COMP9120 Database Management Systems

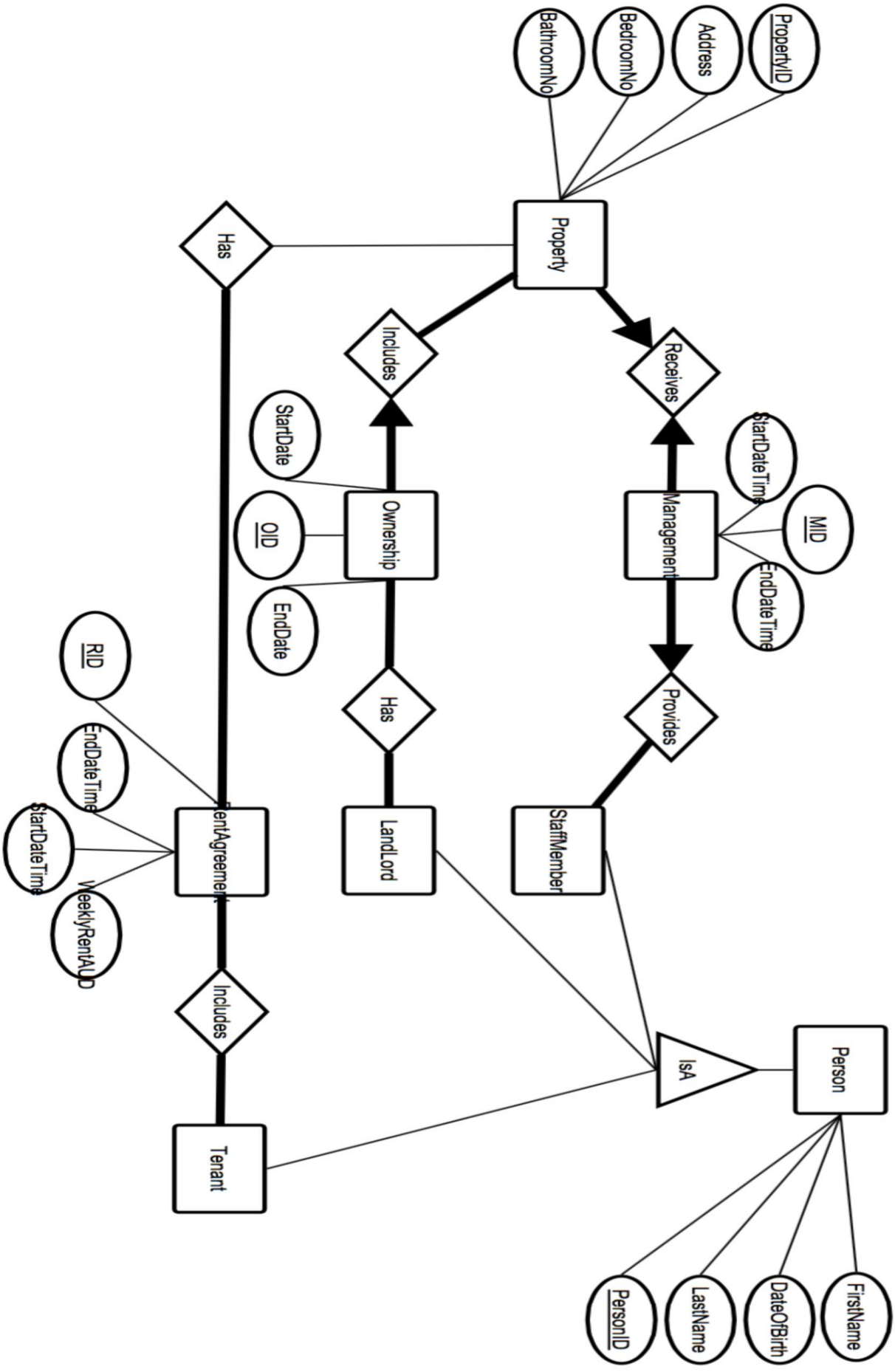
Tutorial Week 2

Conceptual Design with Entity-Relationship Diagrams

Exercise 1. ER Diagram Analysis

Consider a real-estate company, which manages rental properties. Each property has an address, several bedrooms, bathrooms, and car spaces. A rental property may have several rental agreements. A rental agreement details the weekly rent that is agreed upon with one or more tenants for a property for a given period. For each of these tenants, the details that need to be recorded include first name, last name, and date of birth. At any point in time, each staff member must be responsible for managing at least one property, where a property can be managed by multiple staff members and must be managed by at least one staff member at any point in time. Staff members also have a first name, last name, and date of birth. At any point in time a property is owned by one or more landlords. The ownership of each property may also change over time.

