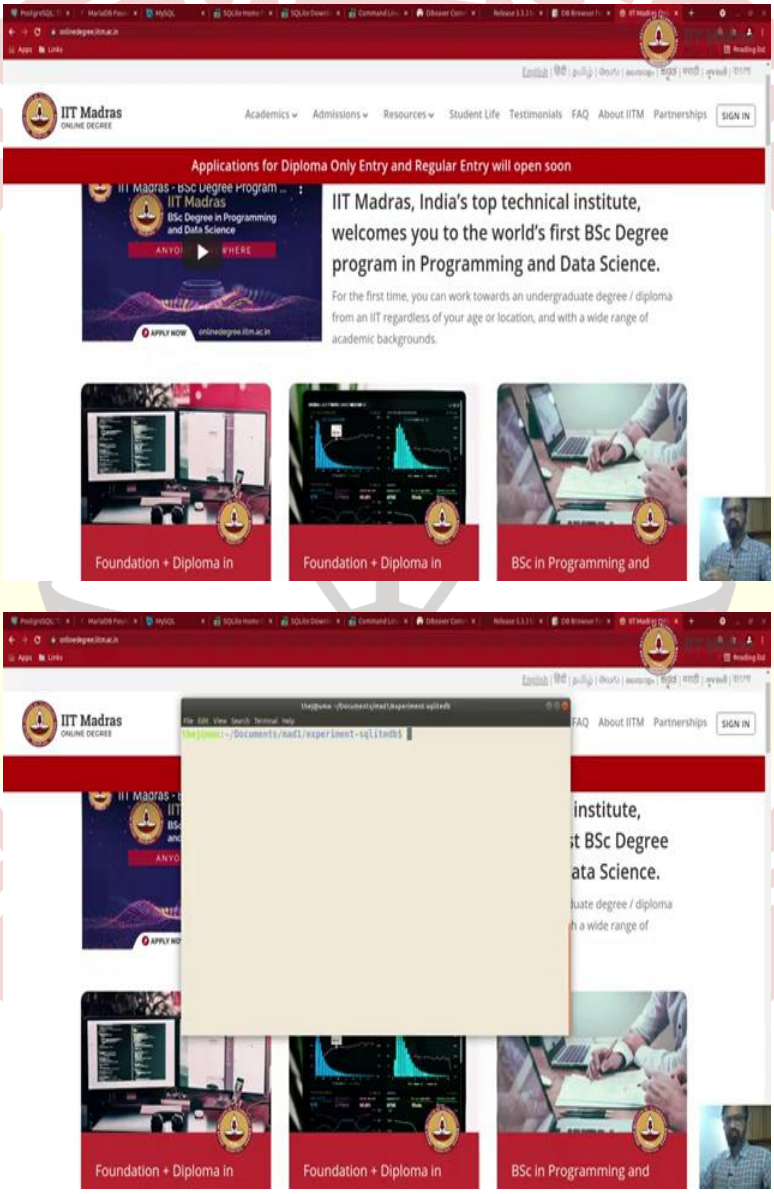
IIT Madras

ONLINE DEGREE

Modern Application Development - I
Professor Thejesh G N
Software Consultant
Bachelor of Science Degree
Indian Institute of Technology, Madras
How to create relational database using SQLite?

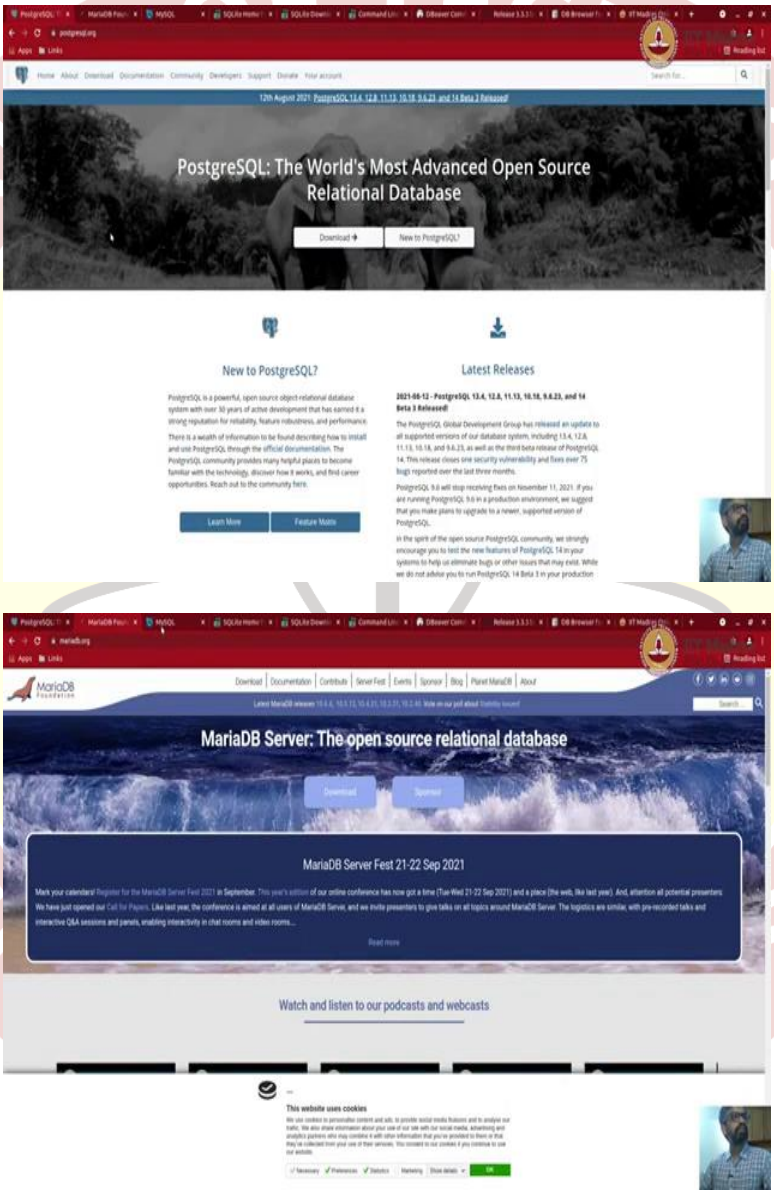
(Refer Slide Time: 0:18)

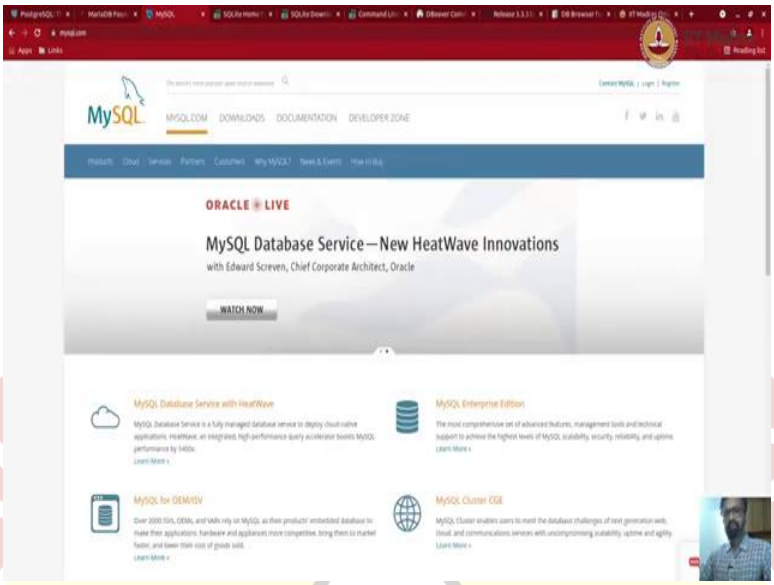


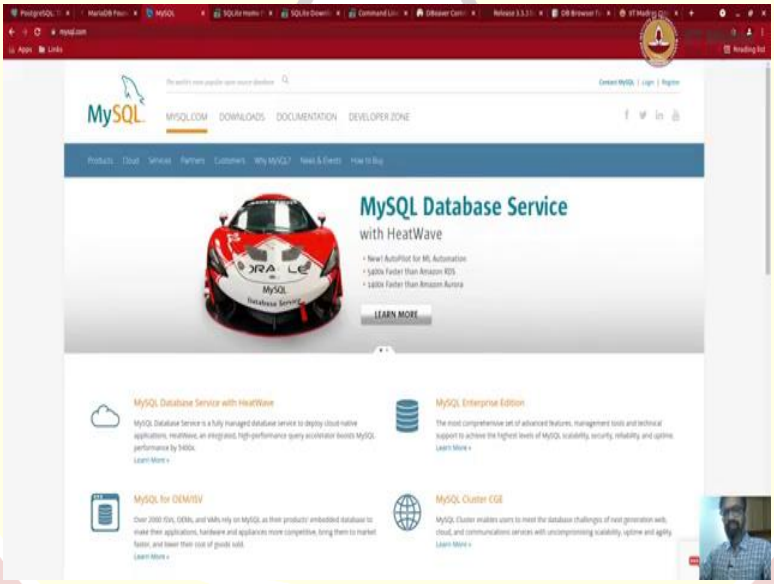
Welcome to the modern application development screencast. In this short screencast, we will learn, how to create and explore a relational database using SQLite. Before, we start open the following applications on your desktop.

So, that you can work along with me, a browser, chrome is fine, terminal the one which comes with your operating system is fine, it would look like this. I have already created a folder called experiment-sqlitedb. So, that we can explore it, or experiment in that folder. There are many open source relational databases available in the market today.

(Refer Slide Time: 1:02)

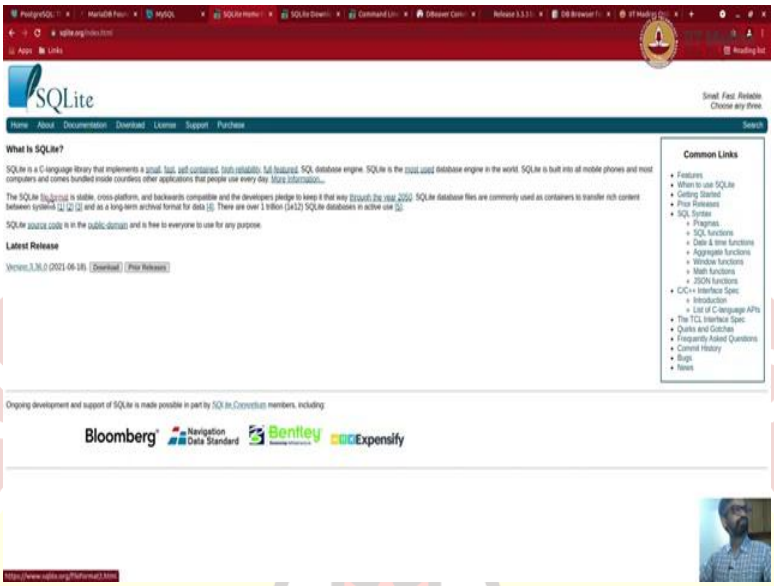






For example, PostgreSQL, MariaDB, MySQL, they are all production ready good databases, they all use client server model, on the other side there is SQLite.

