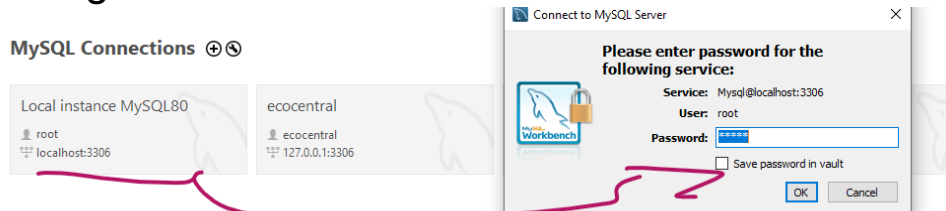


Project Guide for EDP ඩඩඩඩඩඩඩ

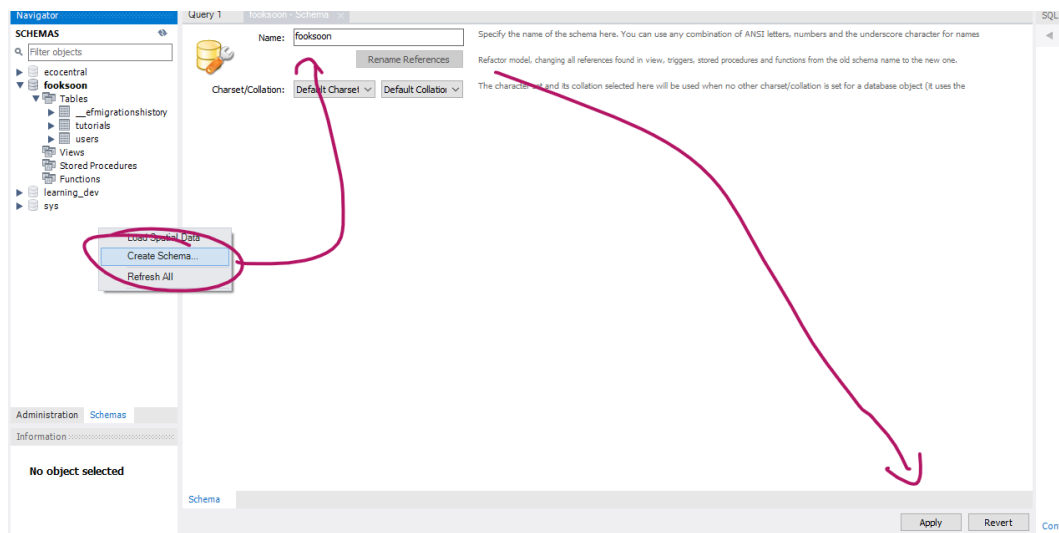
First time setup:

Setup new SQL Workbench

1. Login as local instance

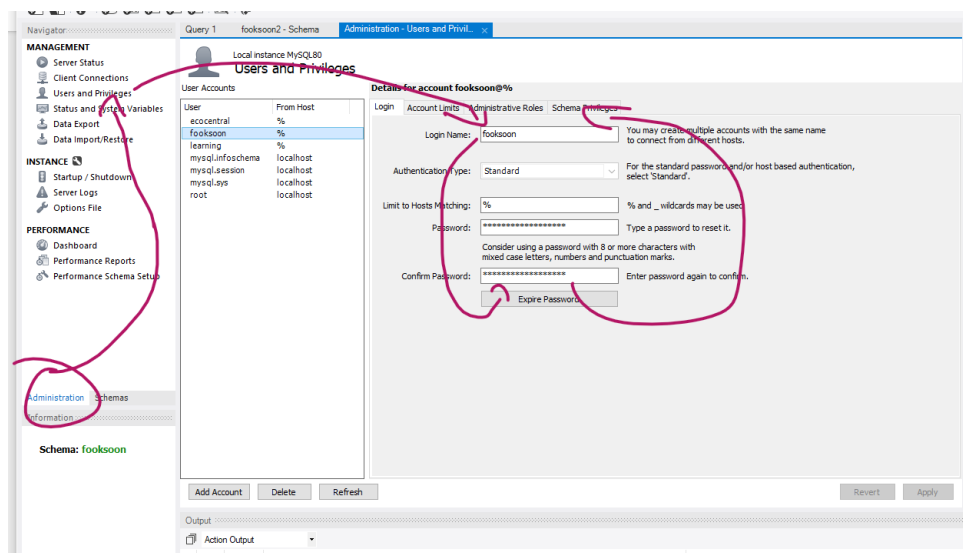


2. Create new schema and name it as fooksoon



3. Make admin account for fooksoon schema with all privileges.

Username: fooksoon | Password: mysql



Local instance MySQL80 Users and Privileges

Details for account fooksoon@%

Schema: fooksoon

Privileges: ALTER, ALTER ROUTINE, CREATE, CREATE ROUTINE, CREATE TEMPORARY TABLES, CREATE VIEW, DELETE, DROP,

Schema and Host fields may use % and _ wildcards. The server will match specific entries before wildcarded ones.