The E-R diagram shown below is one attempt at modelling this domain. Consider what errors you can identify in this model.


PropertyID
Address
BedroomNo
BathroomNo
Property
Has
Includes
Receives
StartDate
OID
EndDate
Ownership
StartDateTime
MID
EndDateTime
Management
RID
EndDateTime
StartDateTime
WeeklyRentAUD
RentAgreement
Has
Provides
LandLord
StaffMember
Includes
IsA
Person
Tenant
PersonID
LastName
DateOfBirth
FirstName

Exercise 2. ER Diagram Creation

This question asks you to develop an ER diagram from the narrative below.

The information system, which you will analyse, aims to assist a student recreational skiing society, which needs to deal with members and to organise events for members. At present the system is paper-based, but in future it is hoped to develop a computer-based system.

The society secretary uses the system to record information when a new member joins the society: the information includes the member's name, term-time address, vacation address, phone number, email, and the different varieties of skiing in which they are interested (for example, Peter Williams may have interest in the beginner's cross-country, the intermediate downhill, and the expert bobsled). Usually, members join for a period of one winter season by paying a fee, and at the end of that time the secretary can extend their membership if they pay another fee; alternatively, some members choose to join for longer periods (the fee they paid is then reduced, compared to paying for seasons one by one). The event manager uses the system to arrange a ski trip, which occurs over a particular period (such as the long weekend of October 4 to October 6 inclusive), includes accommodation at a particular address, and also has one or more sessions of various varieties of skiing, each located on a particular area of a ski slope; for example, one meeting may involve three sessions of intermediate downhill held on Piste X, Piste Y and Piste Z respectively, and one session of beginner's downhill held on Piste X. The event manager schedules each session with the appropriate number of participants from among the members who are interested in that variety of skiing. The cost of a meeting for the society depends on the number of members who are involved (as the accommodation fee is charged per person) as well as on the number of sessions, and the variety of each (because each different session requires booking an appropriate area of the ski slope). The club treasurer uses the system to obtain reports on the total cost of the trips held during each season, and on the total fee income from memberships for that season.

Exercise 3. Setting Up PostgreSQL Account

You can remotely access the PostgreSQL server that is maintained by School of CS via pgAdmin. Note that, if you want to access PostgreSQL server from outside the university network, you will need to first connect securely to the university network using VPN – refer to the following link: https://sydneyuni.service-now.com/sm?id=kb_article_view&sys_kb_id=c0bf9bd6db41b3485beaf9b7f49619a2&sysparm_tsqueryId=f90a62cbdb937f44c8a5773c349619f2&sysparm_rank=7

Your connection information to the PostgreSQL server maintained by SCS is as follows:

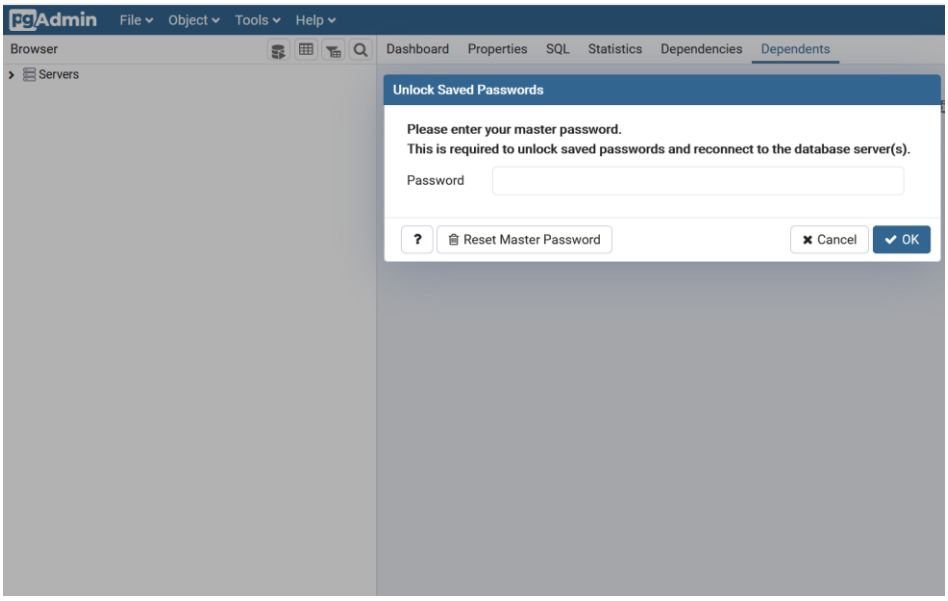
Host: soit-db-pro-2.ucc.usyd.edu.au

DB: y22s1c9120_UNIKEY (replace UNIKEY with your unikey; this is your Database)

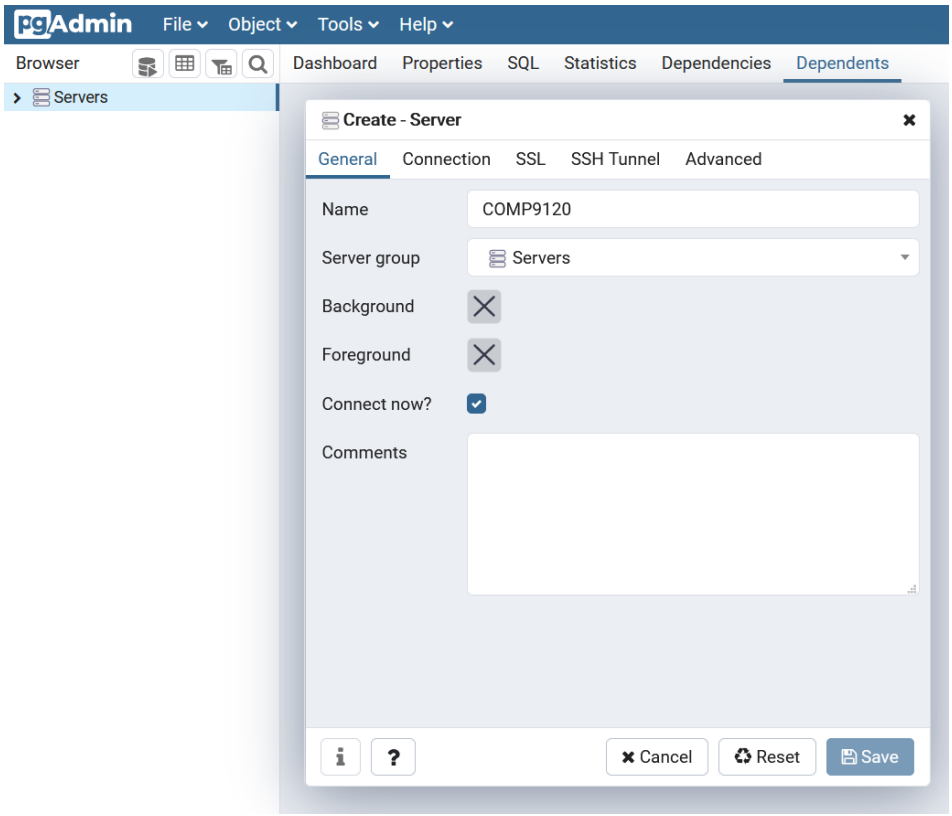
Username: y22s1c9120_UNIKEY (replace UNIKEY with your unikey; the same as your DB)

