Final Project

Jeff Fitih

https://github.com/jfitih/Final_in_SQL

For my final project I would like to have Kodi repos to populate my database with repos to work in my database. To have the data to be retrieved from a Google Drive to run as a server to my URL.

My python code was from the github forum.

<forum>https://github.com/cguZZman/plugin.googledrive/issues</forum>

Which was provided so I can make my own repo for others to bounce off of. I can say I am trying to make a stop shop for all your entertainment needs. Fire has a lot of streaming. But the problem I am have is finding the correct format. Because Apple has a different format than Google and Google has a different format than Android. I want once a person hits my URL they grab my zip drive that will have all add-ons without having my wizard being compromised. I will keep populating SQL with more builds and add-ons.

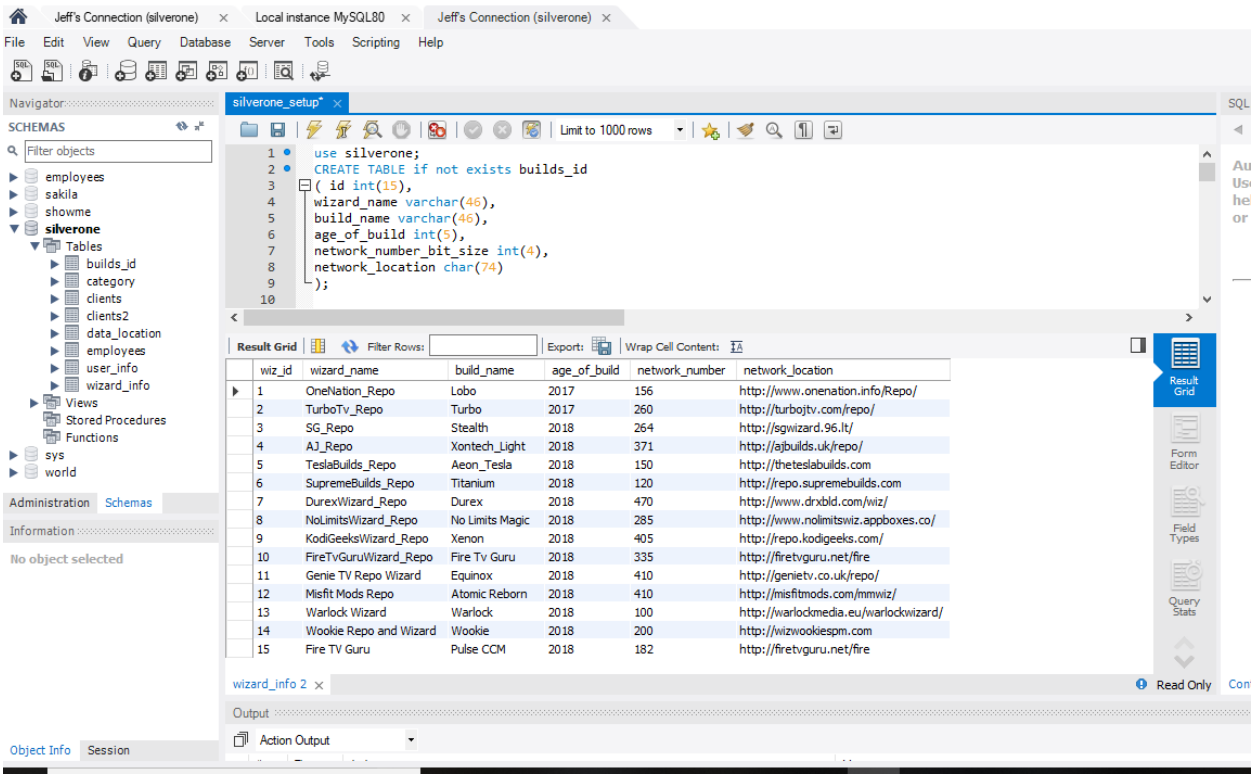
- Open-source software originally designed for Microsoft Xbox
- Can be installed on any media- TV, desktop, laptop, smartphone, and tablet
- Compatible on all platforms, including Mac, Android, Windows, iOS, and Linux
- Used to watch free movies, live streaming of music shows and sporting events, international TV shows, and more
- Download Kodi installation that works for your device
- Download Fusion Installer
- Run configuration wizard, or Addon Installer
- Config Wizard is automated, whereas Addon Installer allows you to choose Kodi Addons before installing them