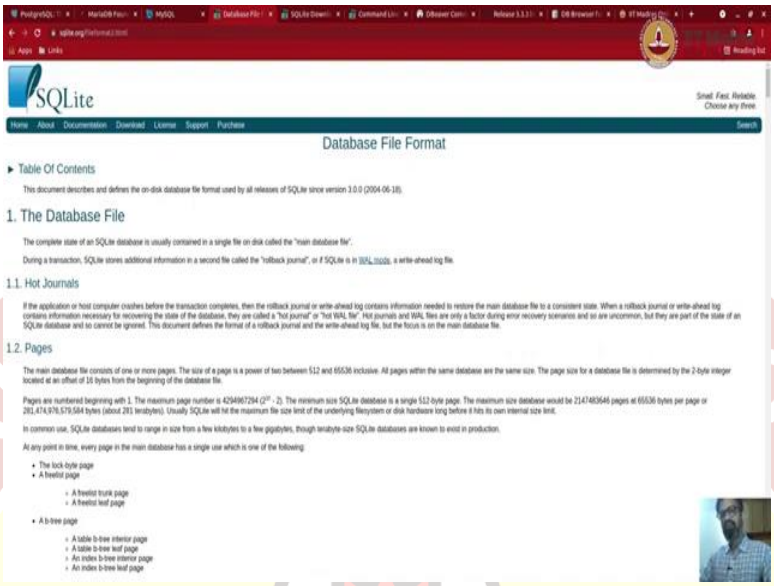
(Refer Slide Time: 1:18)



SQLite is a self-contained, full-featured database contained in a C library, as compared to many other databases that we saw earlier. SQLite does not use a client server model. It is an embedded database, it does not need a separate process for the server to run, it runs within the application. SQLite stores the entire database, definitions, tables, indices, and the data as a single cross platform file, on the machine, just like any other file.

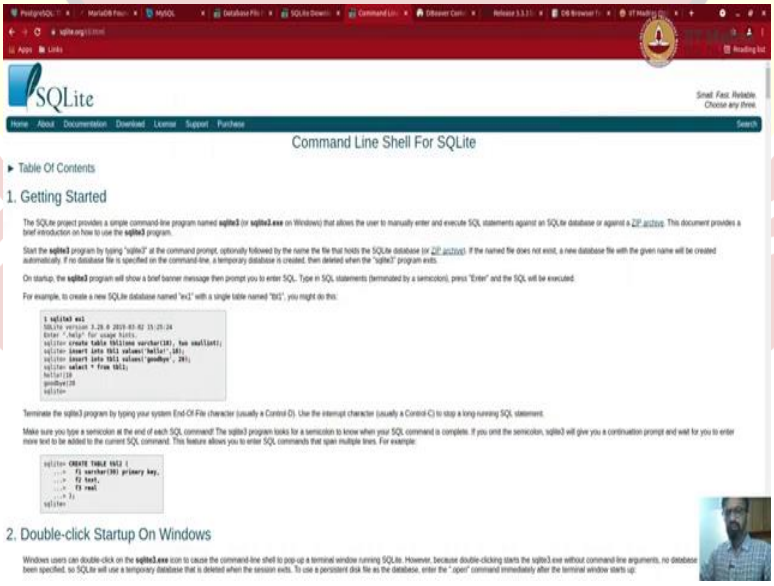
It is a well-defined file format currently in its version three. If you want to know about the file format, you can go to SQLite website, click on this link file format, and you can read about it.

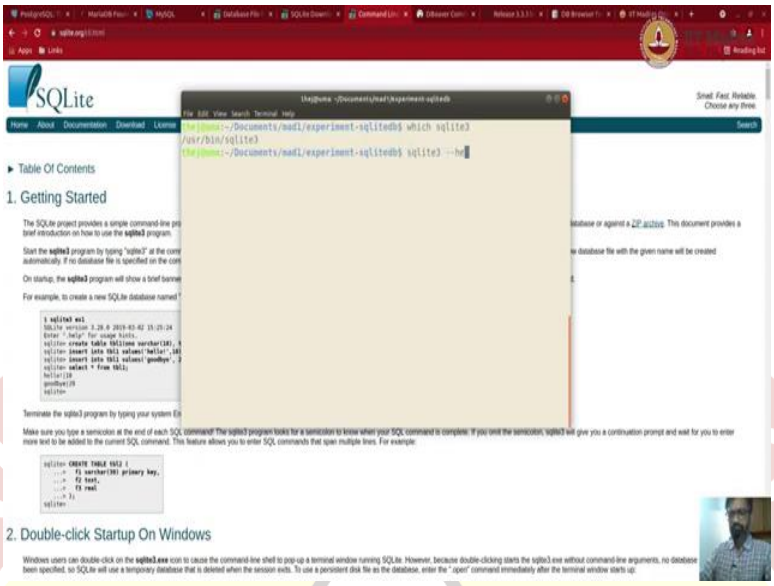
(Refer Slide Time: 2:10)

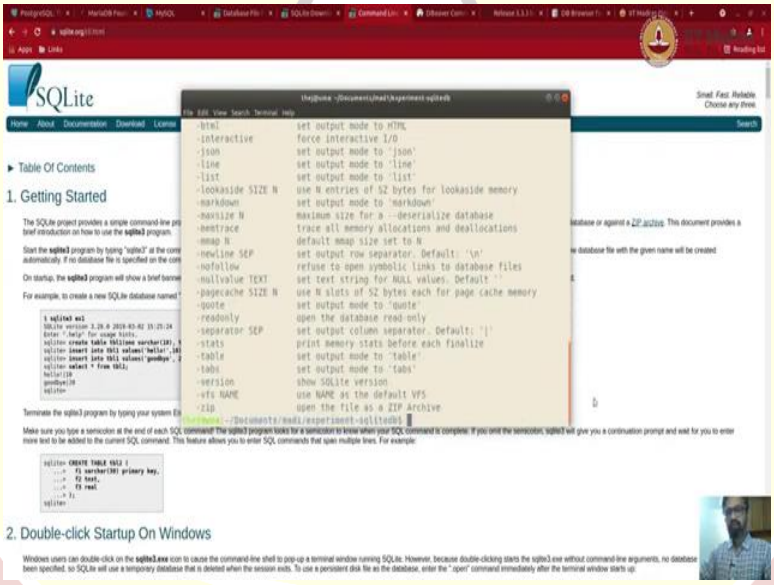


Usually you would see, this file with an extension .db, .sqlite, or .sqlite3. Currently since it is in version 3, we generally end up using .sqlite3. And since it is a database in the file format, it can be handled just like any other file, which means if you move the file, the database moves, if you delete the file, database is deleted. If you copy and backup the file, it is backed up.

(Refer Slide Time: 2:49)

