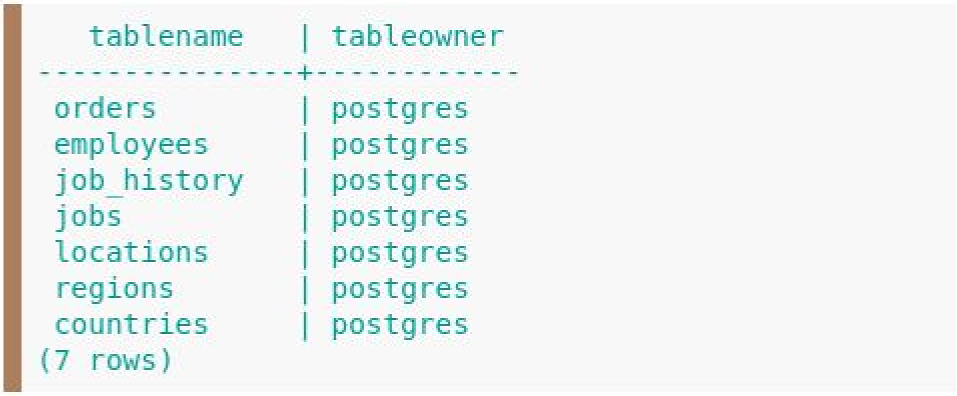
Alter table Queries

1. Write a SQL statement to rename the table countries to country_new.

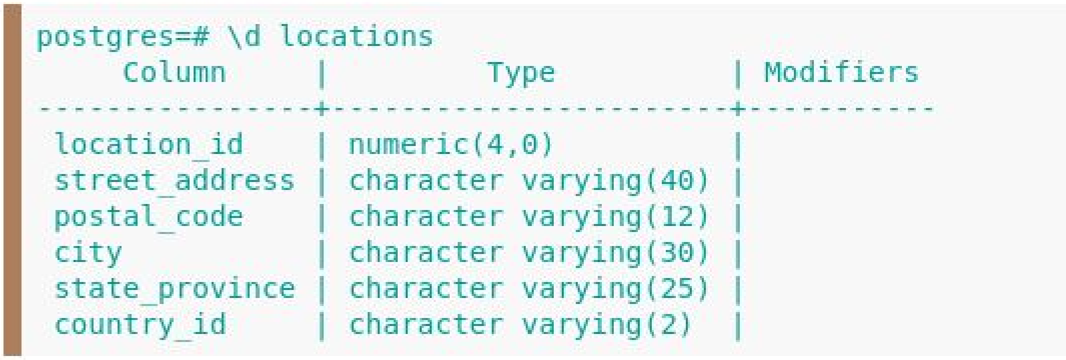
   Here is the list of tables.



```
   tablename    | tableowner
---------------+------------
 orders         | postgres
 employees      | postgres
 job_history    | postgres
 jobs           | postgres
 locations      | postgres
 regions        | postgres
 countries      | postgres
(7 rows)
```

2. Write a SQL statement to add a column region_id to the table locations.

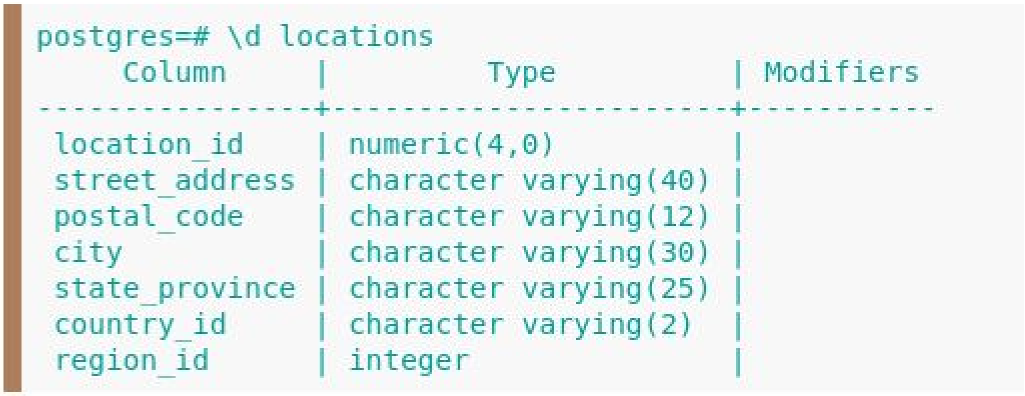
   Here is the structure of the table locations.



```
postgres=# \d locations
     Column     |         Type          | Modifiers
----------------+-----------------------+-----------
 location_id    | numeric(4,0)          |
 street_address | character varying(40) |
 postal_code    | character varying(12) |
 city           | character varying(30) |
 state_province | character varying(25) |
 country_id     | character varying(2)  |
```