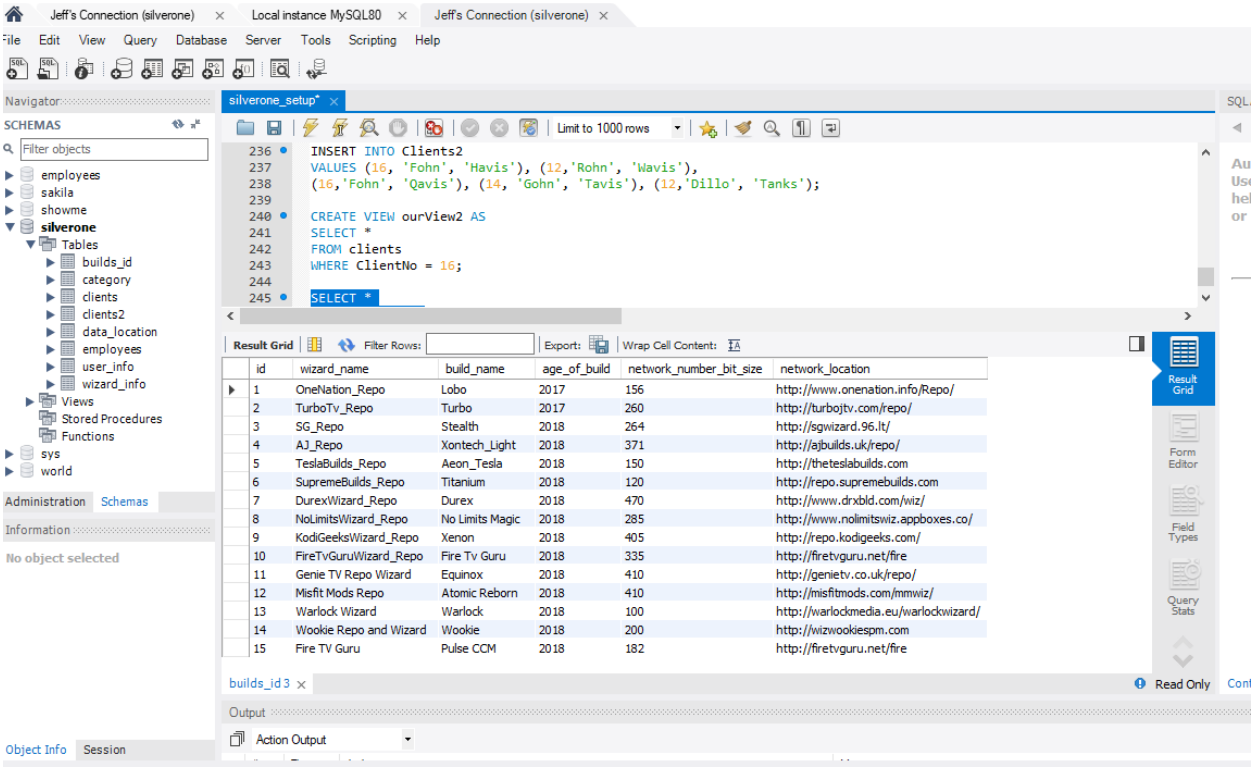
- Takes no more than 15 minutes to install
- Access to thousands of live TV channels
- Unlimited access to international TV shows and live events
- Easy to use
- Safe

- Addons are like apps
- Popular KodiAddons include 1Channel, cCloud- TV, and USTVnow Plus
- Addons are installed under different sections
- Categories include pictures, videos, music, etc.

- Videos can be seen only from a specific year and earlier: This could be because you
- haven’t set the proper date and time on the Kodi.
- No stream available: This has nothing to do with Kodi Addon. You’ll have to search for another stream

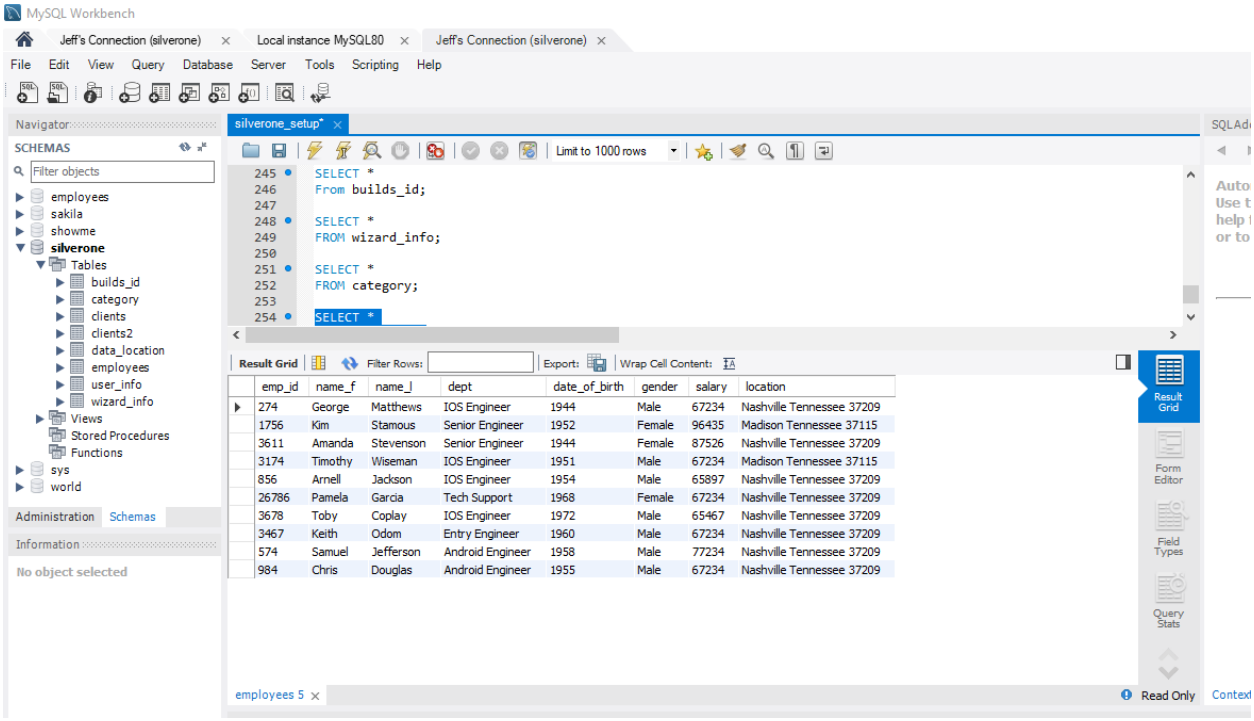
- TV ADDONS offers a platform for several third party developers
- I want to offer thousands of Kodi Addons for users to choose from