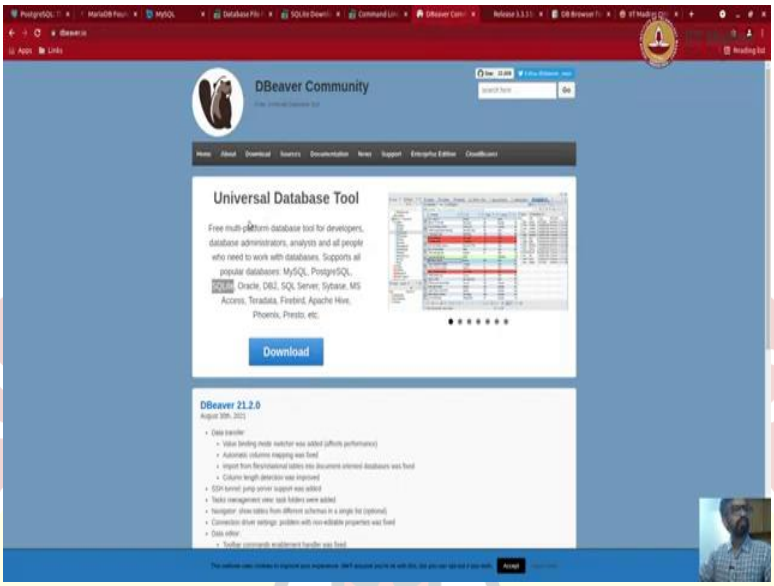


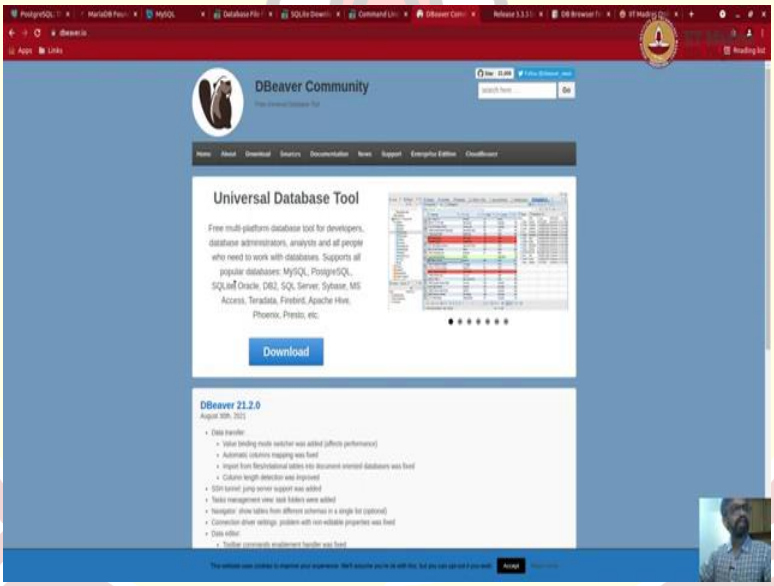


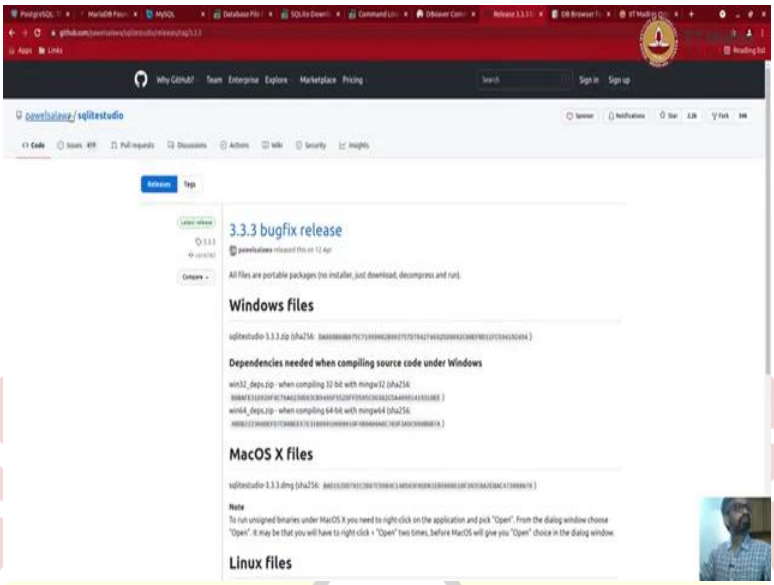


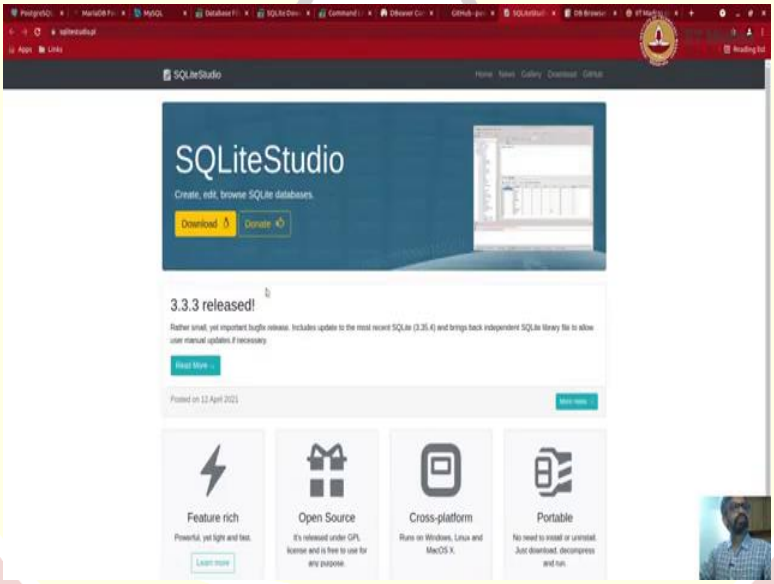
So, to open the database and transact with it, you need some kind of client. You know SQLite comes with default command line client, called SQLite 3, which you can open in your command line, for example if you go to command line, if you have installed SQLite, you can just check whether it is installed by you know calling which SQLite 3, show the path and you can see the details of it by or looking at the help. So, you can use this client to open the file and explore the database.

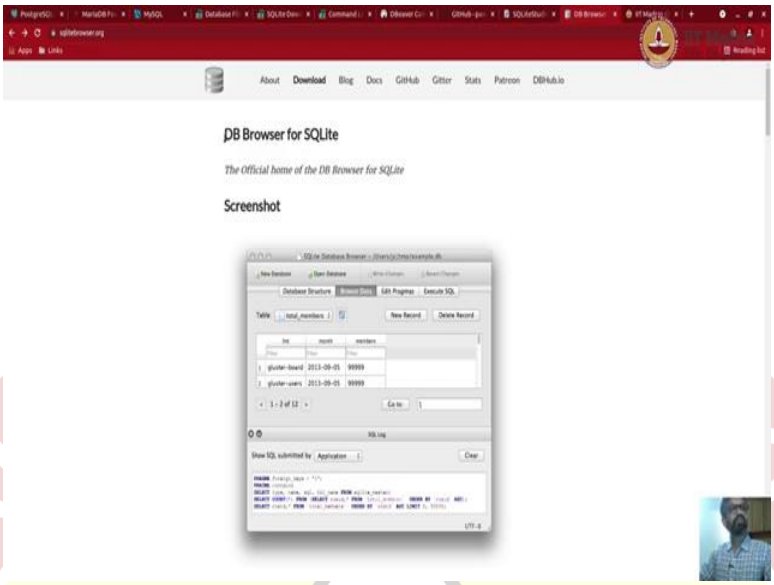
(Refer Slide Time: 3:30)

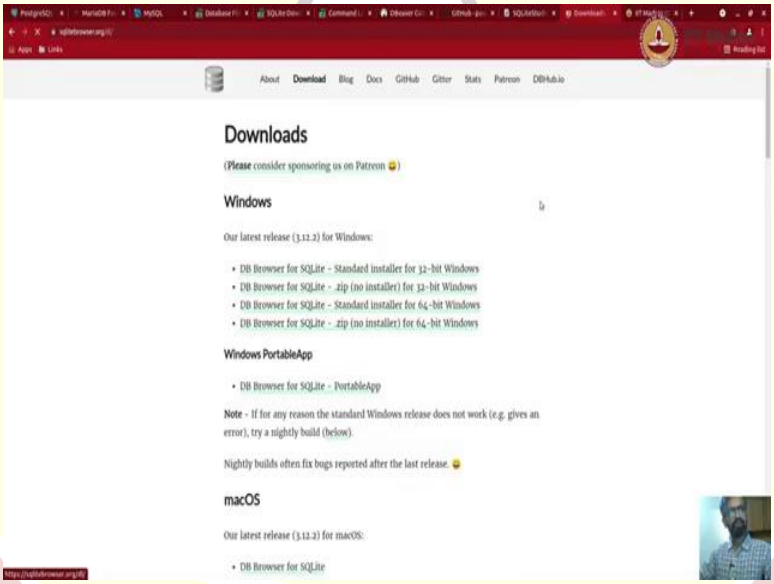


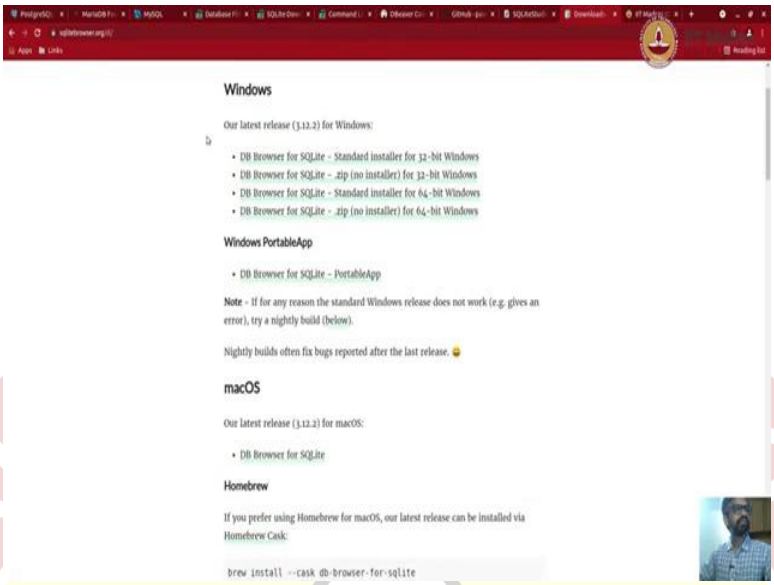



pawelsalawa/sqlitestudio
3.3.3 bugfix release
All files are portable packages (no installer, just download, decompress and run).
Windows files
Dependencies needed when compiling source code under Windows
MacOS X files
Note
Linux files


SQLiteStudio
SQLiteStudio
Create, edit, browse SQLite databases
Download
Donate
3.3.3 released!
Feature rich
Open Source
Cross-platform
Portable


About Download Blog Docs GitHub Gitter Stats Patreon DBHub.io
DB Browser for SQLite
The Official home of the DB Browser for SQLite
Screenshot


About Download Blog Docs GitHub Gitter Stats Patreon DBHub.io
Downloads
(Please consider sponsoring us on Patreon)
Windows
Our latest release (3.12.2) for Windows:
DB Browser for SQLite - Standard installer for 32-bit Windows
DB Browser for SQLite - .zip (no installer) for 32-bit Windows
DB Browser for SQLite - Standard installer for 64-bit Windows
DB Browser for SQLite - .zip (no installer) for 64-bit Windows
Windows PortableApp
DB Browser for SQLite - PortableApp
Note - If for any reason the standard Windows release does not work (e.g. gives an error), try a nightly build (below).
Nightly builds often fix bugs reported after the last release.
macOS
Our latest release (3.12.2) for macOS:
DB Browser for SQLite



Windows

Our latest release (3.12.2) for Windows:

- DB Browser for SQLite - Standard installer for 32-bit Windows
- DB Browser for SQLite - .zip (no installer) for 32-bit Windows
- DB Browser for SQLite - Standard installer for 64-bit Windows
- DB Browser for SQLite - .zip (no installer) for 64-bit Windows

Windows PortableApp

- DB Browser for SQLite - PortableApp

Note - If for any reason the standard Windows release does not work (e.g. gives an error), try a nightly build (below).

Nightly builds often fix bugs reported after the last release. 😀

macOS

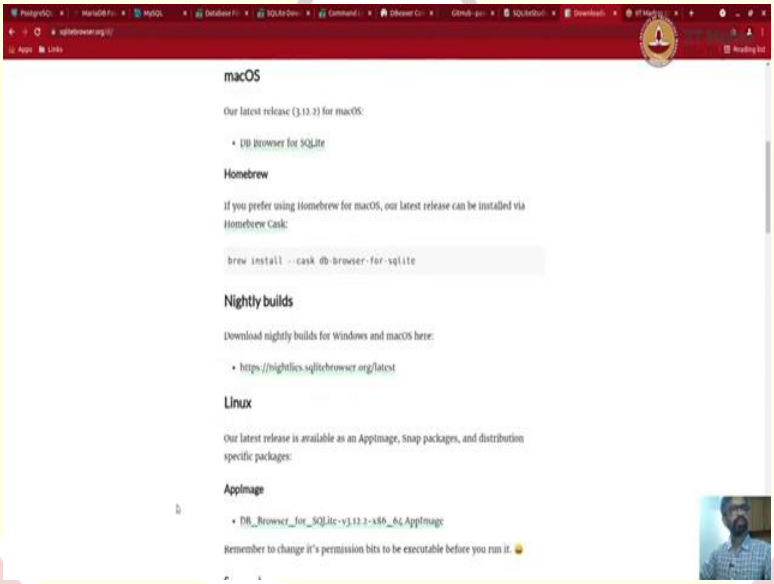
Our latest release (3.12.2) for macOS:

- DB Browser for SQLite

Homebrew

If you prefer using Homebrew for macOS, our latest release can be installed via Homebrew Cask:

```
brew install --cask db-browser-for-sqlite
```

macOS

Our latest release (3.12.2) for macOS:

- DB Browser for SQLite

Homebrew

If you prefer using Homebrew for macOS, our latest release can be installed via Homebrew Cask:

```
brew install --cask db-browser-for-sqlite
```

Nightly builds

Download nightly builds for Windows and macOS here:

- https://nightlies.sqlitebrowser.org/latest

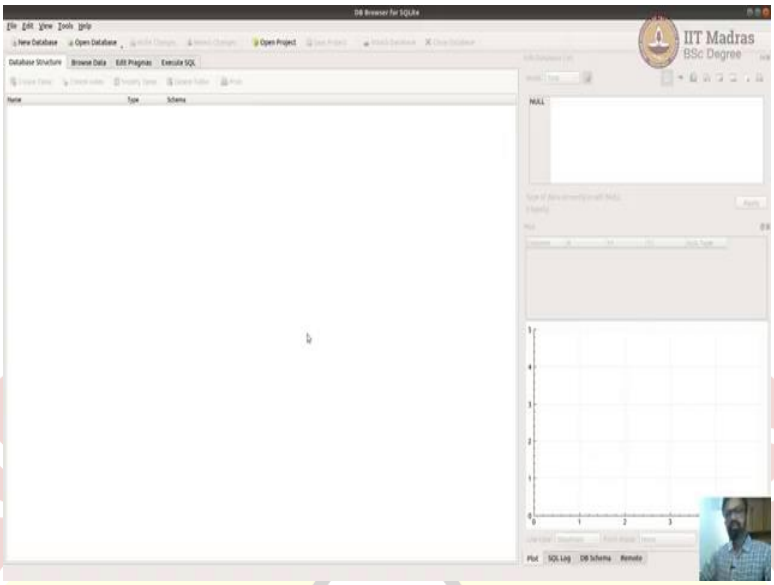
Linux

Our latest release is available as an AppImage, Snap packages, and distribution specific packages:

AppImage

- DB_Browser_for_SQLite-v3.12.2-x86_64.AppImage

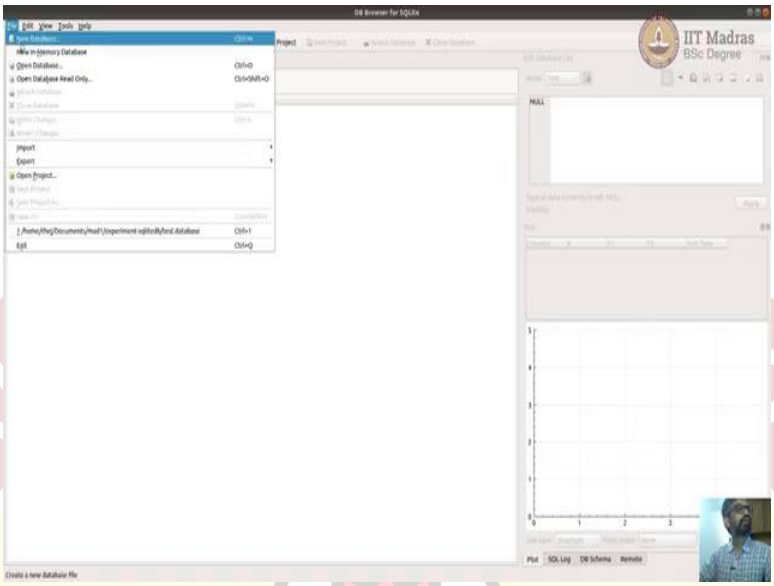
Remember to change it's permission bits to be executable before you run it. 😀

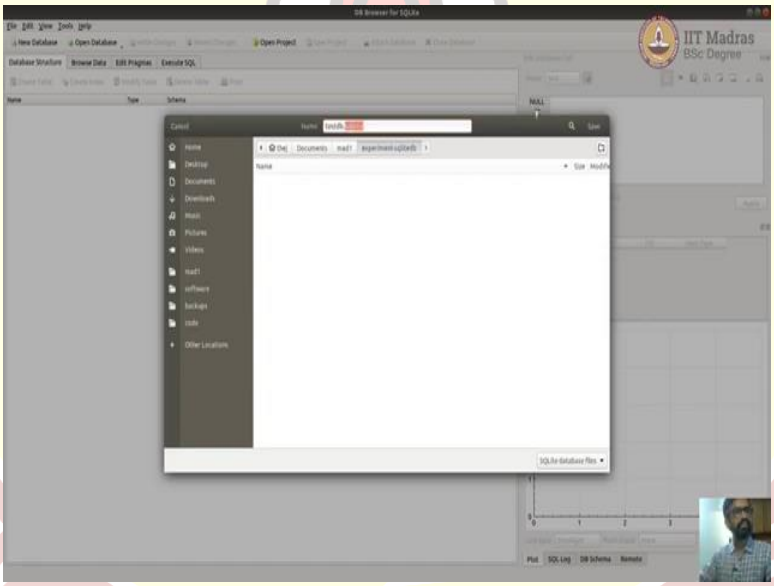


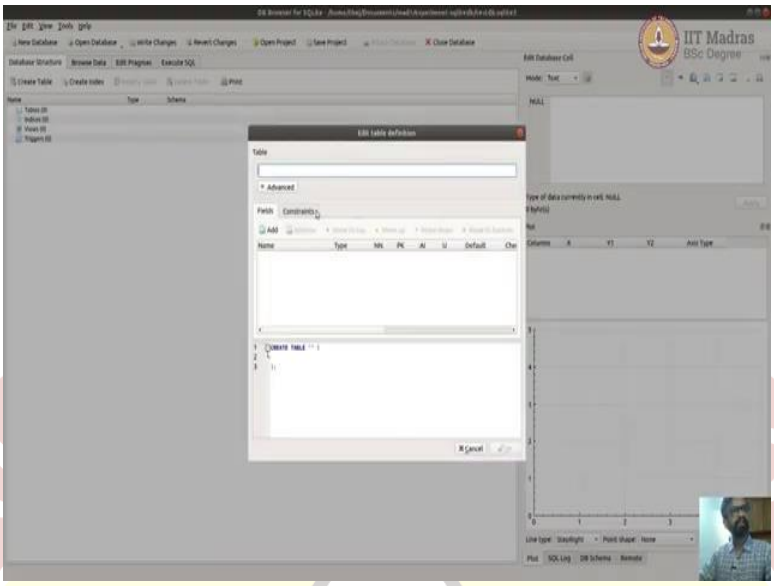
But GUIS also available, which makes it easy to explore. There are GUIS like, DBeaver, community version that you can use to explore many databases including MySQL, PostgreSQL and SQLite. There is a project called SQLite studio, which you can use on your ubuntu desktop to explore the file SQLite file, or SQLite database. And then there is DB browser for SQLite, which is also nice GUI, open source a tool that you can use to explore the database.

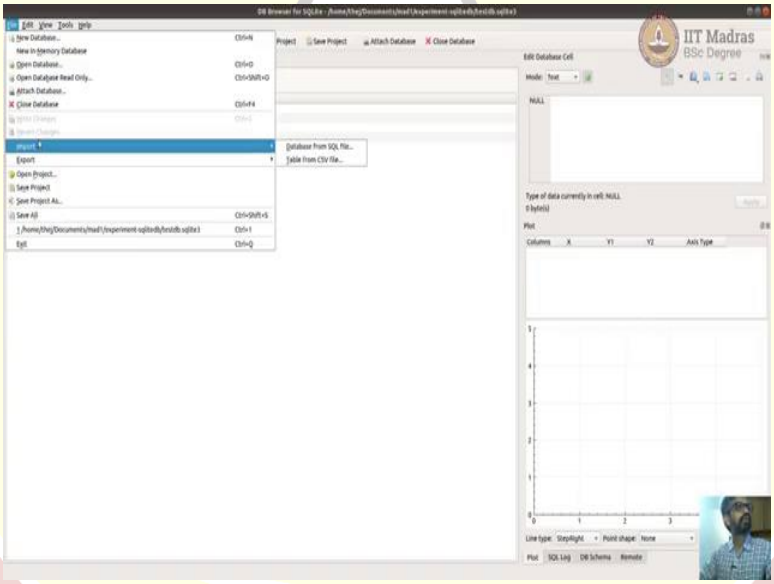
In this experiment we are going to use DB browser SQLite. Hence pause for a second and install this tool, you can go to download and download the version that is required for your operating system, whether it is windows, macOS, or you know Linux, and install it. Once you install it, you can open it, when you open it, it looks like this. It is a full featured client for exploring, or creating the SQLite database.

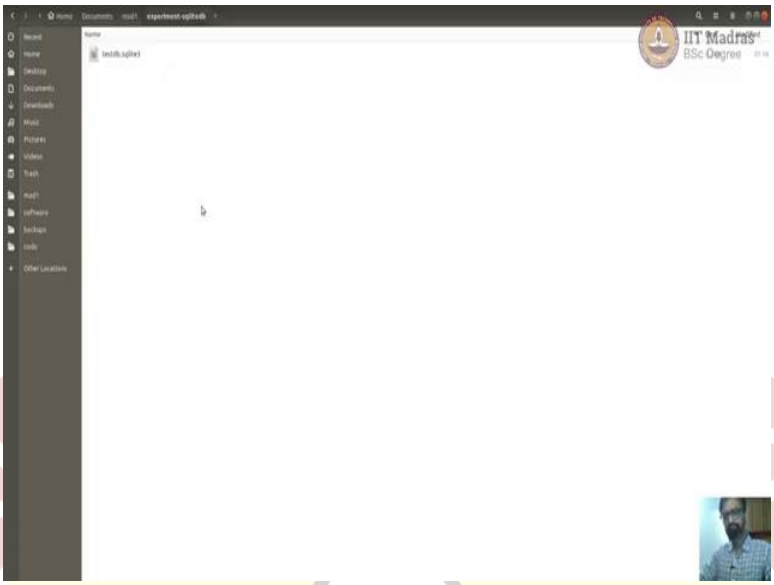
(Refer Slide Time: 4:40)