3. Write a SQL statement to change the data type of the column region_id to text in the table locations.

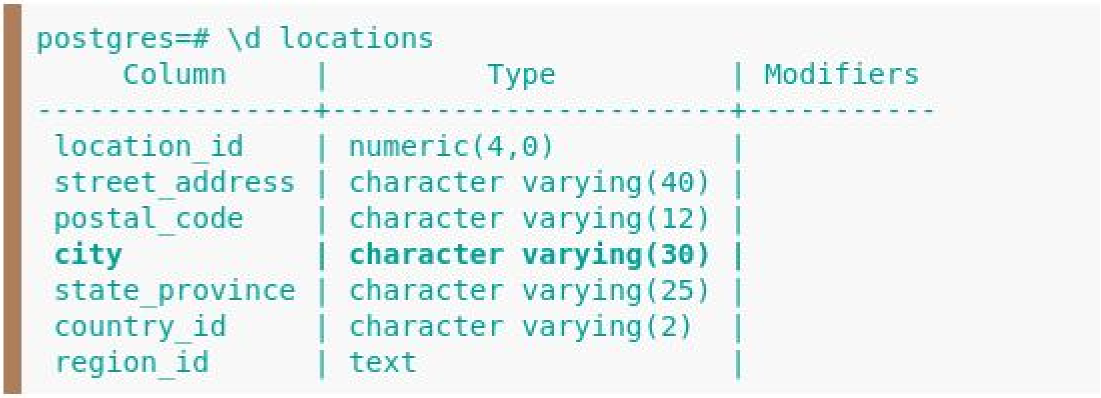
Here is the structure of the table locations.



```
postgres=# \d locations
     Column     |         Type          | Modifiers
----------------+-----------------------+-----------
 location_id    | numeric(4,0)          |
 street_address | character varying(40) |
 postal_code    | character varying(12) |
 city           | character varying(30) |
 state_province | character varying(25) |
 country_id     | character varying(2)  |
 region_id      | integer               |
```

4. Write a SQL statement to drop the column city from the table locations.

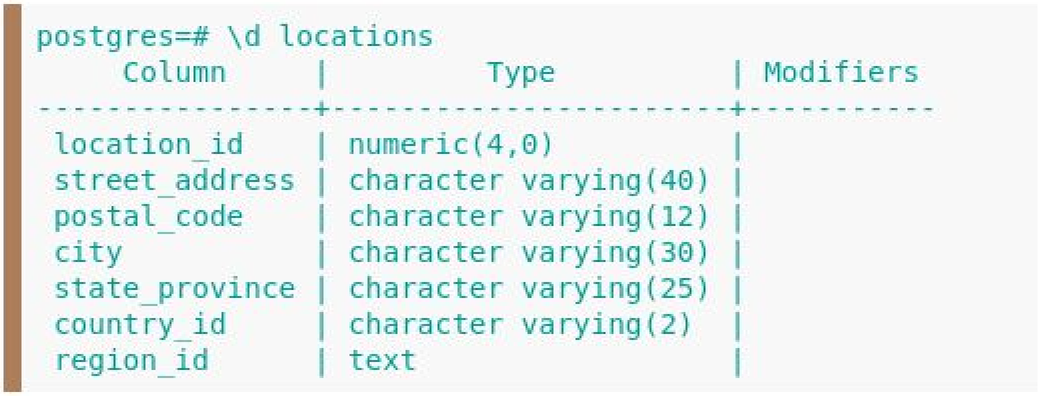
Here is the structure of the table locations.



```
postgres=# \d locations
     Column     |         Type          | Modifiers
----------------+-----------------------+-----------
 location_id    | numeric(4,0)          |
 street_address | character varying(40) |
 postal_code    | character varying(12) |
 city           | character varying(30) |
 state_province | character varying(25) |
 country_id     | character varying(2)  |
 region_id      | text                  |
```