Jeff's Connection (silverone)
Local instance MySQL80
Jeff's Connection (silverone)
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
employees
sakila
showme
silverone
Tables
builds_id
category
clients
clients2
data_location
employees
user_info
wizard_info
Views
Stored Procedures
Functions
sys
world
Administration Schemas
Information
No object selected
Object Info Session
silverone_setup*
Limit to 1000 rows
use silverone;
CREATE TABLE if not exists builds_id
(id int(15),
wizard_name varchar(46),
build_name varchar(46),
age_of_build int(5),
network_number_bit_size int(4),
network_location char(74)
);
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
wiz_id wizard_name build_name age_of_build network_number network_location
1 OneNation_Repo Lobo 2017 156 http://www.onenation.info/Repo/
2 TurboTv_Repo Turbo 2017 260 http://turbojtv.com/repo/
3 SG_Repo Stealth 2018 264 http://sgwizard.96.lt/
4 AJ_Repo Xontech_Light 2018 371 http://ajbuilds.uk/repo/
5 TeslaBuilds_Repo Aeon_Tesla 2018 150 http://theteslabuilds.com
6 SupremeBuilds_Repo Titanium 2018 120 http://repo.supremebuilds.com
7 DurexWizard_Repo Durex 2018 470 http://www.drxbld.com/wiz/
8 NoLimitsWizard_Repo No Limits Magic 2018 285 http://www.nolimitswiz.appboxes.co/
9 KodiGeeksWizard_Repo Xenon 2018 405 http://repo.kodigeeks.com/
10 FireTvGuruWizard_Repo Fire Tv Guru 2018 335 http://firetvguru.net/fire
11 Genie TV Repo Wizard Equinox 2018 410 http://genietv.co.uk/repo/
12 Misfit Mods Repo Atomic Reborn 2018 410 http://misfitmods.com/mmwiz/
13 Warlock Wizard Warlock 2018 100 http://warlockmedia.eu/warlockwizard/
14 Wookie Repo and Wizard Wookie 2018 200 http://wizwookiespm.com
15 Fire TV Guru Pulse CCM 2018 182 http://firetvguru.net/fire
Result Grid
Form Editor
Field Types
Query Stats
wizard_info 2
Read Only
Output
Action Output


