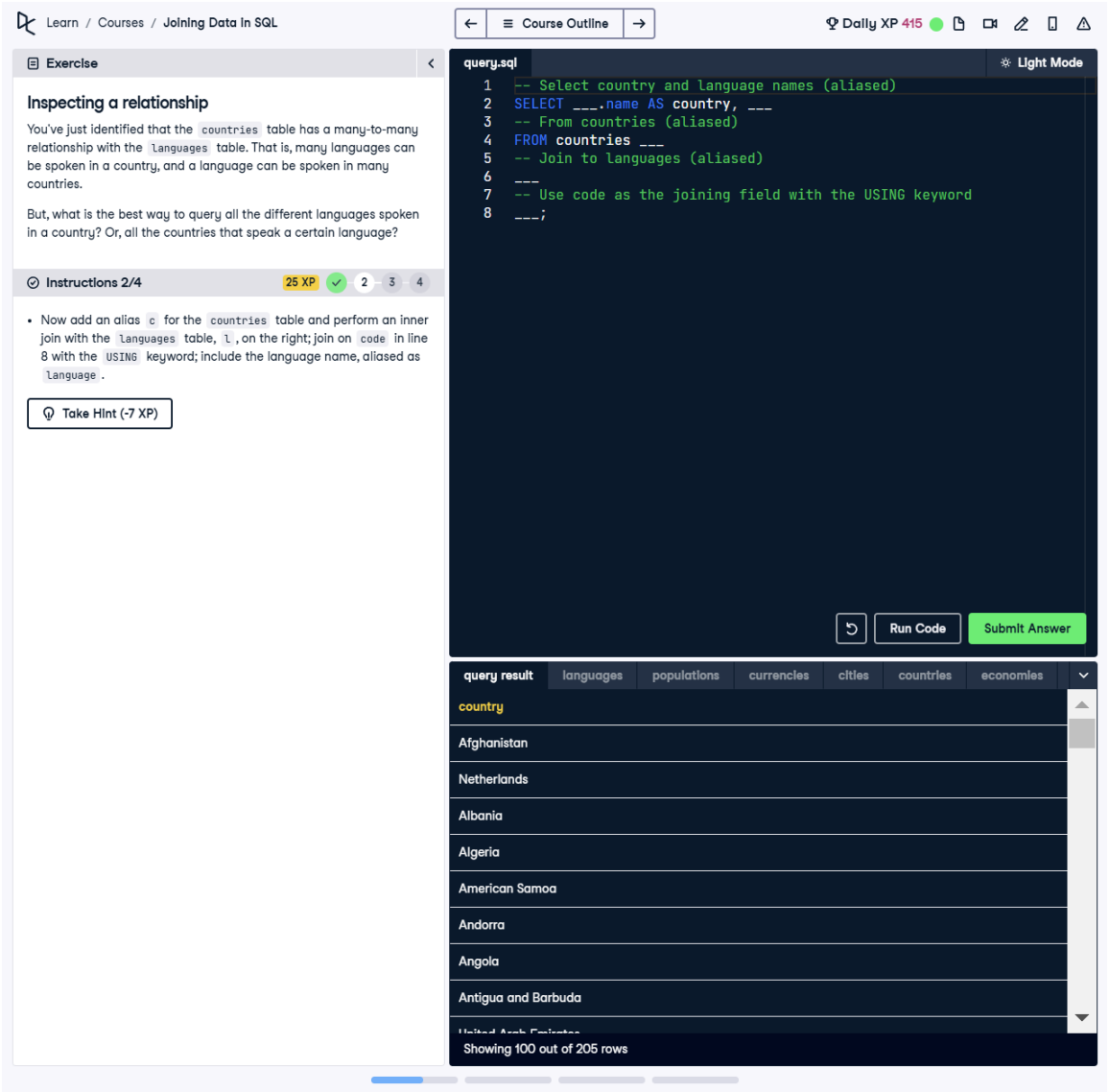
Inspecting a Relationship - Fixing Ambiguous Column Reference

In this exercise, you are exploring how to inspect a relationship between the 'countries' table and the 'languages' table. An error occurred due to an ambiguous column reference because the 'name' column exists in both tables. To resolve this, the alias for the 'countries' table (c) is used to explicitly specify the column.

Below is the instruction and initial setup:



-- Full Answer in SQL

```
SELECT c.name AS country,
       l.name AS language
FROM countries AS c
INNER JOIN languages AS l
USING (code);
```

Explanation: This query resolves the ambiguity by explicitly specifying the alias 'c' for the 'countries' table and 'l' for the 'languages' table. The 'name' column from 'countries' is selected as 'c.name' and aliased as 'country', while the 'name' column from 'languages' is selected as 'l.name' and aliased as 'language'. The USING clause joins the tables on the 'code' column, retrieving the country name and associated language for each match.