Revoke All Privileges Delete Entry Add Entry...

New Schema Privilege Definition

Select the Schema for which the user 'fooksoon' will have the privileges you want to define.

Schema

☐ All Schema (%) This rule will apply to any schema name.

☐ Schemas matching pattern: This rule will apply to schemas that match the given name or pattern. You may use _ and % as wildcards in a pattern. Escape these characters with \ in case you want their literal value.

☒ Selected schema: fooksoon Select a specific schema name for the rule to apply to.

fooksoon ALTER, ALTER ROUTINE, CREATE, CREATE ROUTINE, CREATE TEMPORARY TABLES, CREATE VIEW, DELETE, DROP,

Schema and Host fields may use % and _ wildcards. The server will match specific entries before wildcarded ones.

Revoke All Privileges Delete Entry Add Entry...

The user 'fooksoon'@'%' will have the following access rights to the schema 'fooksoon':

Object Rights

☒ SELECT

☒ INSERT

☒ UPDATE

☒ DELETE

☒ EXECUTE

☒ SHOW VIEW

DDL Rights

☒ CREATE

☒ ALTER

☒ REFERENCES

☒ INDEX

☒ CREATE VIEW

☒ CREATE ROUTINE

☒ ALTER ROUTINE

☒ EVENT

☒ DROP

☒ TRIGGER

Other Rights

☐ GRANT OPTION

☒ CREATE TEMPORARY TABLES

☒ LOCK TABLES

Unselect All Select "ALL"

Revert Apply

5. Make new connection

Welcome to MySQL Workbench

MySQL Connections + -

Local instance MySQL80

root

localhost:3306

Setup New Connection

Connection Name: fooksoon Type a name for the connection

Connection Method: Standard (TCP/IP) Method to use to connect to the RDBMS

Parameters SSL Advanced

Hostname: 127.0.0.1 Port: 3306 Name or IP address of the server host - and TCP/IP port.

Username: root Name of the user to connect with.

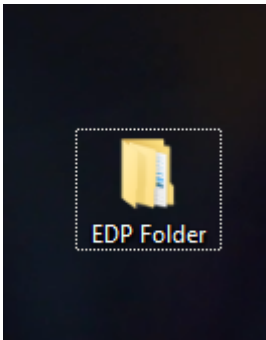
Password: [Show in Vault ...] [Clear] The user's password. Will be requested later if it's not set.

Default Schema: fooksoon The schema to use as default schema. Leave blank to select it later.