Jeff's Connection (silverone)
Local instance MySQL80
Jeff's Connection (silverone)
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
employees
sakila
showme
silverone
Tables
builds_id
category
clients
clients2
data_location
employees
user_info
wizard_info
Views
Stored Procedures
Functions
sys
world
Administration Schemas
Information
No object selected
Object Info Session
silverone_setup*
Limit to 1000 rows
INSERT INTO Clients2
VALUES (16, 'Fohn', 'Havis'), (12,'Rohn', 'Wavis'),
(16,'Fohn', 'Qavis'), (14, 'Gohn', 'Tavis'), (12,'Dillo', 'Tanks');
CREATE VIEW ourView2 AS
SELECT *
FROM clients
WHERE ClientNo = 16;
SELECT *
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
id wizard_name build_name age_of_build network_number_bit_size network_location
1 OneNation_Repo Lobo 2017 156 http://www.onenation.info/Repo/
2 TurboTv_Repo Turbo 2017 260 http://turbojtv.com/repo/
3 SG_Repo Stealth 2018 264 http://sgwizard.96.lt/
4 AJ_Repo Xontech_Light 2018 371 http://ajbuilds.uk/repo/
5 TeslaBuilds_Repo Aeon_Tesla 2018 150 http://theteslabuilds.com
6 SupremeBuilds_Repo Titanium 2018 120 http://repo.supremebuilds.com
7 DurexWizard_Repo Durex 2018 470 http://www.drxbld.com/wiz/
8 NoLimitsWizard_Repo No Limits Magic 2018 285 http://www.nolimitswiz.appboxes.co/
9 KodiGeeksWizard_Repo Xenon 2018 405 http://repo.kodigeeks.com/
10 FireTvGuruWizard_Repo Fire Tv Guru 2018 335 http://firetvguru.net/fire
11 Genie TV Repo Wizard Equinox 2018 410 http://genietv.co.uk/repo/
12 Misfit Mods Repo Atomic Reborn 2018 410 http://misfitmods.com/mmwiz/
13 Warlock Wizard Warlock 2018 100 http://warlockmedia.eu/warlockwizard/
14 Wookie Repo and Wizard Wookie 2018 200 http://wizwookiespm.com
15 Fire TV Guru Pulse CCM 2018 182 http://firetvguru.net/fire
Result Grid
Form Editor
Field Types
Query Stats
builds_id 3
Read Only
Output
Action Output

```
use silverone;
CREATE TABLE if not exists builds_id
( id int(15),
wizard_name varchar(46),
build_name varchar(46),
age_of_build int(5),
network_number_bit_size int(4),
network_location char(74)
);

INSERT INTO builds_id(id, wizard_name, build_name, )
VALUES (01, 'OneNation_Repo', 'Lobo', '2017', '156', 'http://www.onenation.info/Repo/'),
(02,'TurboTv_Repo', 'Turbo', '2017', 260, 'http://turbojtv.com/repo/'),
(03,'SG_Repo', 'Stealth', '2018', 264, 'http://sgwizard.96.lt/'),
(04, 'AJ_Repo', 'Xontech_Light', '2018', 371, 'http://ajbuilds.uk/repo/'),
(05,'TeslaBuilds_Repo', 'Aeon_Tesla', '2018', 150, 'http://theteslabuilds.com'),
(06,'SupremeBuilds_Repo', 'Titanium', '2018', 120, 'http://repo.supremebuilds.com'),
(07,'DurexWizard_Repo', 'Durex', '2018', 470, 'http://www.drxbld.com/wiz/'),
(08,'NoLimitsWizard_Repo', 'No Limits Magic', '2018', 285, 'http://www.nolimitswiz.appboxes.co/'),
(09,'KodiGeeksWizard_Repo', 'Xenon', '2018', 405, 'http://repo.kodigeeks.com/'),
(10,'FireTvGuruWizard_Repo', 'Fire Tv Guru', '2018', 335, 'http://firetvguru.net/fire'),
(11,'Genie TV Repo Wizard', 'Equinox', '2018', 410, 'http://genietv.co.uk/repo/'),
(12,'Misfit Mods Repo', 'Atomic Reborn', '2018', 410, 'http://misfitmods.com/mmwiz/'),
(13,'Warlock Wizard', 'Warlock', '2018', 100, 'http://warlockmedia.eu/warlockwizard/'),
(14,'Wookie Repo and Wizard', 'Wookie', '2018', 200, 'http://wizwookiespm.com'),
(15,'Fire TV Guru', 'Pulse CCM', '2018', 182, 'http://firetvguru.net/fire');
```

```
CREATE TABLE if not exists employees
( id int(5),
name_f varchar(46),
dept varchar(46),
age int(2),
salary int(10),
location varchar(46)
);

INSERT INTO builds_id
VALUES (01, 'OneNation_Repo', 'Lobo', '2017', '156', 'http://www.onenation.info/Repo/'),
(02,'TurboTv_Repo', 'Turbo', '2017', 260, 'http://turbojtv.com/repo/'),
(03,'SG_Repo', 'Stealth', '2018', 264, 'http://sgwizard.96.lt/'),
(04, 'AJ_Repo', 'Xontech_Light', '2018', 371, 'http://ajbuilds.uk/repo/'),
(05,'TeslaBuilds_Repo', 'Aeon_Tesla', '2018', 150, 'http://theteslabuilds.com'),
(06,'SupremeBuilds_Repo', 'Titanium', '2018', 120, 'http://repo.supremebuilds.com'),
(07,'DurexWizard_Repo', 'Durex', '2018', 470, 'http://www.drxbld.com/wiz/'),
(08,'NoLimitsWizard_Repo', 'No Limits Magic', '2018', 285, 'http://www.nolimitswiz.appboxes.co/'),
(09,'KodiGeeksWizard_Repo', 'Xenon', '2018', 405, 'http://repo.kodigeeks.com/'),
(10,'FireTvGuruWizard_Repo', 'Fire Tv Guru', '2018', 335, 'http://firetvguru.net/fire'),
(11,'Genie TV Repo Wizard', 'Equinox', '2018', 410, 'http://genietv.co.uk/repo/'),
(12,'Misfit Mods Repo', 'Atomic Reborn', '2018', 410, 'http://misfitmods.com/mmwiz/'),
(13,'Warlock Wizard', 'Warlock', '2018', 100, 'http://warlockmedia.eu/warlockwizard/'),
(14,'Wookie Repo and Wizard', 'Wookie', '2018', 200, 'http://wizwookiespm.com'),
(15,'Fire TV Guru', 'Pulse CCM', '2018', 182, 'http://firetvguru.net/fire');
```



CREATE TABLE if not exists category

(id int(15),

wizard_name varchar(46),

build_name varchar(46),

age_of_build int(5),

network_number int(46),

network_location char(46)

);

INSERT INTO builds_id

VALUES (01, 'OneNation_Repo', 'Lobo', '2017', '156', 'http://www.onenation.info/Repo/'),