5. Write a SQL statement to change the name of the column state_province to state, keeping the data type and size same.

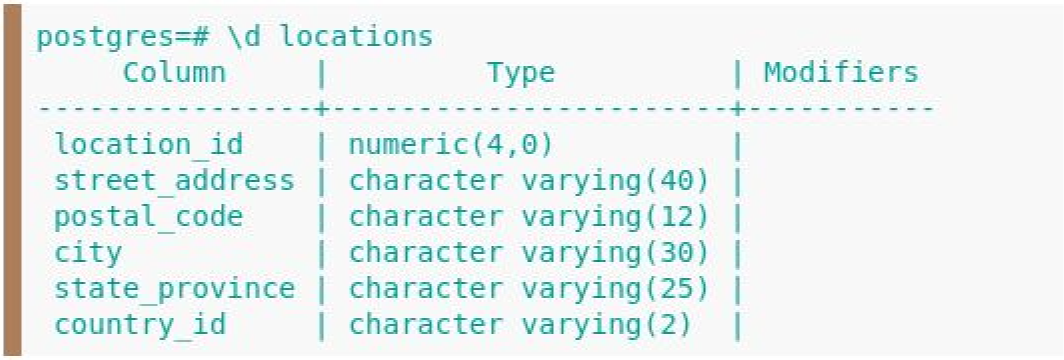
Here is the structure of the table locations.



```
postgres=# \d locations
     Column     |          Type          | Modifiers
----------------+------------------------+-----------
 location_id    | numeric(4,0)           |
 street_address | character varying(40) |
 postal_code    | character varying(12) |
 city           | character varying(30) |
 state_province | character varying(25) |
 country_id     | character varying(2)  |
 region_id      | text                   |
```

6. Write a SQL statement to add a primary key for the columns location_id in the locations table.

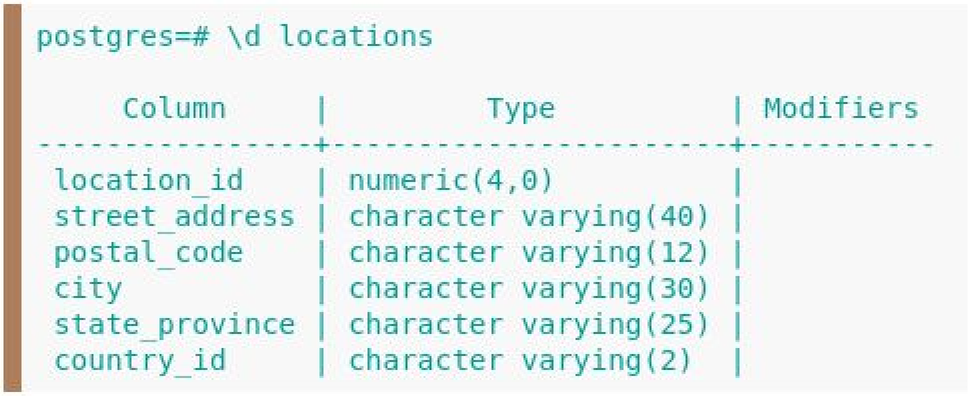
Here is the structure of the table locations.



```
postgres=# \d locations
     Column     |          Type          | Modifiers
----------------+------------------------+-----------
 location_id    | numeric(4,0)           |
 street_address | character varying(40) |
 postal_code    | character varying(12) |
 city           | character varying(30) |
 state_province | character varying(25) |
 country_id     | character varying(2)  |
```

7. Write a SQL statement to add a primary key for a combination of columns location_id and country_id.

Here is the structure of the table locations.

