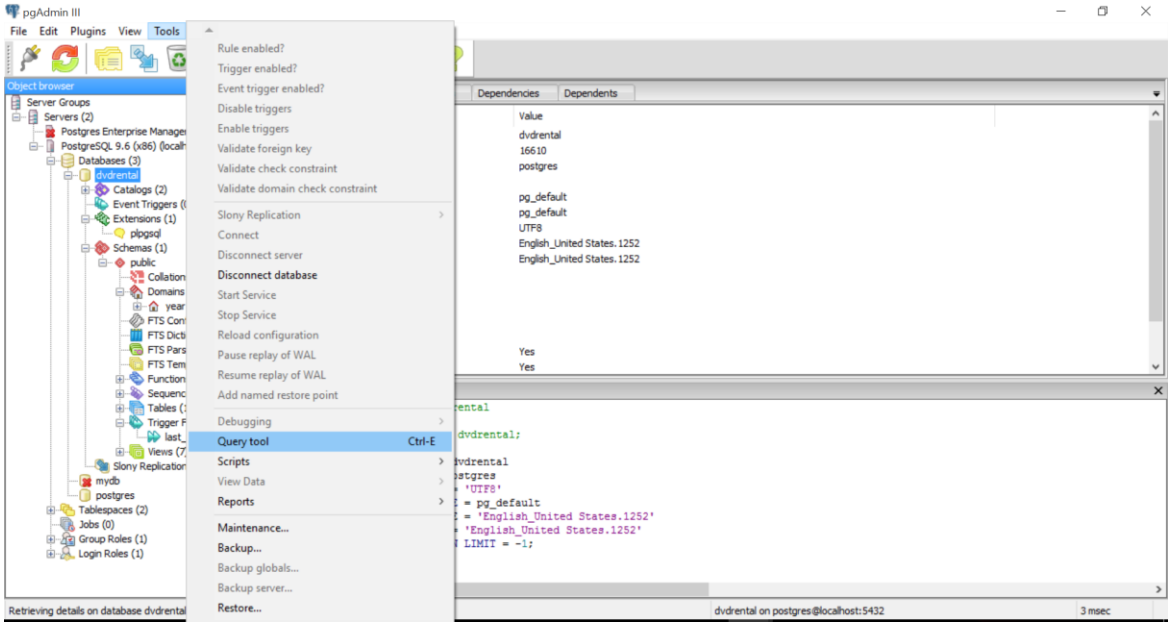
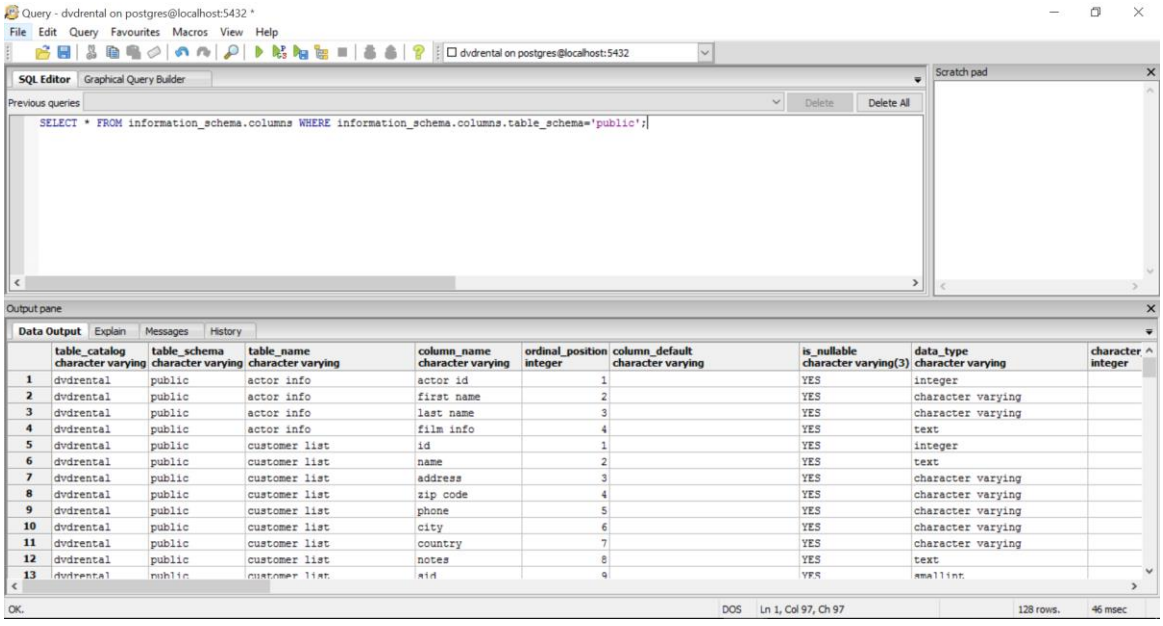
Tutorial 3