(02,'TurboTv_Repo', 'Turbo', '2017', 260, 'http://turbojtv.com/repo/'),

(03,'SG_Repo', 'Stealth', '2018', 264, 'http://sgwizard.96.lt/'),

(04, 'AJ_Repo', 'Xontech_Light', '2018', 371, 'http://ajbuilds.uk/repo/'),

(05,'TeslaBuilds_Repo', 'Aeon_Tesla', '2018', 150, 'http://theteslabuilds.com'),

(06,'SupremeBuilds_Repo', 'Titanium', '2018', 120, 'http://repo.supremebuilds.com'),

(07,'DurexWizard_Repo', 'Durex', '2018', 470, 'http://www.drxbld.com/wiz/'),

(08,'NoLimitsWizard_Repo', 'No Limits Magic', '2018', 285, 'http://www.nolimitswiz.appboxes.co/'),

(09,'KodiGeeksWizard_Repo', 'Xenon', '2018', 405, 'http://repo.kodigeeks.com/'),

(10,'FireTvGuruWizard_Repo', 'Fire Tv Guru', '2018', 335, 'http://firetvguru.net/fire'),

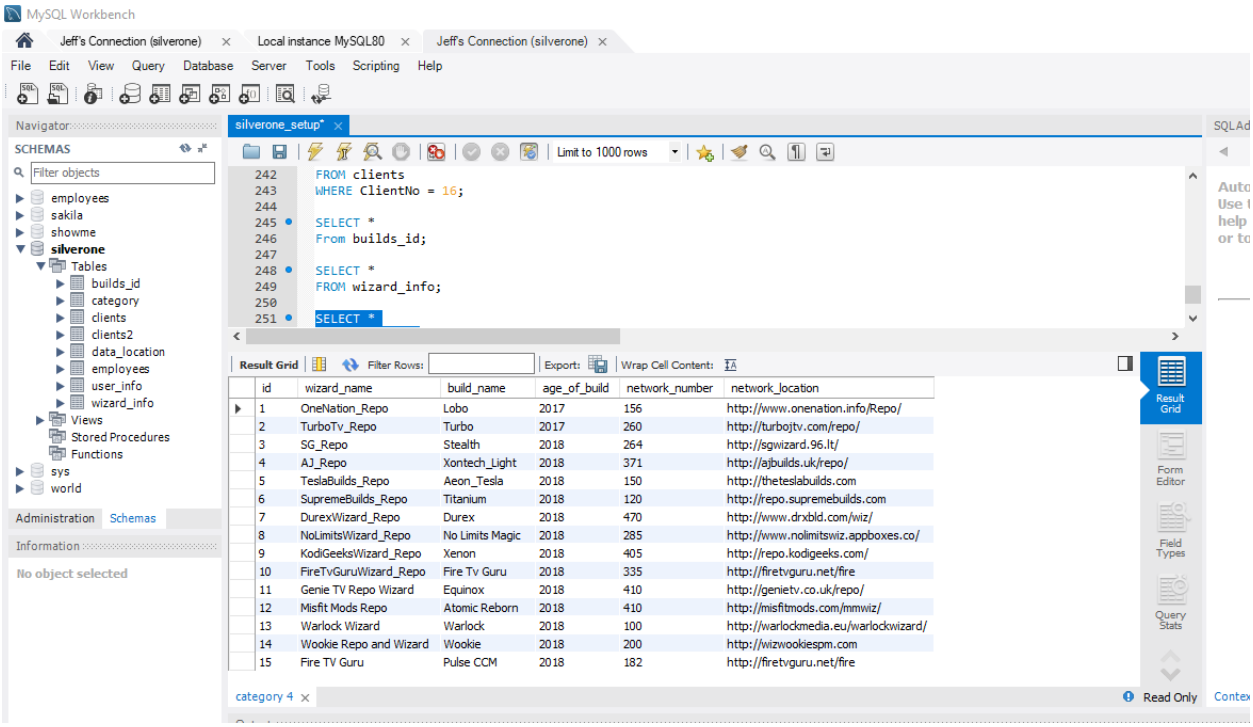
(11,'Genie TV Repo Wizard', 'Equinox', '2018', 410, 'http://genietv.co.uk/repo/'),

(12,'Misfit Mods Repo', 'Atomic Reborn', '2018', 410, 'http://misfitmods.com/mmwiz/'),

(13,'Warlock Wizard', 'Warlock', '2018', 100, 'http://warlockmedia.eu/warlockwizard/'),

(14,'Wookie Repo and Wizard', 'Wookie', '2018', 200, 'http://wizwookiespm.com'),

(15,'Fire TV Guru', 'Pulse CCM', '2018', 182, 'http://firetvguru.net/fire');