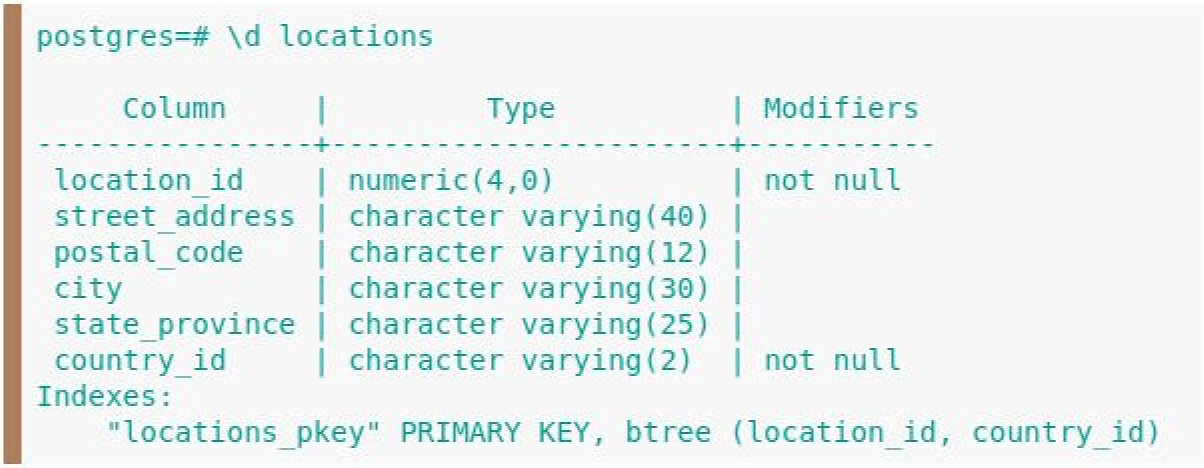


```
postgres=# \d locations

     Column     |          Type          | Modifiers
----------------+------------------------+-----------
 location_id    | numeric(4,0)           |
 street_address | character varying(40) |
 postal_code    | character varying(12) |
 city           | character varying(30) |
 state_province | character varying(25) |
 country_id     | character varying(2)  |
```

8. Write a SQL statement to drop the existing primary from the table locations on a combination of columns location_id and country_id.

Here is the structure of the table locations.