Configure Server Management... Test Connection Cancel OK

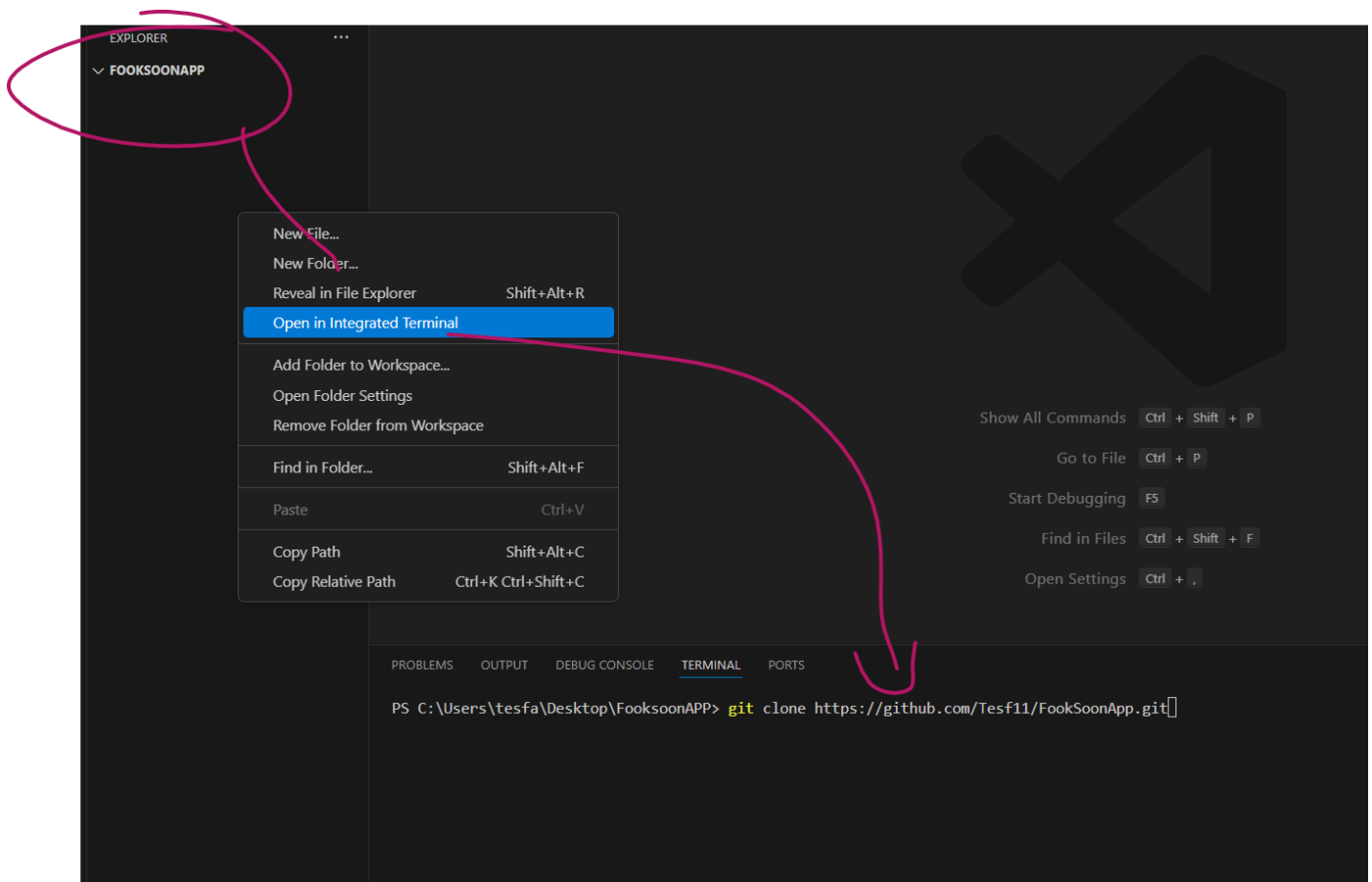
Setting up github repo

1. Make a dedicated folder (Name it like EDP folder or something) in somewhere safe and sound



2. Open it in VS Code and start integrated terminal and type

`git clone https://github.com/Tesf11/FookSoonApp.git`



Project Structure:

Frontend is using React, similar to Full stack development

1. Do “npm i” for first time setup



2. Do “npm start” to start local server”

```
Run `npm audit` for details.
PS C:\Users\tesfa\Desktop\EDP Folder\FookSoonApp\Frontend> npm start

> client@0.0.0 start
> vite

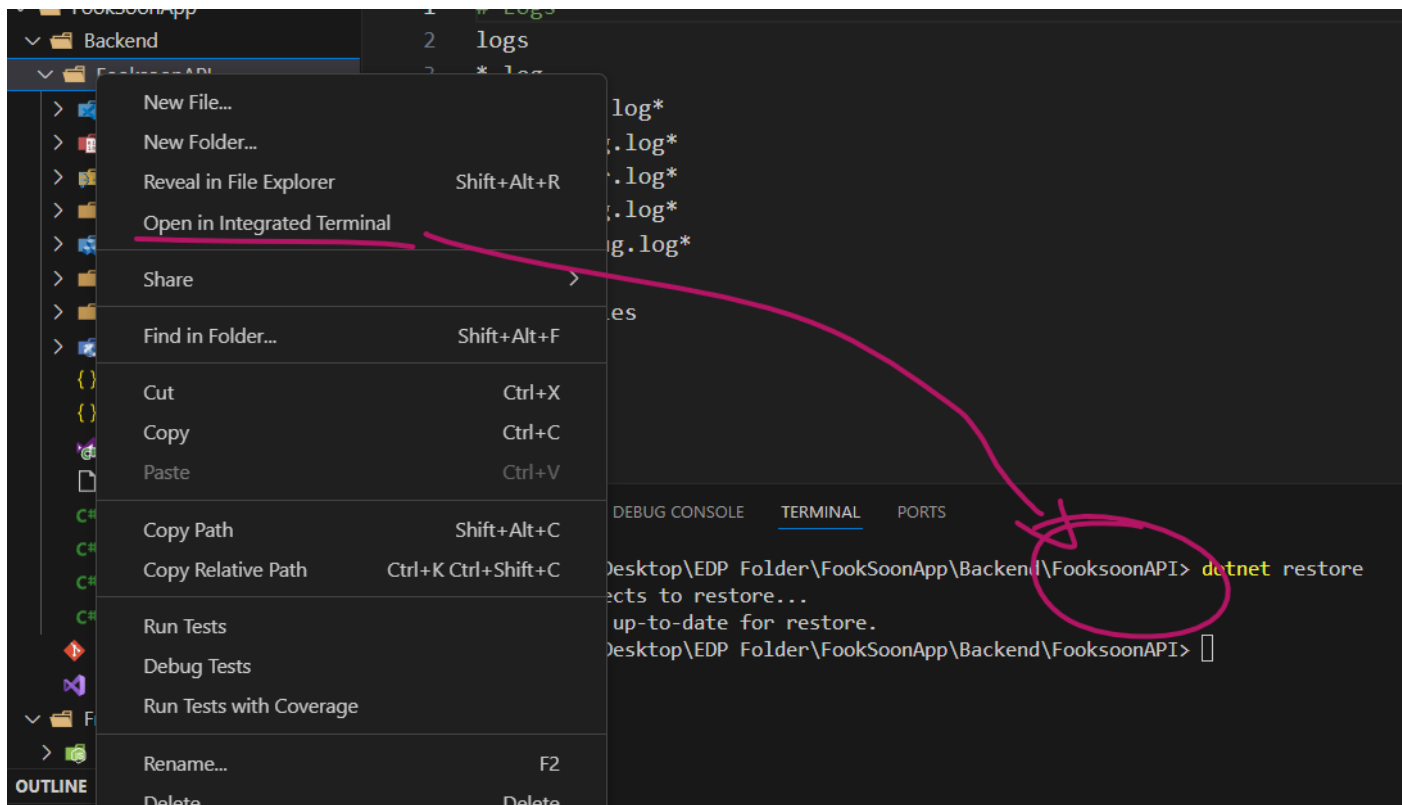
VITE v5.4.3 ready in 209 ms

→ Local:   http://localhost:3000/
→ Network: use --host to expose
→ press h + enter to show help
```