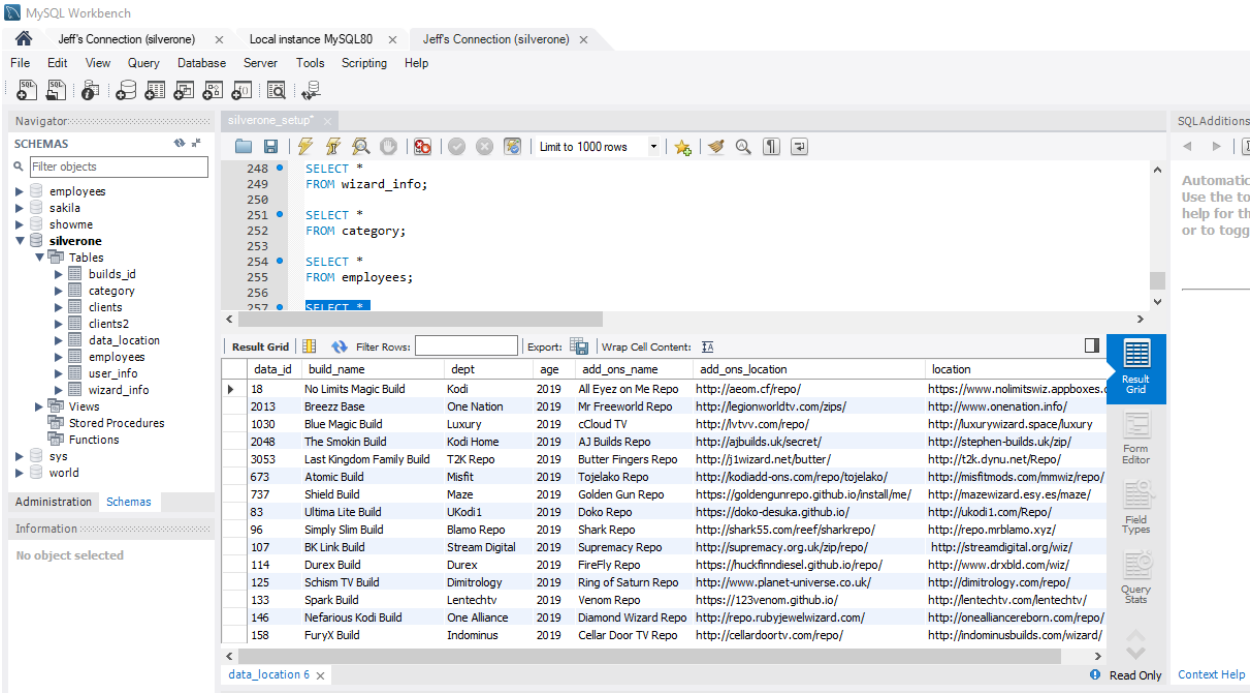
CREATE TABLE if not exists data_location

(data_id int(5),

name_f varchar(46),

dept varchar(46),

```
age int(2),
salary int(10),
location varchar(46)
);

INSERT INTO builds_id
VALUES (01, 'OneNation_Repo', 'Lobo', '2017', '156', 'http://www.onenation.info/Repo/'),
(02,'TurboTv_Repo', 'Turbo', '2017', 260, 'http://turbojtv.com/repo/'),
(03,'SG_Repo', 'Stealth', '2018', 264, 'http://sgwizard.96.lt/'),
(04, 'AJ_Repo', 'Xontech_Light', '2018', 371, 'http://ajbuilds.uk/repo/'),
(05,'TeslaBuilds_Repo', 'Aeon_Tesla', '2018', 150, 'http://theteslabuilds.com'),
(06,'SupremeBuilds_Repo', 'Titanium', '2018', 120, 'http://repo.supremebuilds.com'),
(07,'DurexWizard_Repo', 'Durex', '2018', 470, 'http://www.drxbld.com/wiz/'),
(08,'NoLimitsWizard_Repo', 'No Limits Magic', '2018', 285, 'http://www.nolimitswiz.appboxes.co/'),
(09,'KodiGeeksWizard_Repo', 'Xenon', '2018', 405, 'http://repo.kodigeeks.com/'),
(10,'FireTvGuruWizard_Repo', 'Fire Tv Guru', '2018', 335, 'http://firetvguru.net/fire'),
(11,'Genie TV Repo Wizard', 'Equinox', '2018', 410, 'http://genietv.co.uk/repo/'),
(12,'Misfit Mods Repo', 'Atomic Reborn', '2018', 410, 'http://misfitmods.com/mmwiz/'),
(13,'Warlock Wizard', 'Warlock', '2018', 100, 'http://warlockmedia.eu/warlockwizard/'),
(14,'Wookie Repo and Wizard', 'Wookie', '2018', 200, 'http://wizwookiespm.com'),
(15,'Fire TV Guru', 'Pulse CCM', '2018', 182, 'http://firetvguru.net/fire');
```



CREATE TABLE if not exists wizard_info

(wiz_id int(15),

wizard_name varchar(46),

build_name varchar(46),

age_of_build int(5),

network_number int(46),

network_location varchar(46)

);

INSERT INTO builds_id

VALUES (01, 'OneNation_Repo', 'Lobo', '2017', '156', 'http://www.onenation.info/Repo/'),