IIT Madras
BSc Degree

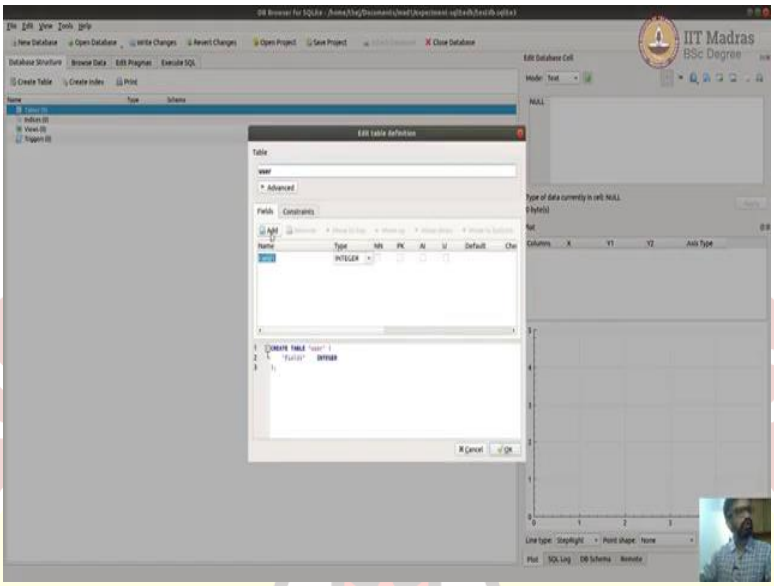

IIT Madras
BSc Degree

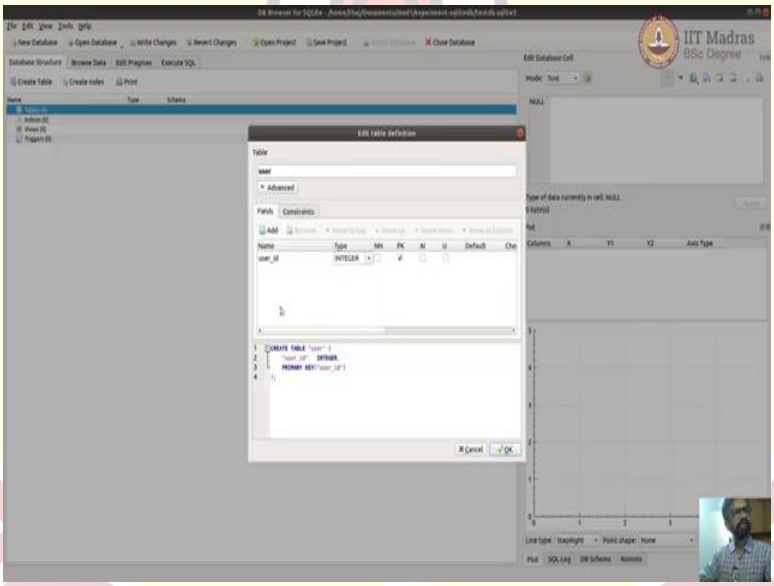


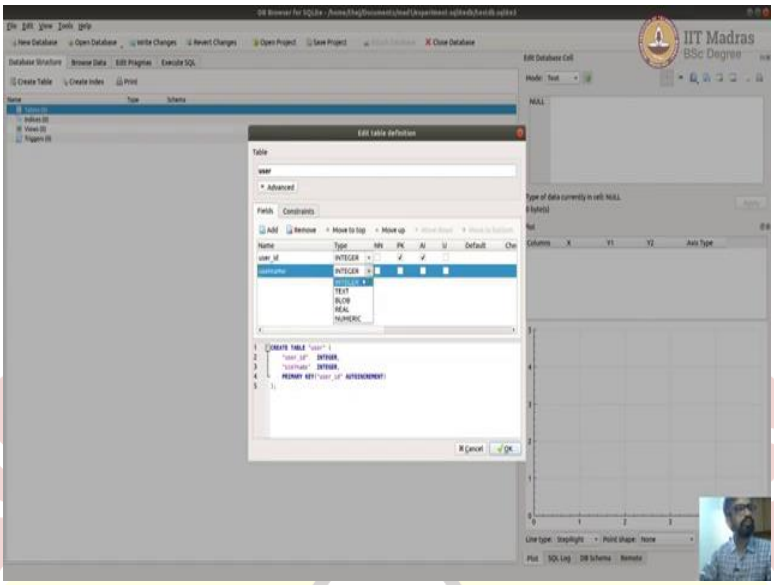
So, let us create a SQLite database, you can do that just by clicking on new database, or file new database and you know you can go to the folder, where you want to store the database, because it is just a file, you need to go to the path, where you want to store it. So, I am going to go, you know my experiments folder that I created and create a file called testdb.sqlite3, just to signify that it is version 3, I am using the extension SQLite 3.

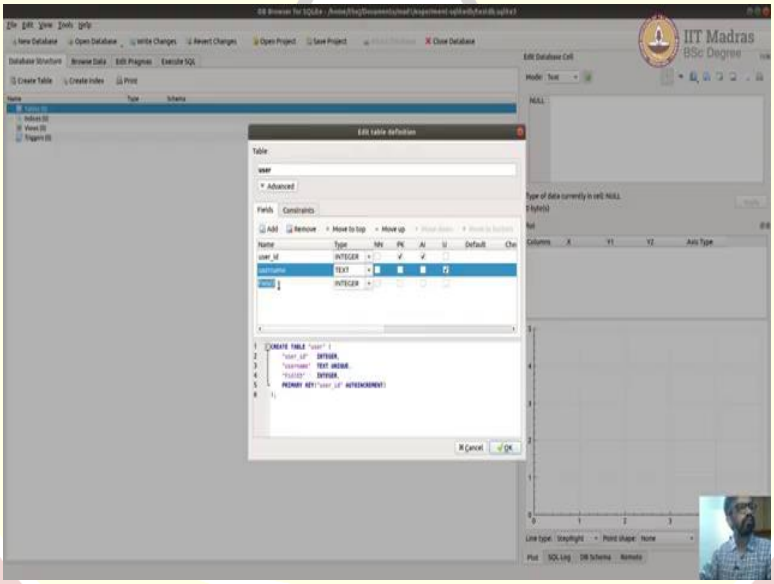
And just click save and it opens up a pop-up to ask you for creating the table. Let us not do that now, and just save it, save is done by clicking on control S, or write changes, or this button. Now, this is, we have created a database, you can now, you can go here and check, as you can see, SQLite 3, 5 on the local file system.

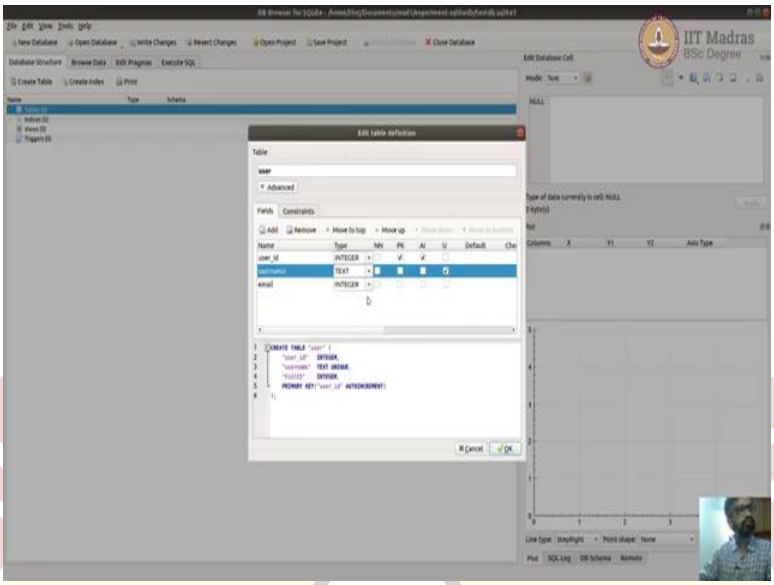
(Refer Slide Time: 5:54)

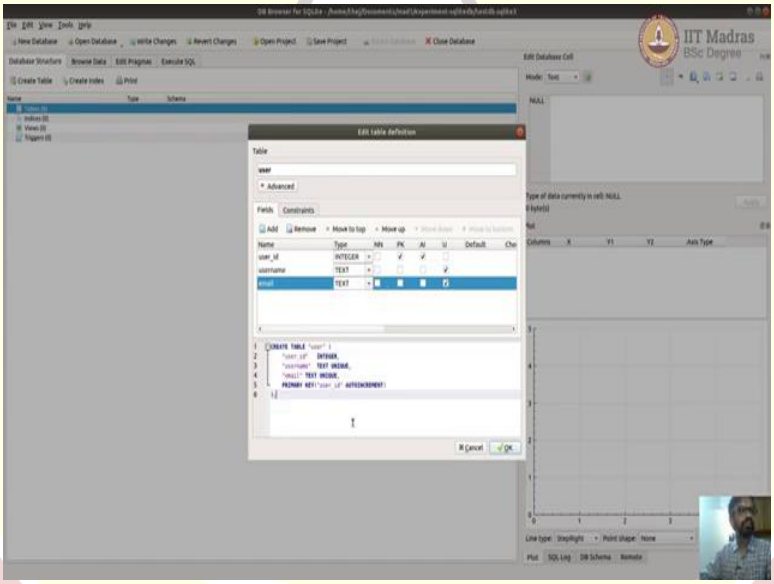


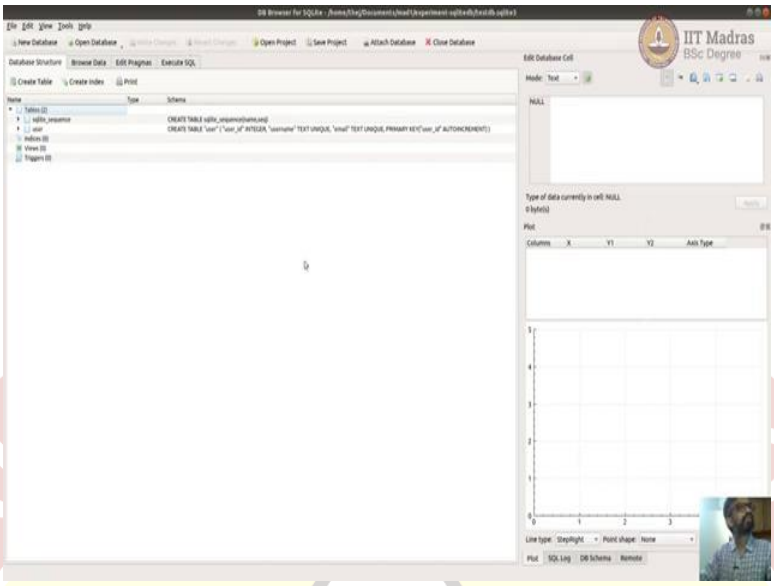



IIT Madras
BSc Degree
Edit table definition
Table
user
Advanced
Fields
Constraints
Add
Remove
Move to top
Move up
Name
Type
NN
PK
AI
U
Default
user_id
INTEGER
username
INTEGER
BLOB
TEXT
REAL
NUMERIC
CREATE TABLE "user" (
"user_id" INTEGER,
"username" INTEGER,
PRIMARY KEY("user_id" AUTOINCREMENT)
);
Cancel
OK


IIT Madras
BSc Degree
Edit table definition
Table
user
Advanced
Fields
Constraints
Add
Remove
Move to top
Move up
Name
Type
NN
PK
AI
U
Default
user_id
INTEGER
username
TEXT
INTEGER
CREATE TABLE "user" (
"user_id" INTEGER,
"username" TEXT UNIQUE,
"field0" INTEGER,
PRIMARY KEY("user_id" AUTOINCREMENT)
);
Cancel
OK