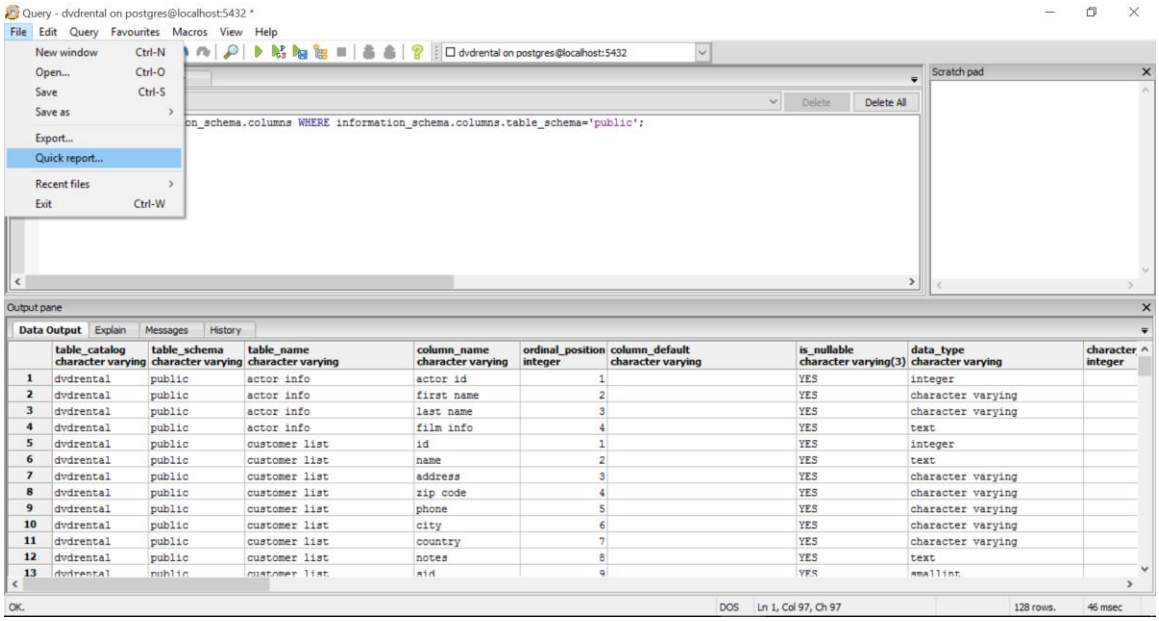
1. To obtain the information needed about base relations (from the data dictionary and the definition of the relations), open pgAdmin III, stand on the node “dvdrental”, and go to “Tools” menu and click on “Query tool”. Then, apply some SQL queries provided in the following link: www.postgresql.org/docs/current/static/information-schema.html

