# Creating Vertical Partitions in SQL - Corrected

## Question:

In this exercise, you need to create a new table `film\_descriptions` containing two fields: `film\_id` (INT) and `long\_description` (TEXT). Then, copy data from the `film` table into the new table, drop the column from `film`, and use a `JOIN` to reconstruct the original table.

## Full Answer (SQL Code):

-- Create a new table called film\_descriptions  
CREATE TABLE film\_descriptions (  
 film\_id INT,  
 long\_description TEXT  
);  
  
-- Copy the descriptions from the film table  
INSERT INTO film\_descriptions  
SELECT film\_id, long\_description FROM film;  
   
-- Drop the column in the original table  
ALTER TABLE film DROP COLUMN long\_description;  
  
-- Join to create the original table  
SELECT \* FROM film   
JOIN film\_descriptions USING(film\_id);

## Explanation of the Answer:

The `CREATE TABLE` statement creates `film\_descriptions` to store `long\_description` separately. The `INSERT INTO ... SELECT` statement copies existing data from `film`. The `ALTER TABLE DROP COLUMN` removes `long\_description` from `film`, completing the vertical partitioning. Finally, the `JOIN` statement allows reconstruction of the original table by merging `film` and `film\_descriptions`.