Edit table definition
Table
user
Advanced
Fields
Constraints
Add
Remove
Move to top
Move up
Name
Type
NN
PK
AI
U
Default
user_id
INTEGER
username
TEXT
email
INTEGER
CREATE TABLE "user" (
"user_id" INTEGER,
"username" TEXT UNIQUE,
"email" INTEGER,
PRIMARY KEY("user_id" AUTOINCREMENT)
);
Cancel
OK

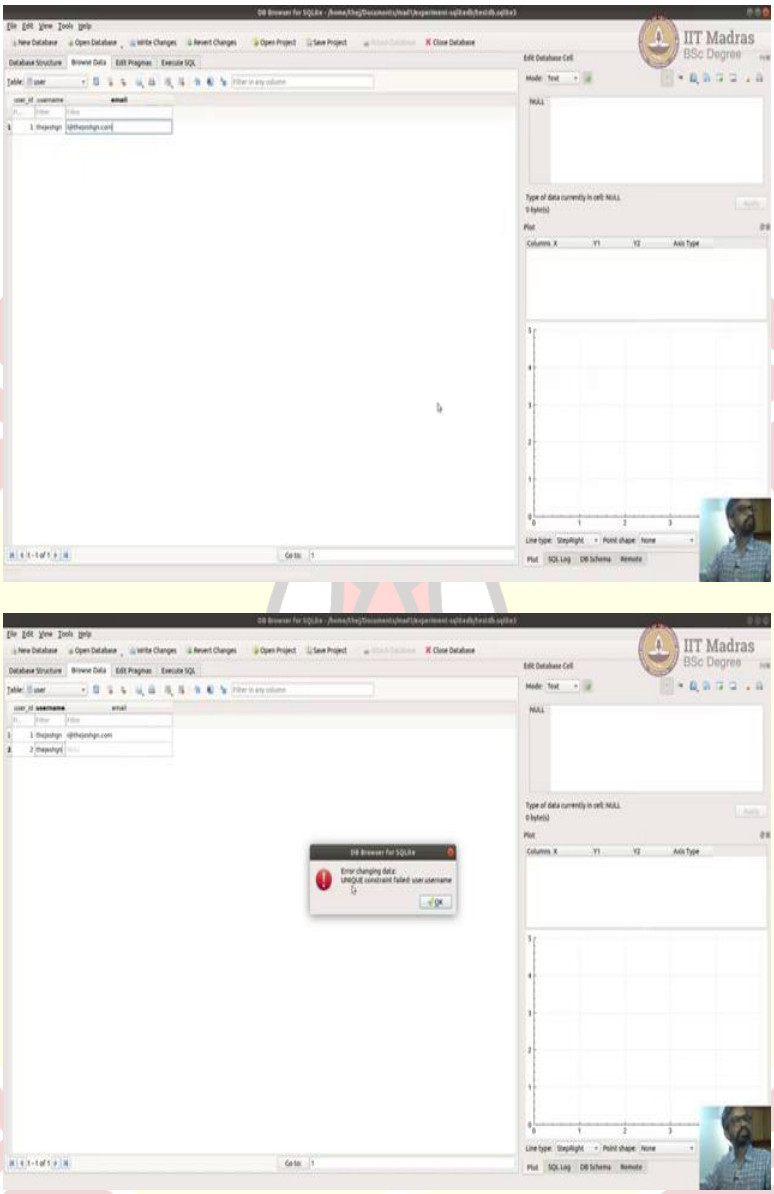

Edit table definition
Table
user
Advanced
Fields
Constraints
Add
Remove
Move to top
Move up
Name
Type
NN
PK
AI
U
Default
user_id
INTEGER
username
TEXT
email
TEXT
CREATE TABLE "user" (
"user_id" INTEGER,
"username" TEXT UNIQUE,
"email" TEXT UNIQUE,
PRIMARY KEY("user_id" AUTOINCREMENT)
);
Cancel
OK

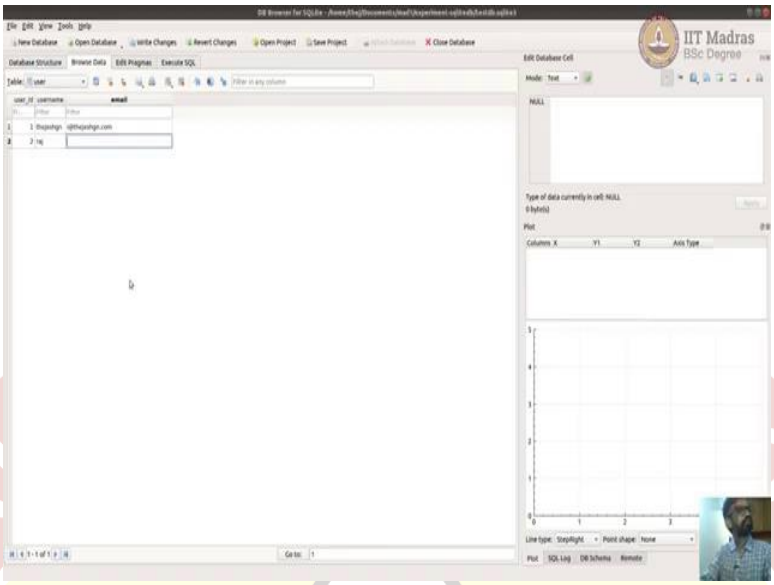


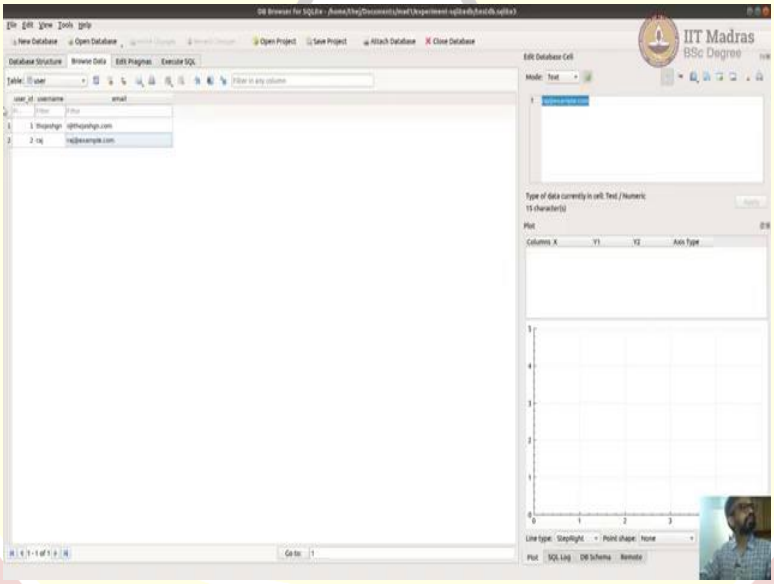
Now, let us go back to our UI, and let us create a table called user, that you can do by clicking on create table. And I am going to give the table name as user, and add a column, it is called fields here, but it is column I am going to create a column called user_Id, which will be an integer, which will be a primary key. And it will be an auto increment and I am going to create another table called user name.

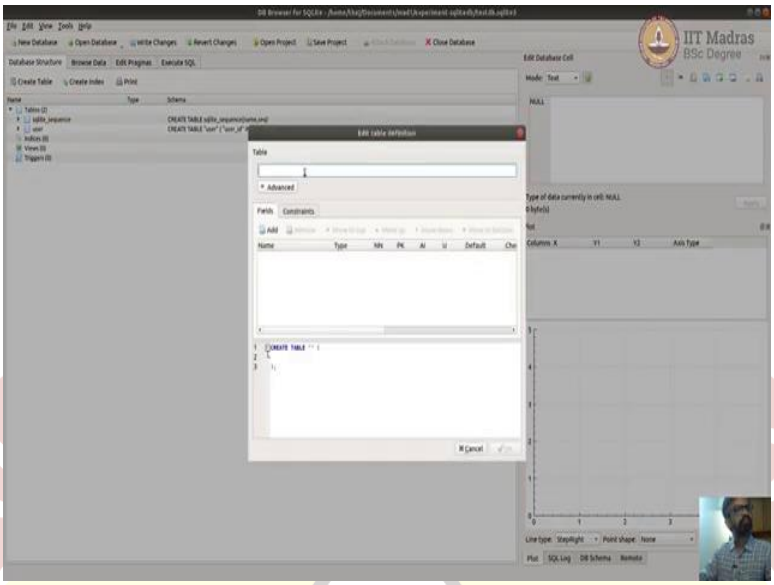
Another field actually, or column called username, which is type text it will be unique. And another column called email, so type text and which will also be unique; you can see the whole script to create the table at the bottom. Now, if you just want to cross verify it, solve the constraints and stuff, click on and then write changes. Here write changes is committing the changes to the file. So, now this file is saved and hence this details of this table.

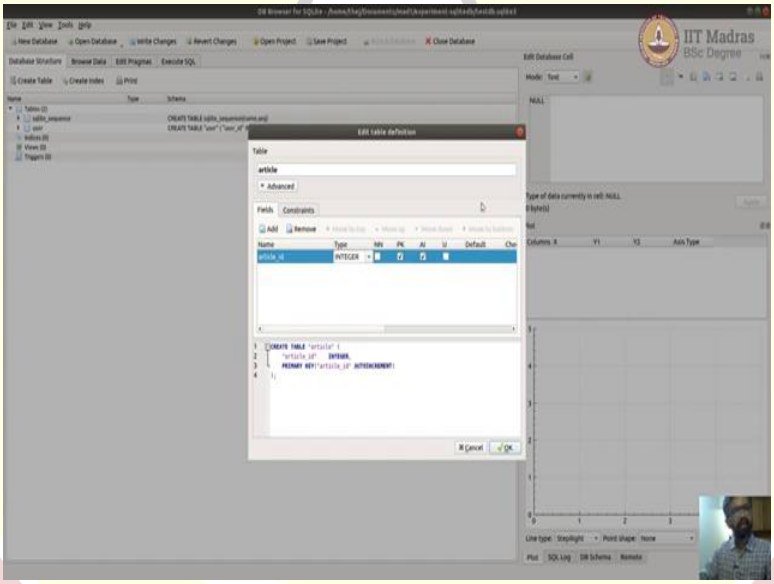
(Refer Slide Time: 7:06)