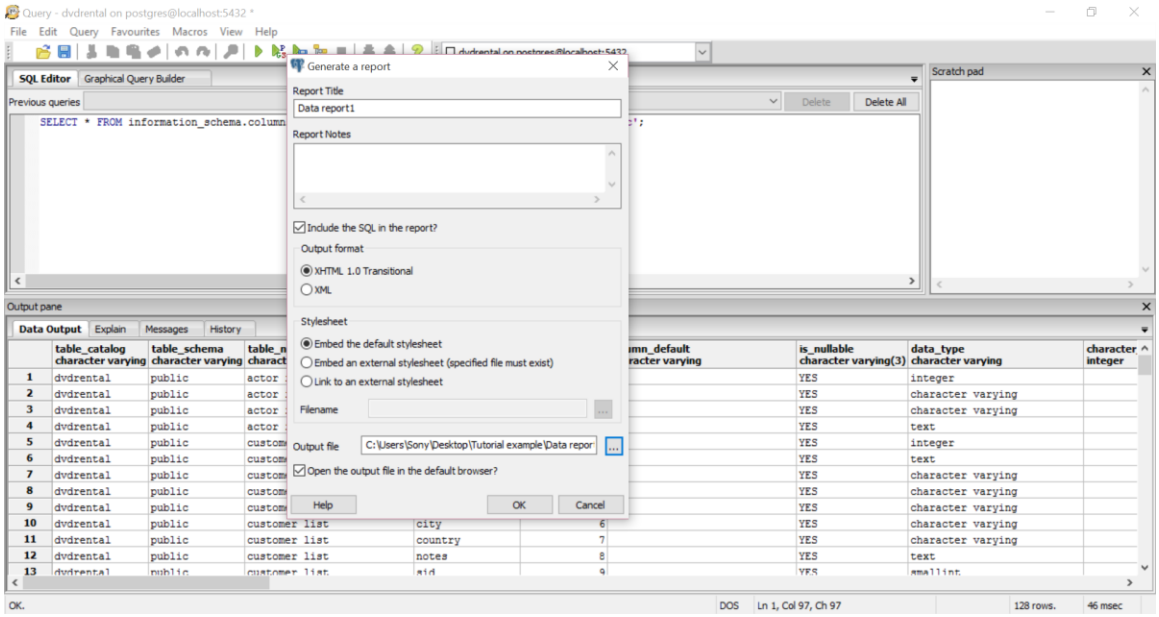


2. Once the SQL query editor opens, type the following SQL statement: “SELECT * FROM information_schema.columns WHERE information_schema.columns.table_schema='public';” and observe the results in the output pane