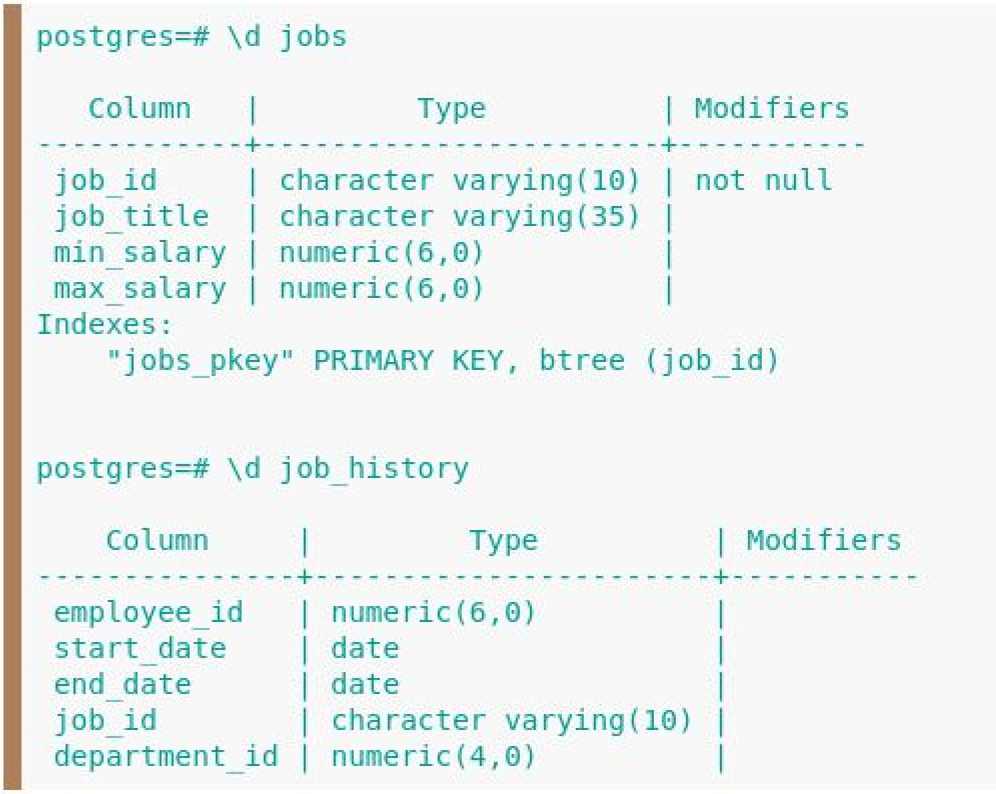


```
postgres=# \d locations

     Column     |         Type          | Modifiers
----------------+-----------------------+-----------
 location_id    | numeric(4,0)          | not null
 street_address | character varying(40) |
 postal_code    | character varying(12) |
 city           | character varying(30) |
 state_province | character varying(25) |
 country_id     | character varying(2)  | not null
Indexes:
    "locations_pkey" PRIMARY KEY, btree (location_id, country_id)
```

9. Write a SQL statement to add a foreign key on job_id column of job_history table referencing to the primary key job_id of jobs table.

Here is the structure of the table jobs and job_history.



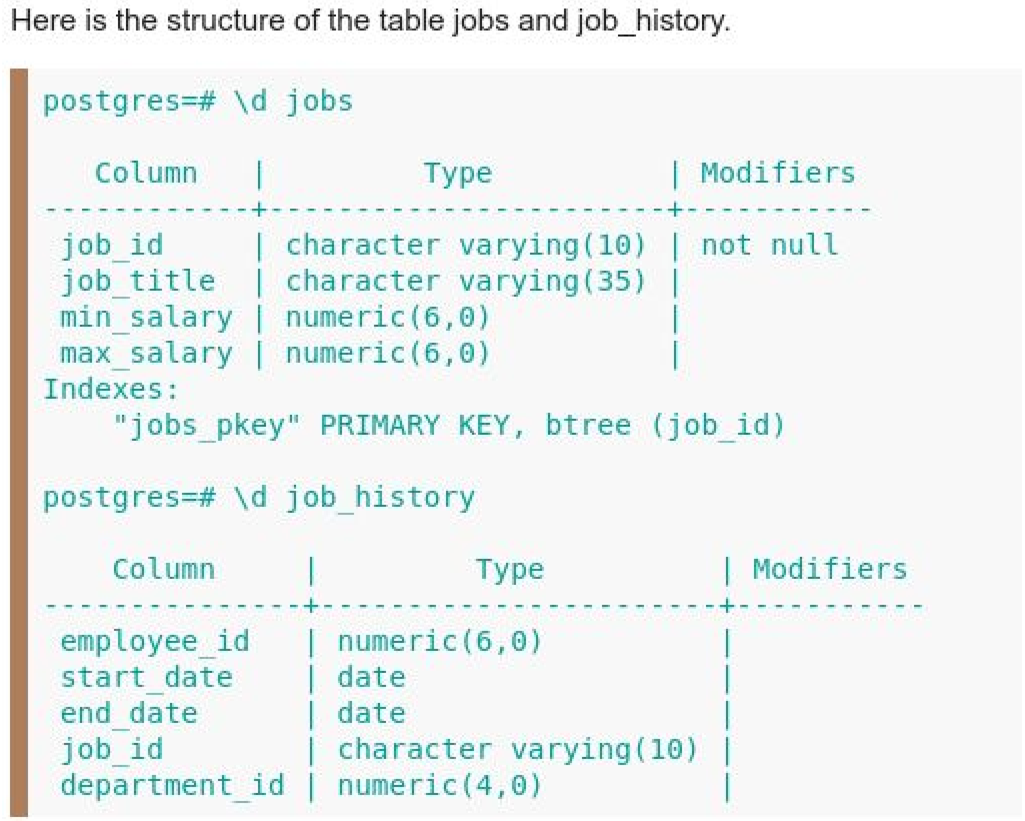
```
postgres=# \d jobs

   Column   |         Type          | Modifiers
------------+-----------------------+-----------
 job_id     | character varying(10) | not null
 job_title  | character varying(35) |
 min_salary | numeric(6,0)          |
 max_salary | numeric(6,0)          |
Indexes:
    "jobs_pkey" PRIMARY KEY, btree (job_id)


postgres=# \d job_history

    Column     |         Type          | Modifiers
---------------+-----------------------+-----------
 employee_id   | numeric(6,0)          |
 start_date    | date                  |
 end_date      | date                  |
 job_id        | character varying(10) |
 department_id | numeric(4,0)          |
```

