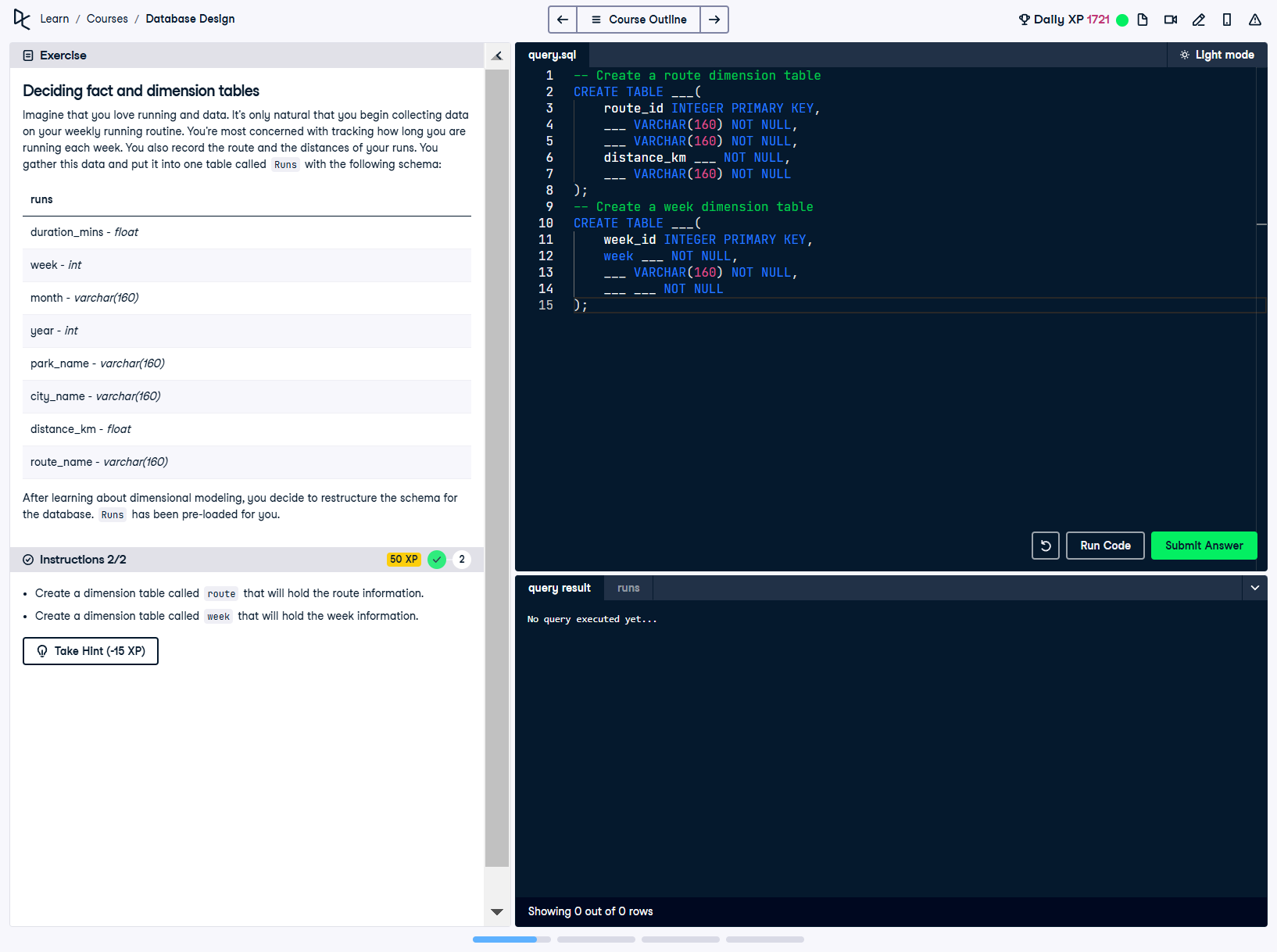
# Deciding Fact and Dimension Tables (Corrected)



## Question

Imagine that you love running and data. It’s only natural that you begin collecting data on your weekly running routine. You’re not concerned with tracking how long you are running each week. You also record the route and the distances of your runs. You gather this data and put it into one table called 'Runs' with the following schema:  
- duration\_mins: float  
- week: int  
- month: varchar(160)  
- year: int  
- park\_name: varchar(160)  
- city\_name: varchar(160)  
- distance\_km: float  
- route\_name: varchar(160)  
  
After learning about dimensional modeling, you decide to restructure the schema for the database. 'Runs' has been pre-loaded for you.  
Instructions:  
- Create a dimension table called 'route' that will hold the route information.  
- Create a dimension table called 'week' that will hold the week information.

## Corrected SQL Solution

-- Create a route dimension table  
CREATE TABLE route (  
 route\_id INTEGER PRIMARY KEY,  
 park\_name VARCHAR(160) NOT NULL,  
 city\_name VARCHAR(160) NOT NULL,  
 distance\_km FLOAT NOT NULL,  
 route\_name VARCHAR(160) NOT NULL  
);  
  
-- Create a week dimension table  
CREATE TABLE week (  
 week\_id INTEGER PRIMARY KEY,  
 week INTEGER NOT NULL,  
 month VARCHAR(160) NOT NULL,  
 year INTEGER NOT NULL  
);

## Answer Explanation

In dimensional modeling, fact and dimension tables are used to optimize data organization for analytical purposes:  
- \*\*Route Dimension Table:\*\* The 'route' table captures route-related details such as park name, city name, distance, and route name. Each route is uniquely identified by 'route\_id', which serves as the primary key.  
- \*\*Week Dimension Table:\*\* The 'week' table stores temporal information such as the week, month, and year. 'week\_id' acts as the primary key to uniquely identify each week.  
By restructuring the schema into dimension tables, the data becomes more organized and efficient for analysis.