3. You can save the output in a Web-like format by doing the following steps:





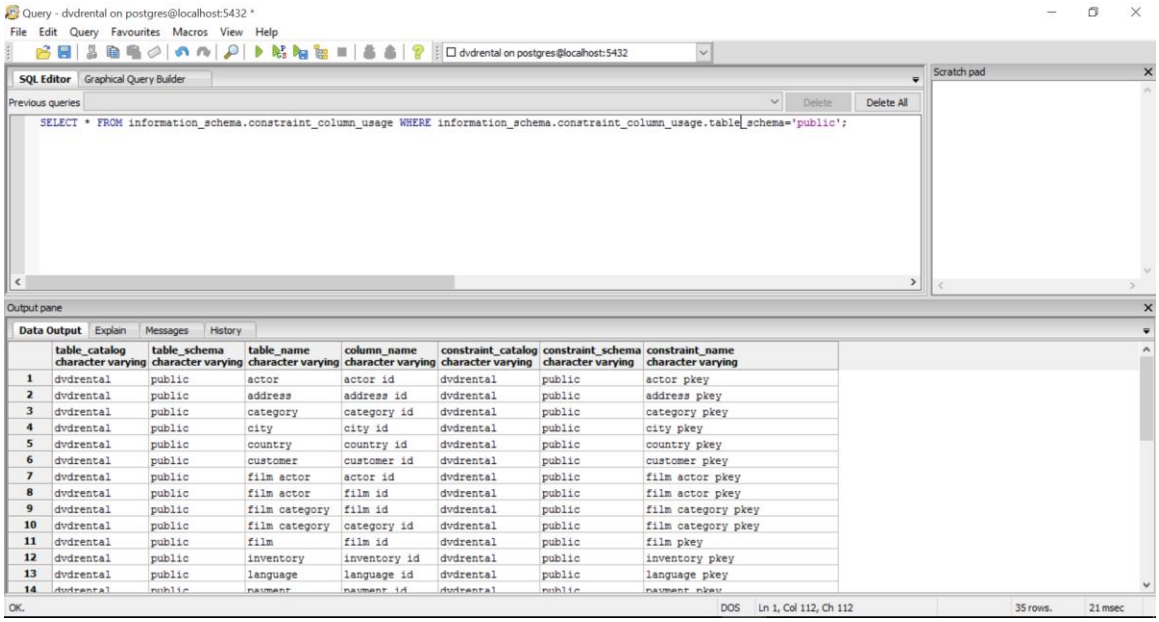
Data report1

Generated: 7/27/2017 9:37:27 PM
Database: dvdrental on postgres@localhost:5432

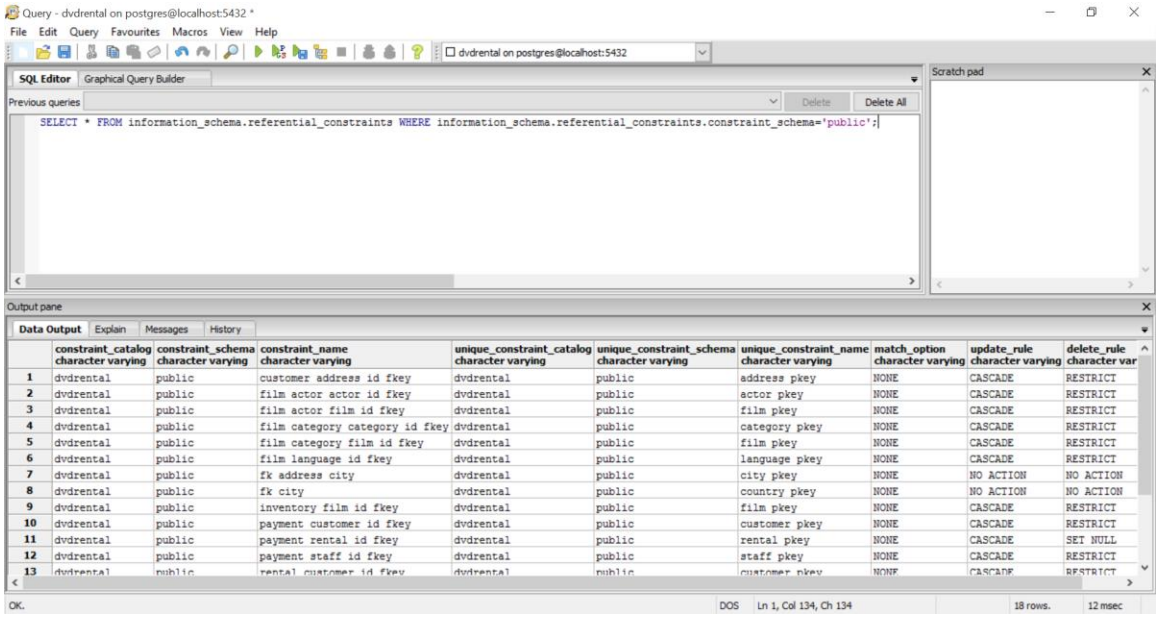
Query results

table_catalog	table_schema	table_name	column_name	ordinal_position	column_default	is_nullable	data_type	character_maximum_len
dvdrental	public	actor_info	actor_id	1		YES	integer	
dvdrental	public	actor_info	first_name	2		YES	character varying	45
dvdrental	public	actor_info	last_name	3		YES	character varying	45
dvdrental	public	actor_info	film_info	4		YES	text	
dvdrental	public	customer_list	id	1		YES	integer	
dvdrental	public	customer_list	name	2		YES	text	
dvdrental	public	customer_list	address	3		YES	character varying	50
dvdrental	public	customer_list	zip code	4		YES	character varying	10
dvdrental	public	customer_list	phone	5		YES	character varying	20
dvdrental	public	customer_list	city	6		YES	character varying	50
dvdrental	public	customer_list	country	7		YES	character varying	50
dvdrental	public	customer_list	notes	8		YES	text	
dvdrental	public	customer_list	sid	9		YES	smallint	
dvdrental	public	film_list	fid	1		YES	integer	
dvdrental	public	film_list	title	2		YES	character varying	255
dvdrental	public	film_list	description	3		YES	text	
dvdrental	public	film_list	category	4		YES	character varying	25
dvdrental	public	film_list	price	5		YES	numeric	
dvdrental	public	film_list	length	6		YES	smallint	
dvdrental	public	film_list	rating	7		YES	USER-DEFINED	
dvdrental	public	film_list	actors	8		YES	text	
dvdrental	public	nicer_but_slower_film_list	fid	1		YES	integer	
dvdrental	public	nicer_but_slower_film_list	title	2		YES	character varying	255
dvdrental	public	nicer_but_slower_film_list	description	3		YES	text	