(02,'TurboTv_Repo', 'Turbo', '2017', 260, 'http://turbojtv.com/repo/'),

(03,'SG_Repo', 'Stealth', '2018', 264, 'http://sgwizard.96.lt/'),

(04, 'AJ_Repo', 'Xontech_Light', '2018', 371, 'http://ajbuilds.uk/repo/'),

(05,'TeslaBuilds_Repo', 'Aeon_Tesla', '2018', 150, 'http://theteslabuilds.com'),

(06,'SupremeBuilds_Repo', 'Titanium', '2018', 120, 'http://repo.supremebuilds.com'),

(07,'DurexWizard_Repo', 'Durex', '2018', 470, 'http://www.drxbld.com/wiz/'),

(08,'NoLimitsWizard_Repo', 'No Limits Magic', '2018', 285, 'http://www.nolimitswiz.appboxes.co/'),

(09,'KodiGeeksWizard_Repo', 'Xenon', '2018', 405, 'http://repo.kodigeeks.com/'),

(10,'FireTvGuruWizard_Repo', 'Fire Tv Guru', '2018', 335, 'http://firetvguru.net/fire'),

(11,'Genie TV Repo Wizard', 'Equinox', '2018', 410, 'http://genietv.co.uk/repo/'),

(12,'Misfit Mods Repo', 'Atomic Reborn', '2018', 410, 'http://misfitmods.com/mmwiz/'),

(13,'Warlock Wizard', 'Warlock', '2018', 100, 'http://warlockmedia.eu/warlockwizard/'),

(14,'Wookie Repo and Wizard', 'Wookie', '2018', 200, 'http://wizwookiespm.com'),

(15,'Fire TV Guru', 'Pulse CCM', '2018', 182, 'http://firetvguru.net/fire');

CREATE TABLE if not exists user_info

(user_id int(5),

name_f varchar(46),

name_l varchar(46),

username varchar(46),

email varchar(255),

age int(2),

signal_strength int(10),

location varchar(46)

);

INSERT INTO builds_id

VALUES (01, 'OneNation_Repo', 'Lobo', '2017', '156', 'http://www.onenation.info/Repo/'),

(02,'TurboTv_Repo', 'Turbo', '2017', 260, 'http://turbojtv.com/repo/'),

(03,'SG_Repo', 'Stealth', '2018', 264, 'http://sgwizard.96.lt/'),

(04, 'AJ_Repo', 'Xontech_Light', '2018', 371, 'http://ajbuilds.uk/repo/'),

(05,'TeslaBuilds_Repo', 'Aeon_Tesla', '2018', 150, 'http://theteslabuilds.com'),

(06,'SupremeBuilds_Repo', 'Titanium', '2018', 120, 'http://repo.supremebuilds.com'),

(07,'DurexWizard_Repo', 'Durex', '2018', 470, 'http://www.drxbld.com/wiz/'),

(08,'NoLimitsWizard_Repo', 'No Limits Magic', '2018', 285, 'http://www.nolimitswiz.appboxes.co/'),

(09,'KodiGeeksWizard_Repo', 'Xenon', '2018', 405, 'http://repo.kodigeeks.com/'),

(10,'FireTvGuruWizard_Repo', 'Fire Tv Guru', '2018', 335, 'http://firetvguru.net/fire'),

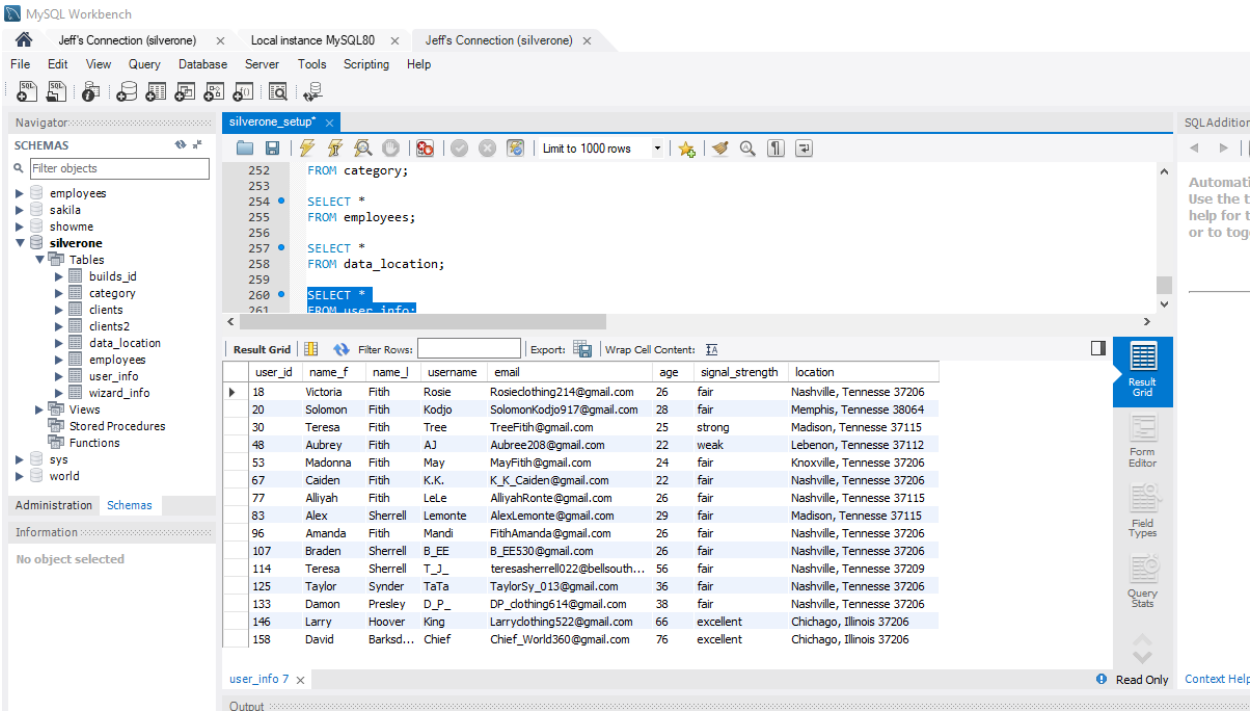
(11,'Genie TV Repo Wizard', 'Equinox', '2018', 410, 'http://genietv.co.uk/repo/'),