Password: random password sent to your email

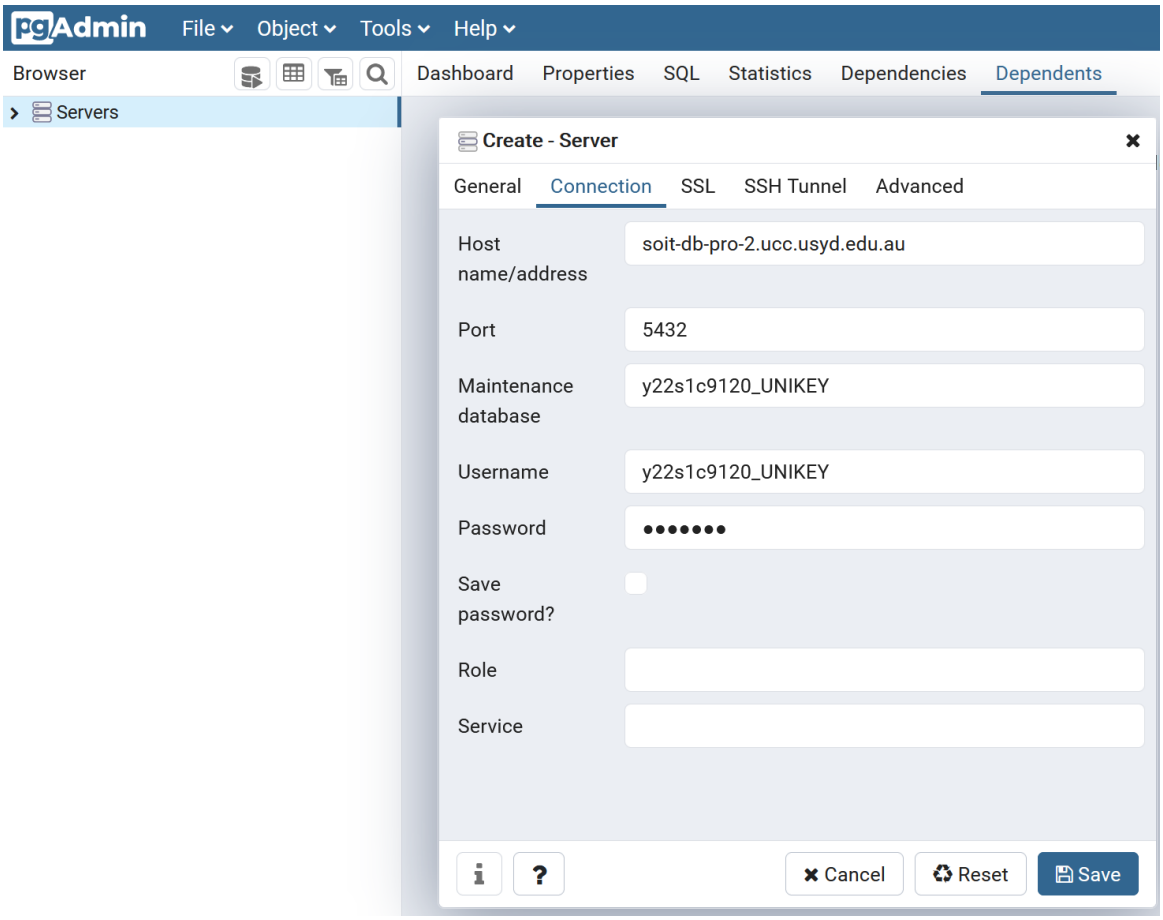
1. Start pgAdmin 4. You should see a screen like the one shown in the next picture. Type in the master password for pgAdmin and click OK.