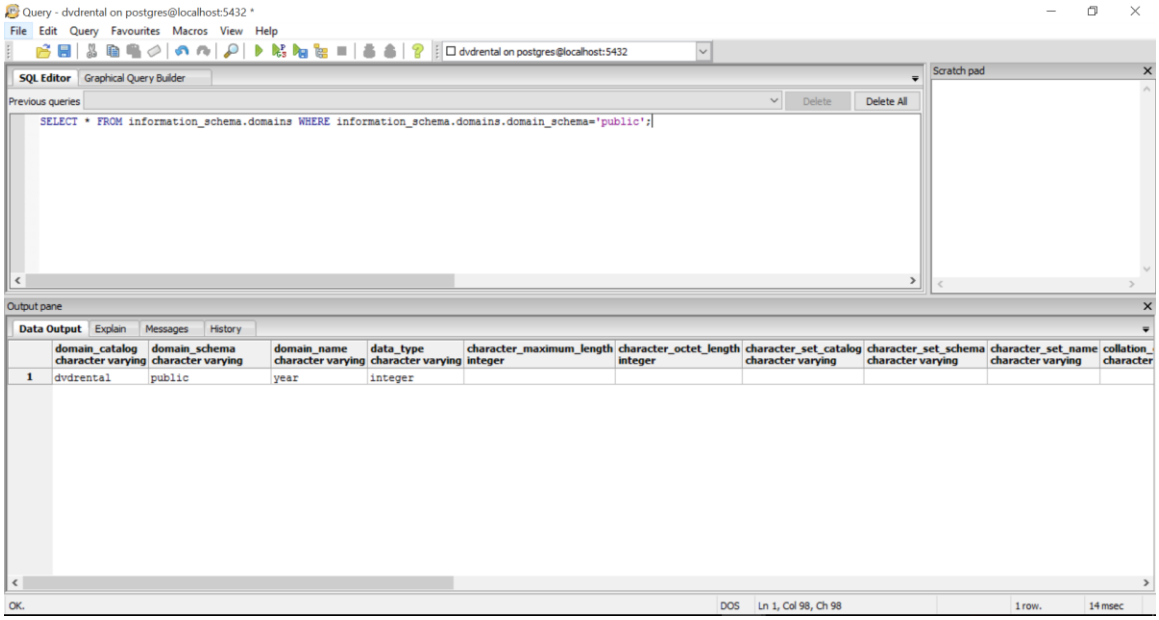
4. To obtain information about primary key and foreign keys, use the following SQL statement: “SELECT * FROM information_schema.constraint_column_usage WHERE information_schema.constraint_column_usage.table_schema='public';”



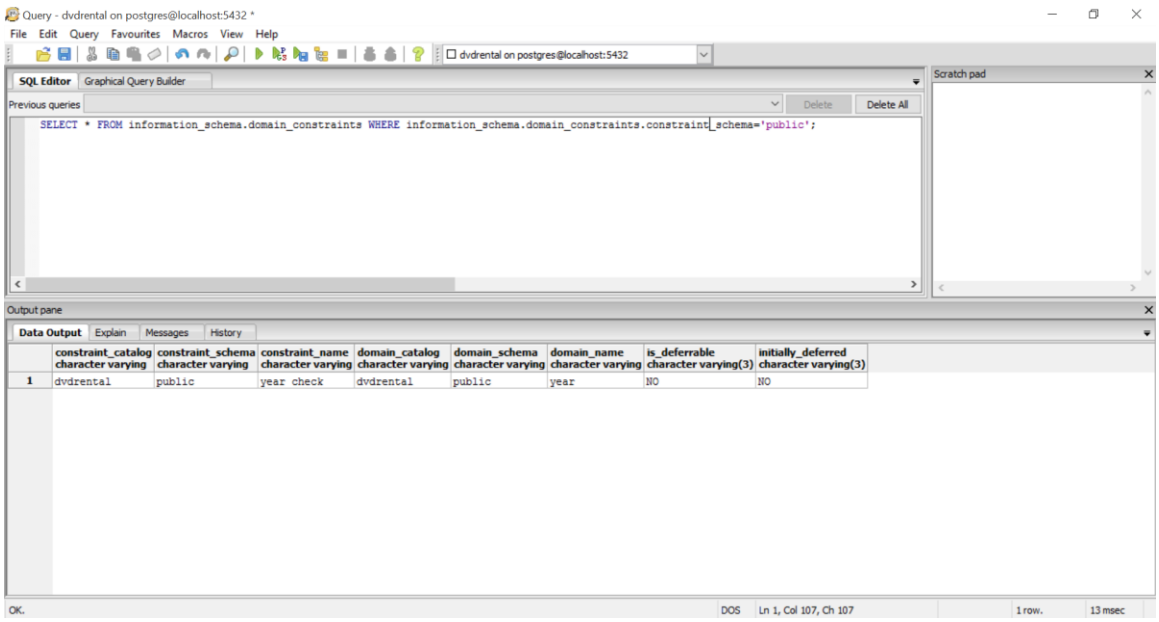
5. To obtain information about the referential integrity constraints for foreign keys, try this SQL statement: “SELECT * FROM information_schema.referential_constraints WHERE information_schema.referential_constraints.constraint_schema='public';”



6. To obtain information about the attributes' domain and constraints on that domain, there are two SQL statements to execute:
    a. "SELECT * FROM information_schema.domains WHERE information_schema.domains.domain_schema='public';"



    b. "SELECT * FROM information_schema.domain_constraints WHERE information_schema.domain_constraints.constraint_schema='public';"



7. To obtain information about general constraints other than those on primary keys and foreign keys, run the following SQL statement: “SELECT * FROM information_schema.check_constraints WHERE information_schema.check_constraints.constraint_schema='public';”