10. Write a SQL statement to add a foreign key constraint named fk_job_id on job_id column of job_history table referencing to the primary key job_id of jobs table.



Here is the structure of the table jobs and job_history.

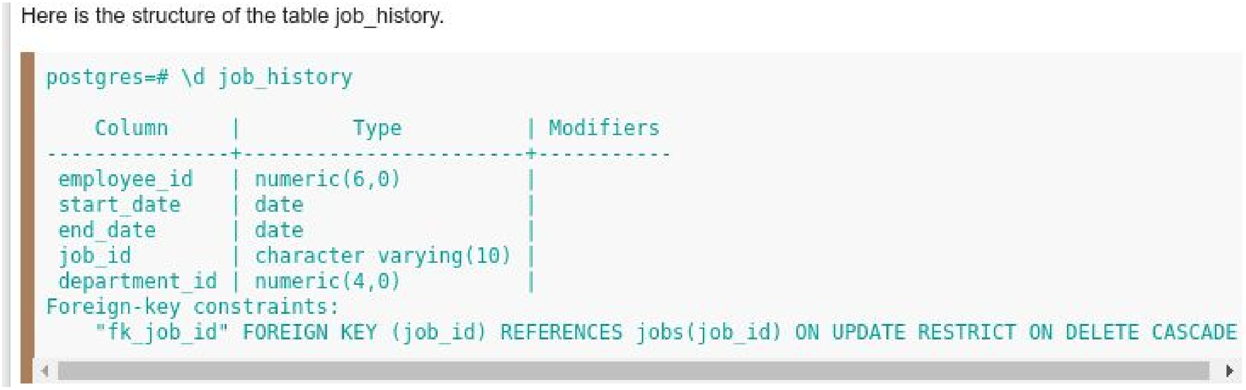
```
postgres=# \d jobs

   Column   |         Type          | Modifiers
------------+-----------------------+-----------
 job_id     | character varying(10) | not null
 job_title  | character varying(35) |
 min_salary | numeric(6,0)          |
 max_salary | numeric(6,0)          |
Indexes:
    "jobs_pkey" PRIMARY KEY, btree (job_id)

postgres=# \d job_history

    Column     |         Type          | Modifiers
---------------+-----------------------+-----------
 employee_id   | numeric(6,0)          |
 start_date    | date                  |
 end_date      | date                  |
 job_id        | character varying(10) |
 department_id | numeric(4,0)          |
```

11. Write a SQL statement to drop the existing foreign key fk_job_id from job_history table on job_id column which is referencing to the job_id of jobs table.