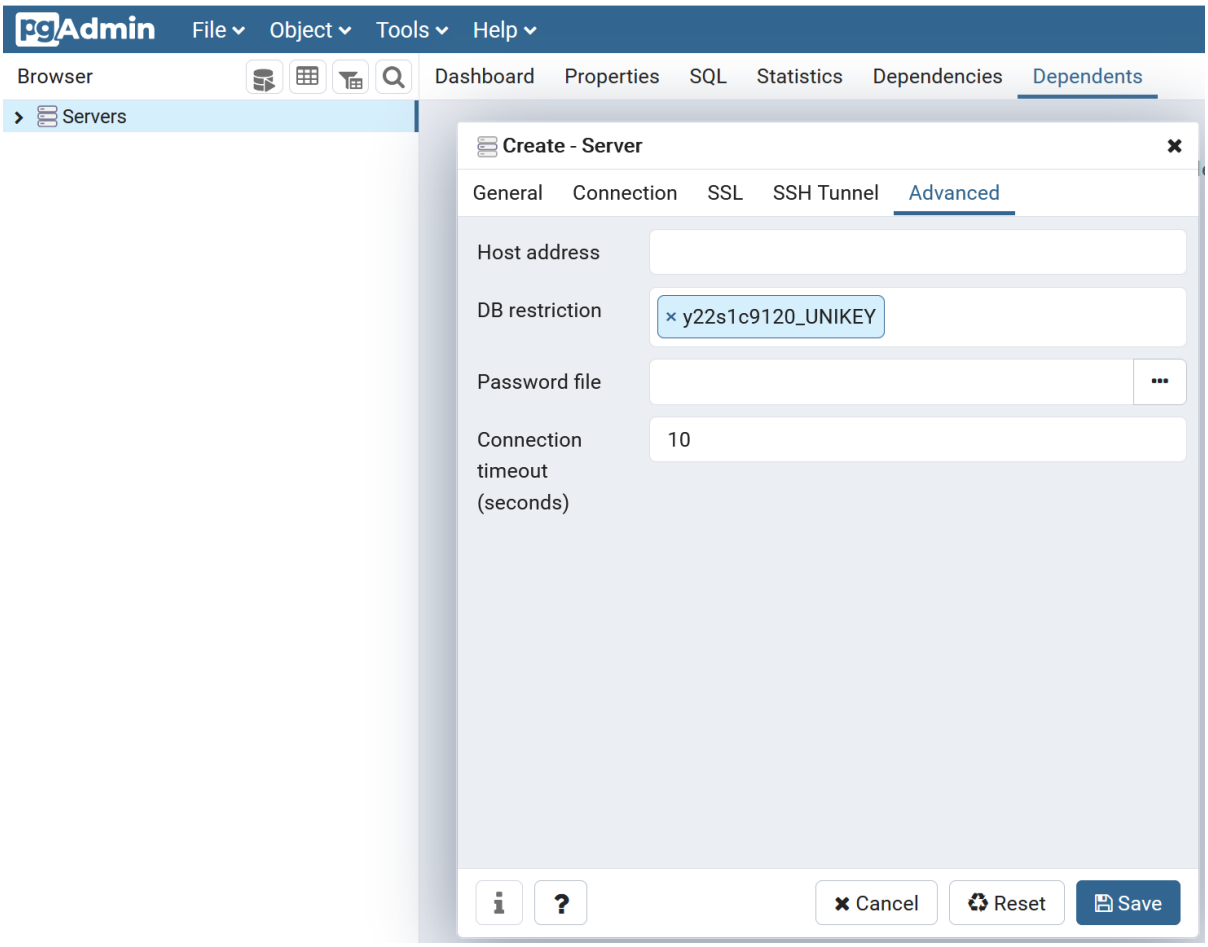
2. Right-click “Servers->Create->Server”. You should see the following screen. Type in Name (this can be anything you like, e.g. COMP9120)



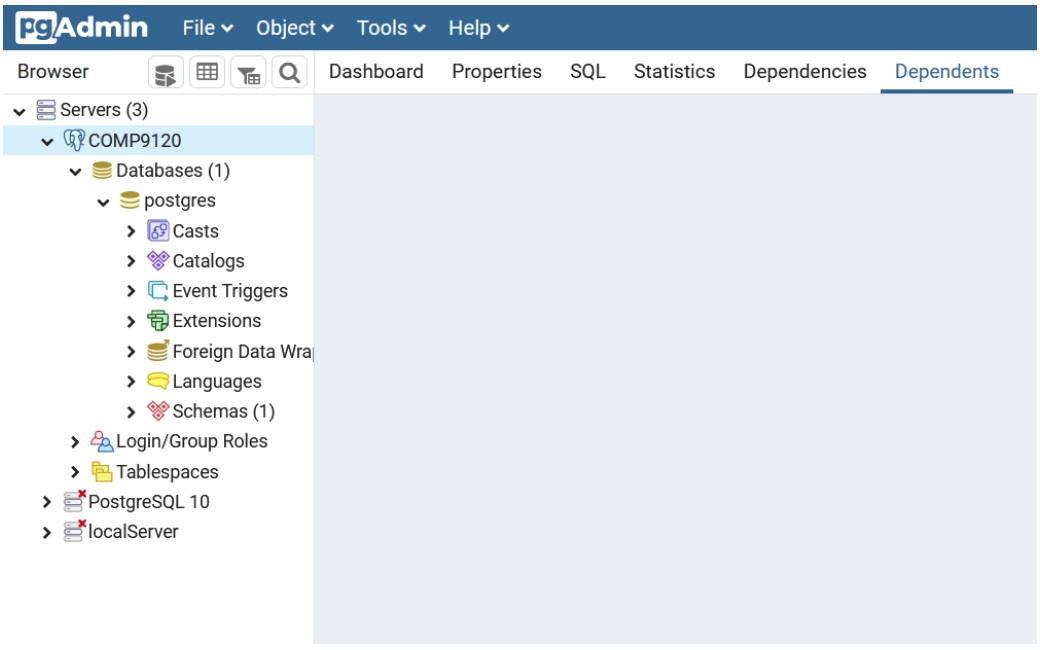
3. Click the tab “Connection”. Type in Host name/address, Maintenance database, Username, Password, as per your connection information above.



