Follow these steps to set up your own copy of Key Trader

Step I. Setting Up the Discord Bot

1. First you need a key for your bot. Create a new application at
   https://discord.com/developers/applications
2. Navigate to the General Information tab, and find your ClientID.
3. Create an invite link for your bot:
   https://discordapp.com/oauth2/authorize?&client_id=CLIENTID&scope=bot&permissions=8
   where CLIENTID is your ClientID
4. You can use this link to invite your bot to your Discord guild.
5. Navigate to the Bot tab, and copy your token to the clipboard.
6. In your se498-s2020-Key-Trader project, navigate to the discordBot folder. In this folder, create a new file called bot-token.json.
7. Open your newly created file and paste your bot's token. Put quote marks before and after your key.
8. Save bot-token.json.
9. Navigate to the src folder.
10. Install NPM and Node.js.
11. Open your terminal.
12. Run the command "npm install"
13. Run the command "node ktbot.js"
14. In the future, you will be able to start up the bot by repeating step 13.

Step II. Setting Up The Database and Web Server

1. Start your VM.
2. Login to your VM using the capstone login and password.
3. Type in `ip a s` and hit enter. You will find your ip shown in the 4th/5th line of code from the bottom. . Type in the folliwng commands, there seems to be an issue with the new VM.

```
> cd /etc/nginx/sites-available
> sudo rm keytrader
> sudo wget
https://raw.githubusercontent.com/harna100/udemy-tutorial/master/keytrader
> sudo service nginx restart
```

4. Create a folder on your machine (not the VM) where you will store KeyTrader files.
5. Open a new terminal/command prompt window on your machine (not the VM) and navigate to that folder (`cd folderName`)
6. Enter ``` git clone https://github.com/ChapmanCPSC/se498-s2020-Key-Trader.git '''. You should now have KeyTrader files in the folder you created.
7. Open the project (se498-s2020-Key-Trader) in Visual Studio Code.
8. Navigate to the sftp.json file located in the .vscode section of the project.
9. Update the host IP address (line 2) with the IP address you found in step 3.
10. Save the project, then press `Shift + Command + P` to bring up a menu of actions you can take.
11. You should see an option, "SFTP: Upload Project". Select that option and hit enter (2 times).

. Open 2 new terminal/command prompt windows on your machine (not the VM). . Enter `ssh capstone@ip` where the ip address is the address you found in step 3.

. Open DataGrip, create a new project (File -> New -> Project). . In that new project, select new data source or press Command + N. . Select MariaDB, you do not need to change the name, but in the General section the user should be 'root' and the password should be '1234'. Make sure that the Save option is set to 'Forever'. . Navigate to the SSH/SSL section of the properties page. Select 'Use SSH Tunnel'. Fill out the following sections