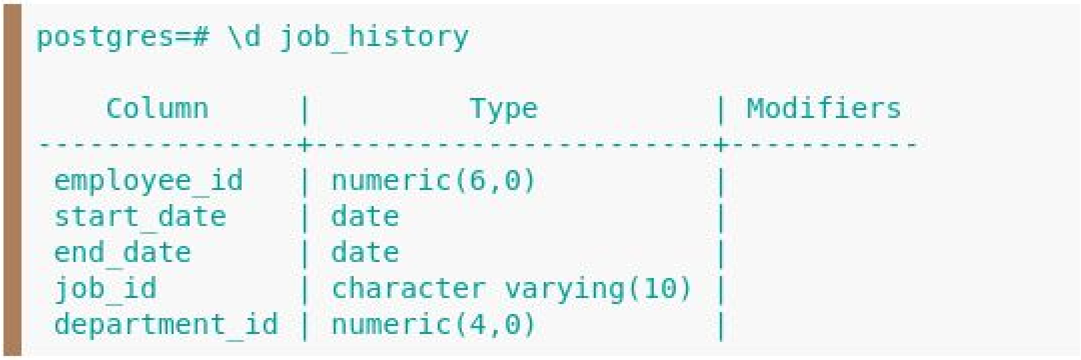
Here is the structure of the table job_history.

```
postgres=# \d job_history

    Column     |         Type          | Modifiers
---------------+-----------------------+-----------
 employee_id   | numeric(6,0)          |
 start_date    | date                  |
 end_date      | date                  |
 job_id        | character varying(10) |
 department_id | numeric(4,0)          |
Foreign-key constraints:
    "fk_job_id" FOREIGN KEY (job_id) REFERENCES jobs(job_id) ON UPDATE RESTRICT ON DELETE CASCADE
```

12. Write a SQL statement to add an index named index_job_id on job_id column in the table job_history.

Here is the structure of the table job_history.

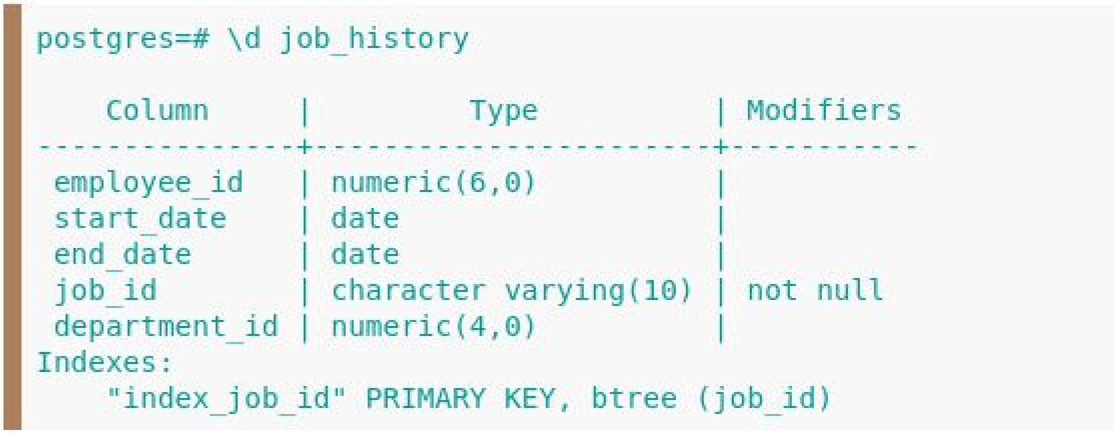


```
postgres=# \d job_history

    Column     |         Type          | Modifiers
---------------+-----------------------+-----------
 employee_id   | numeric(6,0)          |
 start_date    | date                  |
 end_date      | date                  |
 job_id        | character varying(10) |
 department_id | numeric(4,0)          |
```

13. Write a SQL statement to drop the index indx_job_id from job_history table.

Here is the structure of the job_history and index file of the table job_history.



```
postgres=# \d job_history

    Column     |         Type          | Modifiers
---------------+-----------------------+-----------
 employee_id   | numeric(6,0)          |
 start_date    | date                  |
 end_date      | date                  |
 job_id        | character varying(10) | not null
 department_id | numeric(4,0)          |
Indexes:
    "index_job_id" PRIMARY KEY, btree (job_id)
```