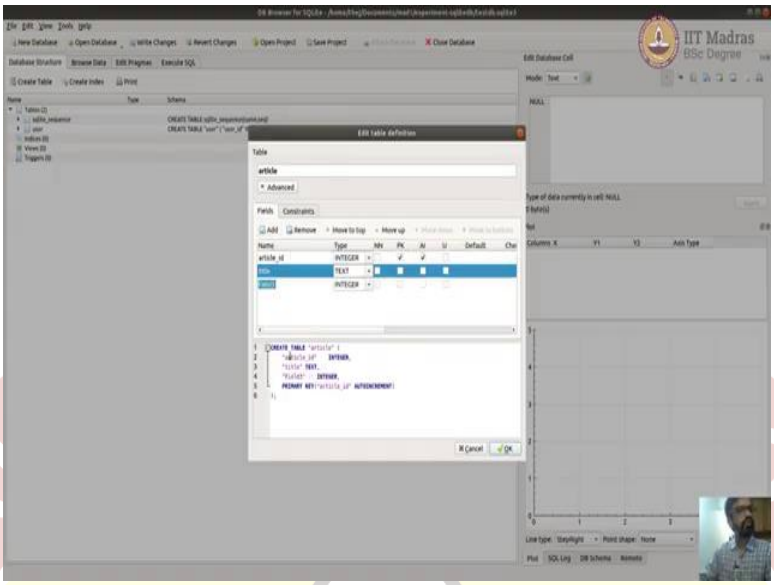


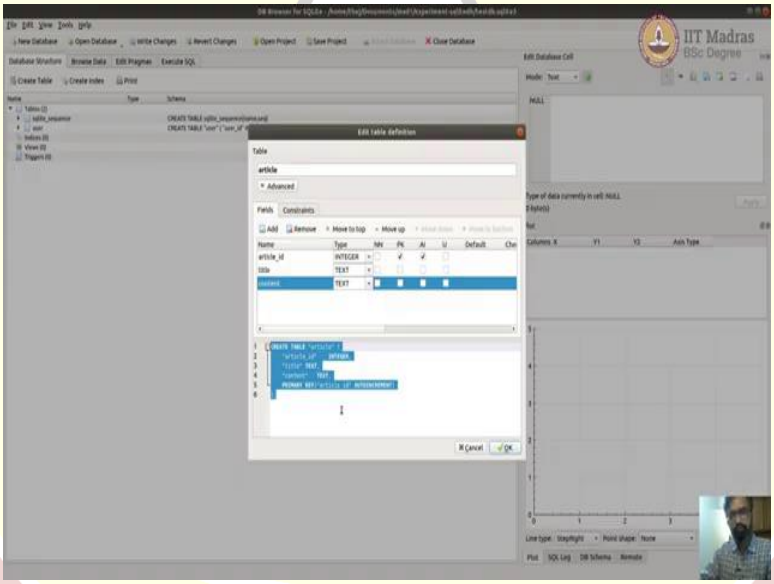

IIT Madras
BSc Degree


IIT Madras
BSc Degree


IIT Madras
BSc Degree


IIT Madras
BSc Degree



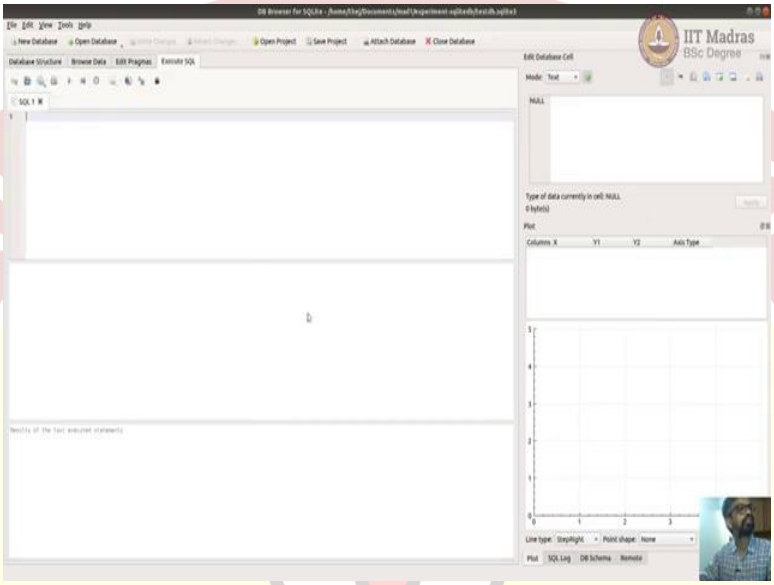


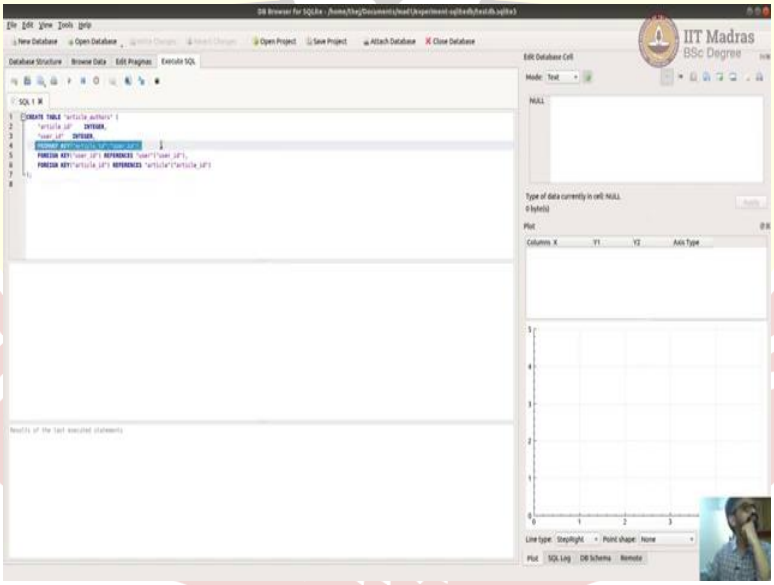
To explore the data in the table, you can go click on user here in the browse data, you can see there is no data, you can insert by clicking this plus icon and let me just insert ‘Thejesh’ email id i@thejeshgn.com. And I am going to write changes, or control S, and then we click another add another row, let us give same name, let us see what happens, thejeshgn and if I change it try to change the column, it shows me this error, unique constraint on user, user name.

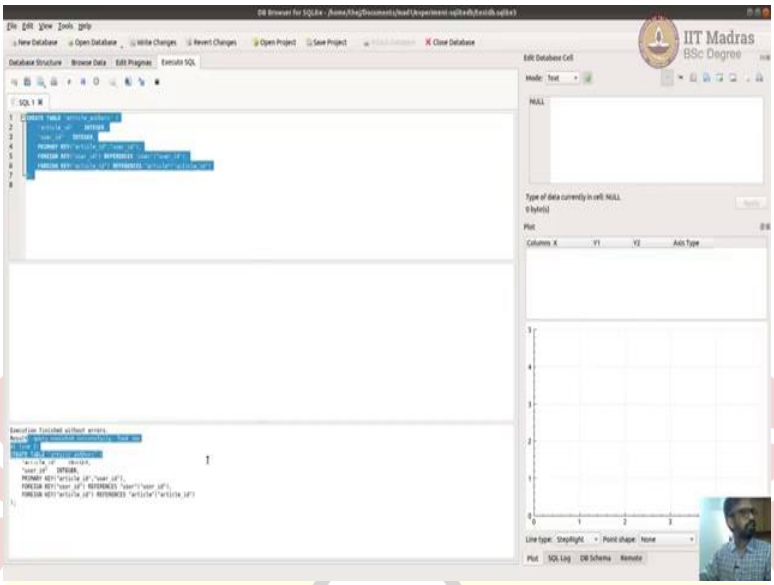
Because it is repeated, it supposed to be unique. And hence it is throwing an arrow, it would not allow me to go to the next column. So, let us make it raj and raj@example.com, just going to write the changes. I want to create another table by clicking on create table, I am going to call it as article and add the column, article_id, which is an integer, primary key and auto increment.

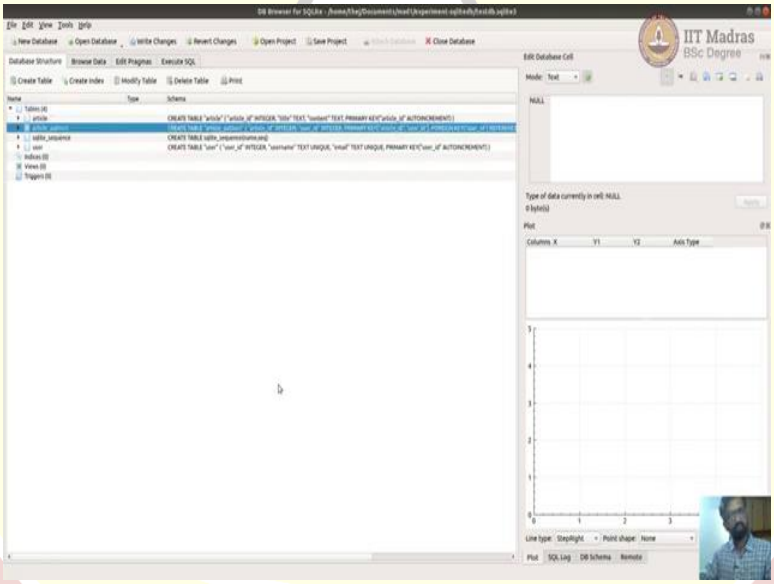
And then I am going to add another column called title, it will be article title, I am going to make it like a text and then add another column called content, which will be article content, again text. This is the script that is going to run and say, and we have the, we have the table.