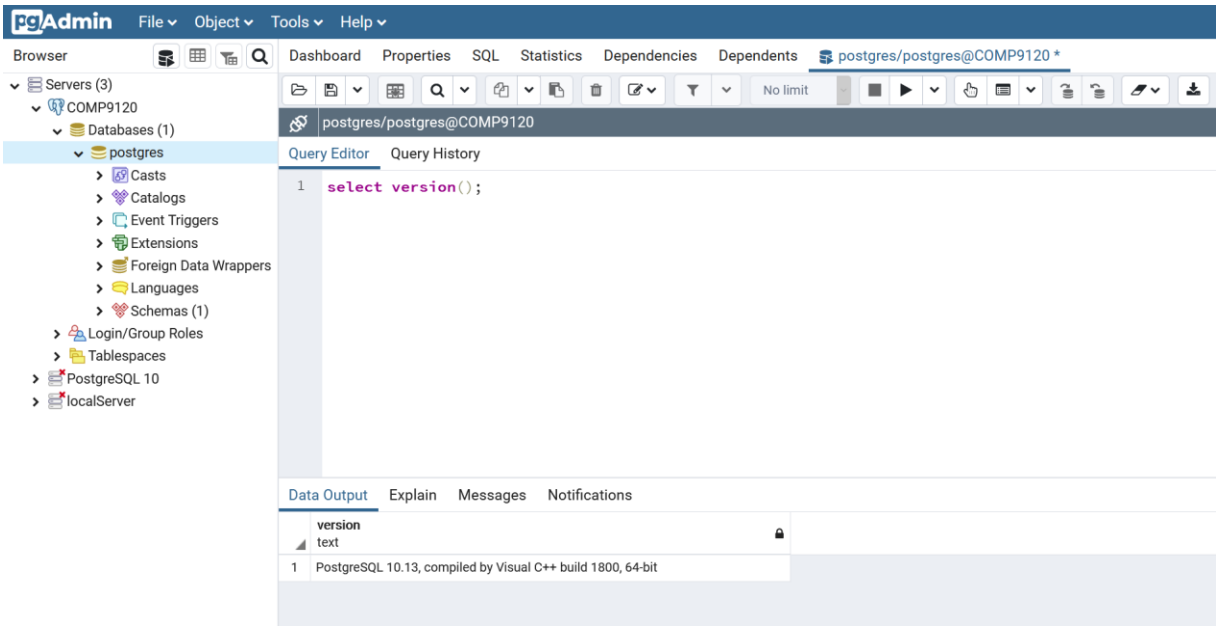
4. Click the tab “Advanced”, and copy the name of your DB onto the field “DB restriction”.



5. Click “Save”. You should see the following screen:



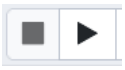
6. Select your Database, and right-click your Database -> “Query Tool”. You should see the following screen that allows you to write and execute SQL commands



7. Change your password by executing the following SQL command:

ALTER USER y22s1c9120_UNIKEY PASSWORD ‘pwd’;

Replace UNIKEY with your own unikey, and pwd with a new password that you wish to set for yourself. **Please remember your new password as you will need it later in this course.**

Click  “Execute/Refresh” or F5.

You should then see “query returned successfully in ** msec.” displayed on the Messages tab in the Output panel, meaning your password was changed successfully!