You're suppose to use Visual studio 2022 for the backend but for convenience sake, we will be using VS Code. Bonus: Jialong can code on his mac 🤝🤝🤝🤝🤝🤝🤝🤝 as well.

To start backend:

1. Right click and start integrated terminal (**ALERT: MAKE SURE THE TARGET DIRECTORY IS FOOKSOONAPI, NOT BACKEND**)



2. (First time setup)

Type:

`dotnet restore`

`dotnet dev-certs https --trust`

`dotnet ef database update` (Must update database everytime you make a model change like a new User table)

```
PS C:\Users\tesfa\Desktop\EDP Folder\FookSoonApp\Backend\FooksoonAPI> dotnet restore
Determining projects to restore...
All projects are up-to-date for restore.

PS C:\Users\tesfa\Desktop\EDP Folder\FookSoonApp\Backend\FooksoonAPI> dotnet dev-certs https --trust
Trusting the HTTPS development certificate was requested. A confirmation prompt will be displayed if the
certificate was not previously trusted. Click yes on the prompt to trust the certificate.
Successfully trusted the existing HTTPS certificate.

PS C:\Users\tesfa\Desktop\EDP Folder\FookSoonApp\Backend\FooksoonAPI> dotnet ef database update
Build started...
Build succeeded.
The Entity Framework tools version '7.0.0' is older than that of the runtime '8.0.11'. Update the tools
latest features and bug fixes. See https://aka.ms/AAC1fw for more information.
No migrations were applied. The database is already up to date.
Done.
PS C:\Users\tesfa\Desktop\EDP Folder\FookSoonApp\Backend\FooksoonAPI>
```

3. To run the program, type in the following (this command will restart and run the backend upon any changes)

`dotnet watch run --urls "https://localhost:7004;http://localhost:5082"`

The usual stuff:

Github Link:

`git clone https://github.com/Tesf11/FookSoonApp.git`

To make changes:

Note | Do it under this directory:

```
PS C:\Users\tesfa\Desktop\EDP Folder\FookSoonApp\Backend\FooksoonAPI> cd ..  
PS C:\Users\tesfa\Desktop\EDP Folder\FookSoonApp\Backend> cd ..  
PS C:\Users\tesfa\Desktop\EDP Folder\FookSoonApp> █
```

Go to Github

`git add .`

`git commit -m "Type your Changes here"` e.g. Include change messages

`git push origin main`

To Update and Pull changes:

`git pull origin main`

Run backend

`dotnet watch run --urls https://localhost:7004;http://localhost:5082`

Run Frontend

`npm start`

Update database:

`dotnet ef database update` (do it under backend/FooksoonAPI directory)