(Refer Slide Time: 9:03)



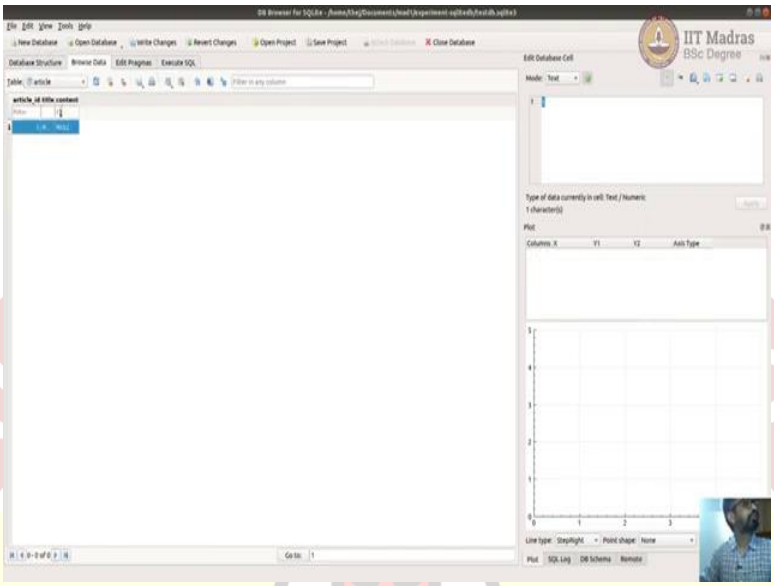


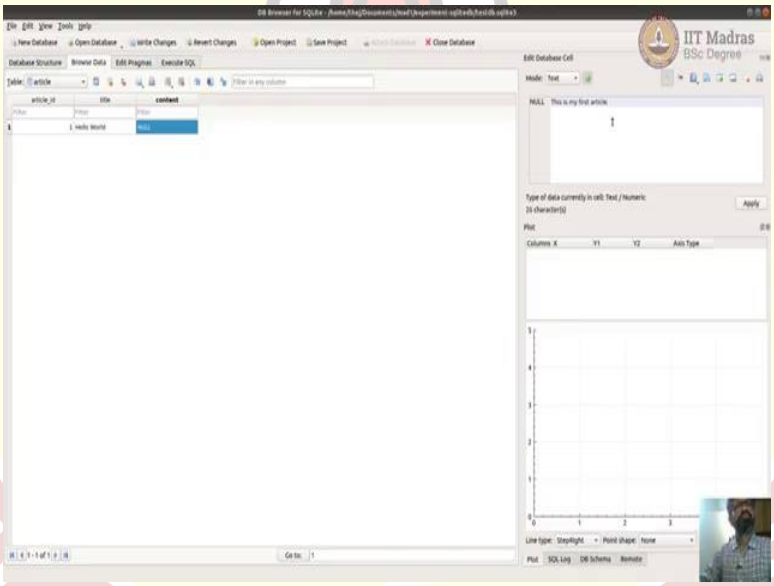


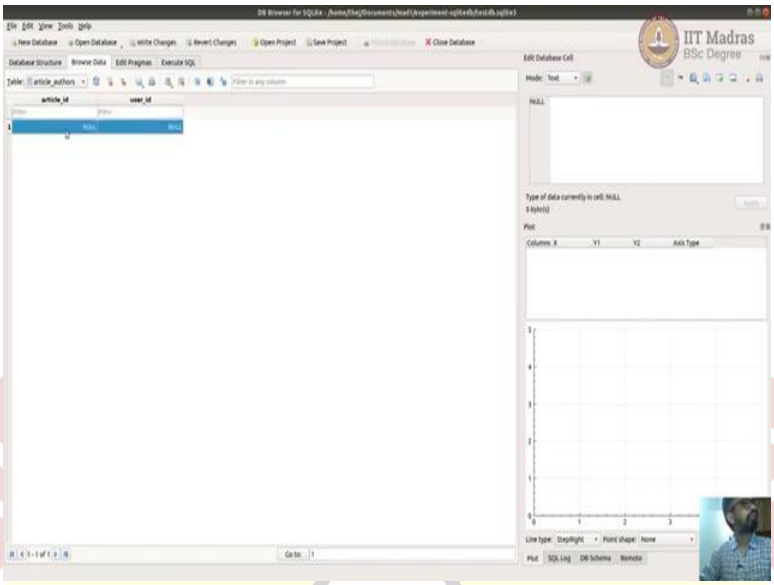


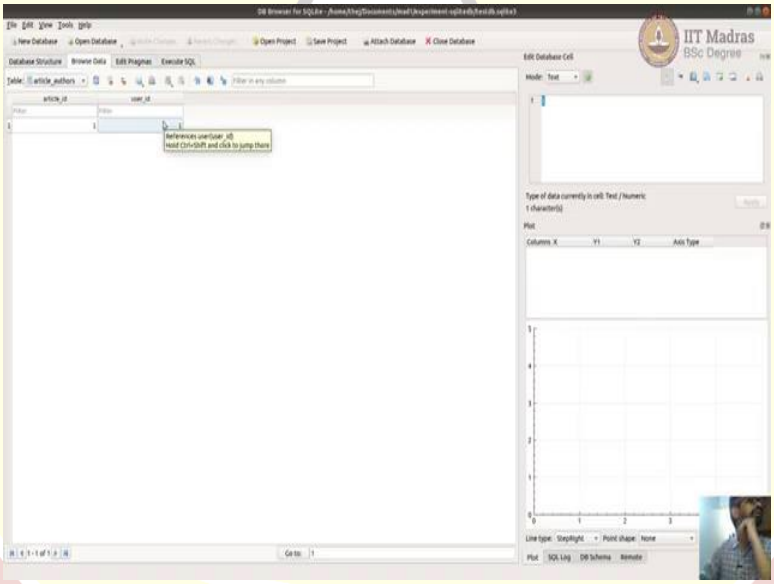
Now, we have to create a relationship table, which tells that article can have one, or many writers, which are users, or one or many authors, you can call it. So, I am just going to create the table using the script this time, I am going to create a table called article_authors, which is article_id and user_id, which I just foreign keys to the article table and user table and they together form the primary key. I am just going to select it, and run it and it has created the table and you can see here article_authors. So, this way you can create either by you know using GUI, or executing the query.

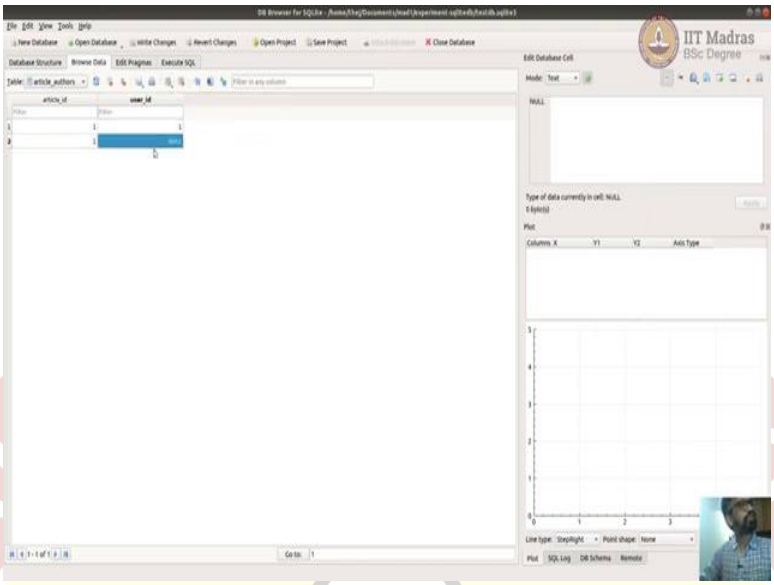
(Refer Slide Time: 9:56)






DB Browser for SQLite
IIT Madras
BSc Degree

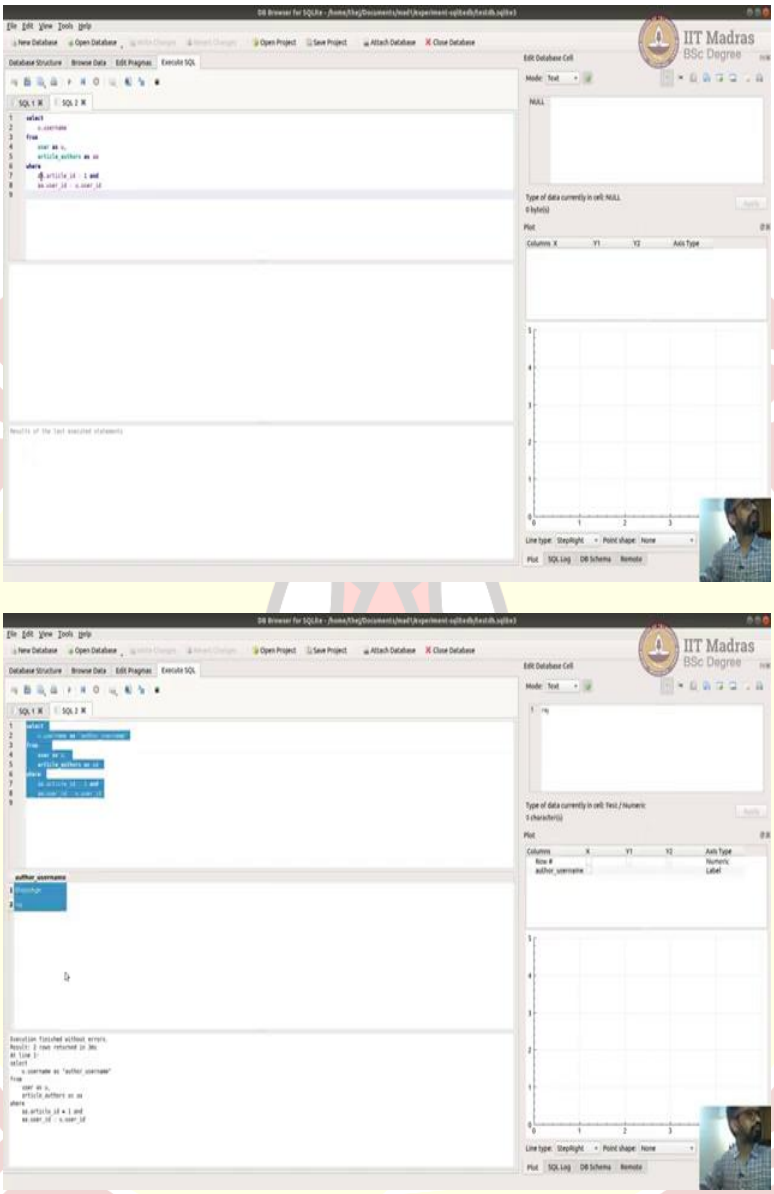

DB Browser for SQLite
IIT Madras
BSc Degree



Now, I am just going to insert one row into article table, it is one title is let us say hello, well it is a big test you can go to this right side area and fill up the things. And then content will be this is my first article, and save it. Now, I want to establish a relationship between article and it is you know use writers using this articles authors table.

Let us go to the table article authors, now we know the article_id with one. So, let us add a row the article_id one and user id one all right. So, that should give us you know the article has one author, or one writer, whose id is one. If you want to have more than one, let us say you know same article has raj also, also writer, you can enter these two same.

(Refer Slide Time: 11:12)



Now, we can also run other queries to get the you know, let us say if you want to get all the writers of the articles, for article number one. Then you can run the query, select user name from user and article authors, we are article id is one and we are just joining it using user id, same as you use the keyword as it is just a alias, when I run it you can see author username is thegeshgn, raj. So, that is it actually, now you can you know while before you end the thing you can close the database, and you know file is closed.

So, now you saw that, you know how simple working with SQLite is, it is because of its simplicity and zero configuration it could be the easiest database that we can use to experiment with, since we are learning a lot we are going with a simple database. But that said whatever the concepts that you learn with SQLite, you can easily translate to other RDBMS like MySQL, or Postgres, you will often see that when you set up the local development environment all most developers use SQLite as the local development, database.

And then when they move to production, they will probably use you know some other database like MySQL, as Postgres, that is it in today's screencast. Thank you so much for watching.