

Q)Next drop the table takes2 (and its rows, as a result), and create it again, but this time with a primary key. Run the insert again and measure how long it takes to run. Give its query plan, and explain why the time taken is different this time. Compare the time taken with the sum of the times for the previous two parts of the question. What do you conclude from the above.

Solution:

```
CREATE TABLE takes2 (  
    id integer PRIMARY KEY,  
    course_id varchar(20),  
    sec_id varchar(10),  
    semester varchar(6),  
    year numeric(4,0),  
    grade varchar(2)  
);  
O/P:
```

```
CREATE TABLE  
Time: 4.559 ms
```

followed by

```
INSERT INTO takes2 (id, course_id, sec_id, semester, year, grade)  
SELECT id::integer, course_id, sec_id, semester, year, grade FROM takes  
WHERE 1=0;
```

(had to do this cause was getting dtype mismatch error)

```
O/P:  
INSERT 0 0  
Time: 0.933 ms
```

Explanation:

Based on the time taken for each of the previous two parts of the question, we can conclude that adding a primary key constraint to an existing table using ALTER TABLE command can take significantly longer than creating a new table with the primary key constraint.

This is because adding a primary key constraint requires PostgreSQL to scan through the entire table and verify that the column(s) used to create the primary key do not contain duplicate values. This operation can be time-consuming, especially for larger tables.

On the other hand, creating a new table with a primary key constraint does not require scanning through an existing table, so it can be relatively faster.