

1. Write a SQL statement to create a simple table countries including columns country\_id, country\_name and region\_id.

```
create table countries(country_id integer, country_name char(20), region_id int);
```

Table "public.countries"

Column	Type	Collation	Nullable	Default
country_id	integer			
Country_name	character(20)			
region_id	integer			

2. Write a SQL statement to create a simple table countries including columns country\_id, country\_name and region\_id which already exist.

```
create table if not exists countries(country_id integer, country_name char(20), region_id int);
```

relation "countries" already exists, skipping

3. Write a SQL statement to create the structure of a table dup\_countries similar to countries.

```
create table dup_countries as table countries with no data;
```

```
cdac1=# \d dup_countries
```

Table "public.dup_countries"				
Column	Type	Collation	Nullable	Default
country_id	integer			
Country_name	character(20)			
region_id	integer			

4. Write a SQL statement to create a duplicate copy of countries table including structure and data by name dup\_countries.

```
create table dup_countries as table countries
```

```
cdac1=# select * from dup2_countries;
```

country_id	country_name	region_id
1	India	1
2	australia	2
3	USA	3

5. Write a SQL statement to create a table named **countries**, including **country\_id**, **country\_name** and **region\_id** and make sure that no duplicate data against column **country\_id** will be allowed at the time of insertion.

```
create table if not exists countries2(country_id integer primary key, country_name char(20), region_id serial);
```

```
cdac1=# \d countries2
```

```
Table "public.countries2"
```

Column	Type	Collation	Nullable	Default
country_id	integer		not null	
country_name	character(20)			
region_id	integer		not null	

```
nextval('countries2_region_id_seq'::regclass)
```

Indexes:

```
"countries2_pkey" PRIMARY KEY, btree (country_id)
```