- Proxy Host: IP from step 3
- Proxy Name: capstone
- Auth Type: Set to password
- Proxy Password: 1234
- Remember: Forever . Click 'Test Connection' to ensure that everything has been set up correctly. If connection has failed, please review the steps and try again. . Click Apply and then Ok. . Right-click on the data source you just set up and select 'Open Console'. Copy and paste the following code into the console. Highlight all of it (or Command+A) and click the play button (or Command+Enter).

```
USE KeyTrader;

--
```

```sql
-- Table structure for table `Discord_Servers`
--

DROP TABLE IF EXISTS `Discord_Servers`;
CREATE TABLE `Discord_Servers` (
  `Server_ID` varchar(50) NOT NULL,
  `Server_Name` varchar(50) DEFAULT NULL,
  `Server_Link` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`Server_ID`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Table structure for table `Game_Keys`
--

DROP TABLE IF EXISTS `Game_Keys`;
CREATE TABLE `Game_Keys` (
  `Game_PK` int(11) NOT NULL AUTO_INCREMENT,
  `U_ID` varchar(20) DEFAULT NULL,
  `Server_ID` varchar(20) DEFAULT NULL,
  `Key_name` varchar(50) DEFAULT NULL,
  `Key_price` int(10) DEFAULT NULL,
  `Key_string` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`Game_PK`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Dumping data for table `Game_Keys`
--


LOCK TABLES `Game_Keys` WRITE;
/*!40000 ALTER TABLE `Game_Keys` DISABLE KEYS */;
/*!40000 ALTER TABLE `Game_Keys` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `UBS`
--

DROP TABLE IF EXISTS `UBS`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `UBS` (
  `UBS_PK` int(11) NOT NULL AUTO_INCREMENT,
  `U_ID` int(11) DEFAULT NULL,
  `Server_ID` int(11) DEFAULT NULL,
  `User_Role_Type` int(11) DEFAULT NULL,
```

```sql
  PRIMARY KEY (`UBS_PK`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Dumping data for table `UBS`
--

LOCK TABLES `UBS` WRITE;
/*!40000 ALTER TABLE `UBS` DISABLE KEYS */;
/*!40000 ALTER TABLE `UBS` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `User_Role`
--

DROP TABLE IF EXISTS `User_Role`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `User_Role` (
 `User_Role_ID` int(11) NOT NULL AUTO_INCREMENT,
 `User_Role_Type` int(11) DEFAULT NULL,
 `User_Role_Def` varchar(50) DEFAULT NULL,
 PRIMARY KEY (`User_Role_ID`)
) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=latin1;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `User_Role`
--

DROP TABLE IF EXISTS `Roles`;
CREATE TABLE `Roles`(
    `Role_Name` varchar(20) NOT NULL,
    `User_Role_ID` int(11) NOT NULL,
    `Server_ID` varchar(50) NOT NULL,
   FOREIGN KEY (`User_Role_ID`) REFERENCES User_Role(`User_Role_ID`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

LOCK TABLES `User_Role` WRITE;
/*!40000 ALTER TABLE `User_Role` DISABLE KEYS */;
INSERT INTO `User_Role` VALUES
(1,1,'Admin'),(2,2,'Donor/Recipient'),(3,3,'Donor'),(4,4,'Recipient'),(5,5,'Vie
wer'),(6,6,'Blocked');
/*!40000 ALTER TABLE `User_Role` ENABLE KEYS */;
UNLOCK TABLES;

--
```

```sql
-- Table structure for table `Users`
--

DROP TABLE IF EXISTS `Users`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `Users` (
 `U_ID` varchar(20) NOT NULL,
 `User_Name` varchar(50) DEFAULT NULL,
 `Discord_Token` varchar(50) DEFAULT NULL,
 PRIMARY KEY (`U_ID`)
) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=latin1;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `Users`
--

LOCK TABLES `Users` WRITE;
/*!40000 ALTER TABLE `Users` DISABLE KEYS */;
/*!40000 ALTER TABLE `Users` ENABLE KEYS */;
UNLOCK TABLES;

DROP TABLE IF EXISTS `NotificationSettings`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `NotificationSettings` (
 `Setting_ID` int primary key auto_increment,
 `Notification` boolean,
 `Server_ID` int
) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=latin1;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `Users`
--

LOCK TABLES `NotificationSettings` WRITE;
/*!40000 ALTER TABLE `NotificationSettings` DISABLE KEYS */;
/*!40000 ALTER TABLE `NotificationSettings` ENABLE KEYS */;
UNLOCK TABLES;

DROP TABLE IF EXISTS `Game_Keys`;
CREATE TABLE `Game_Keys` (
 `Game_PK` int(11) NOT NULL AUTO_INCREMENT,
 `U_ID` varchar(20) DEFAULT NULL,
 `Server_ID` varchar(20) DEFAULT NULL,
 `Key_ID` int(10) DEFAULT NULL,
```

```
  `Key_name` varchar(50) DEFAULT NULL,
  `Key_price` int(10) DEFAULT NULL,
  `Key_string` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`Game_PK`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```