(12,'Misfit Mods Repo', 'Atomic Reborn', '2018', 410, 'http://misfitmods.com/mmwiz/'),

(13,'Warlock Wizard', 'Warlock', '2018', 100, 'http://warlockmedia.eu/warlockwizard/'),

(14,'Wookie Repo and Wizard', 'Wookie', '2018', 200, 'http://wizwookiespm.com'),

(15,'Fire TV Guru', 'Pulse CCM', '2018', 182, 'http://firetvguru.net/fire');



CREATE TABLE IF NOT EXISTS Clients (

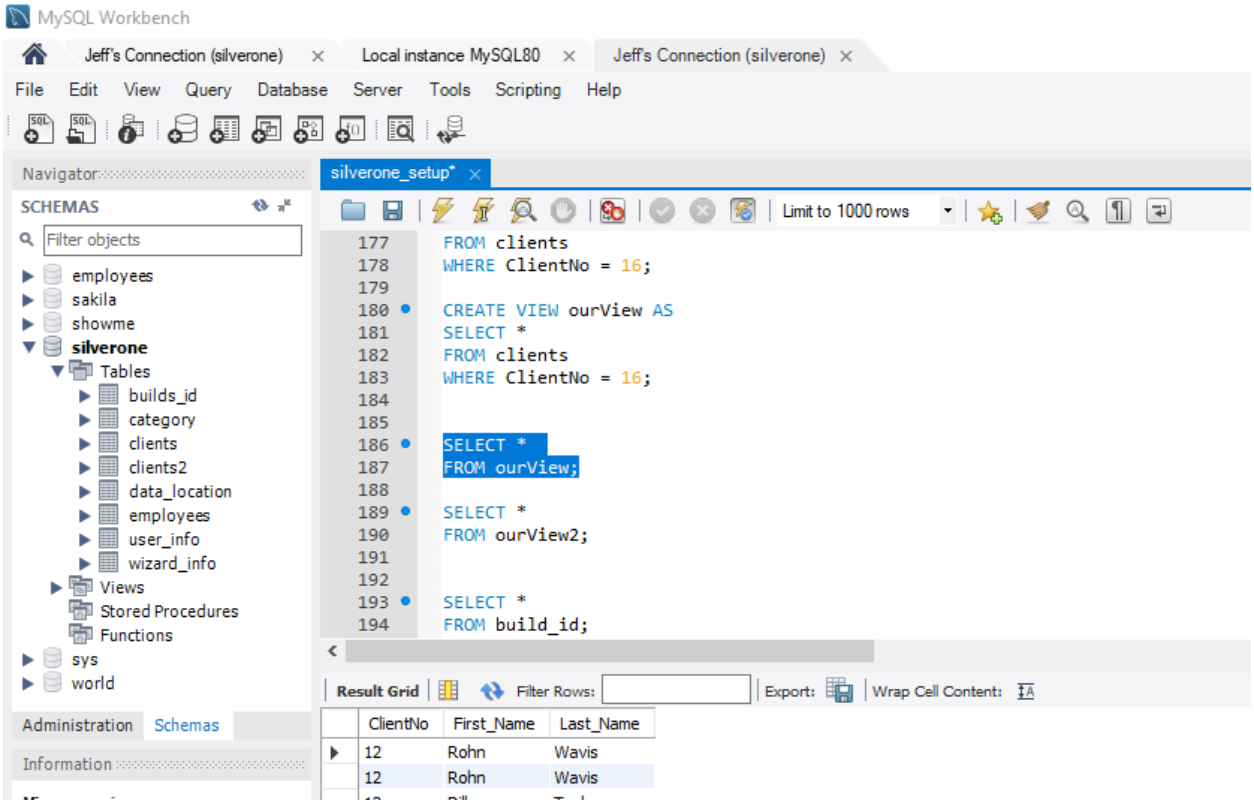
ClientNo INT NOT NULL,

First_Name VARCHAR (45) NULL,

Last_Name VARCHAR (45) NULL

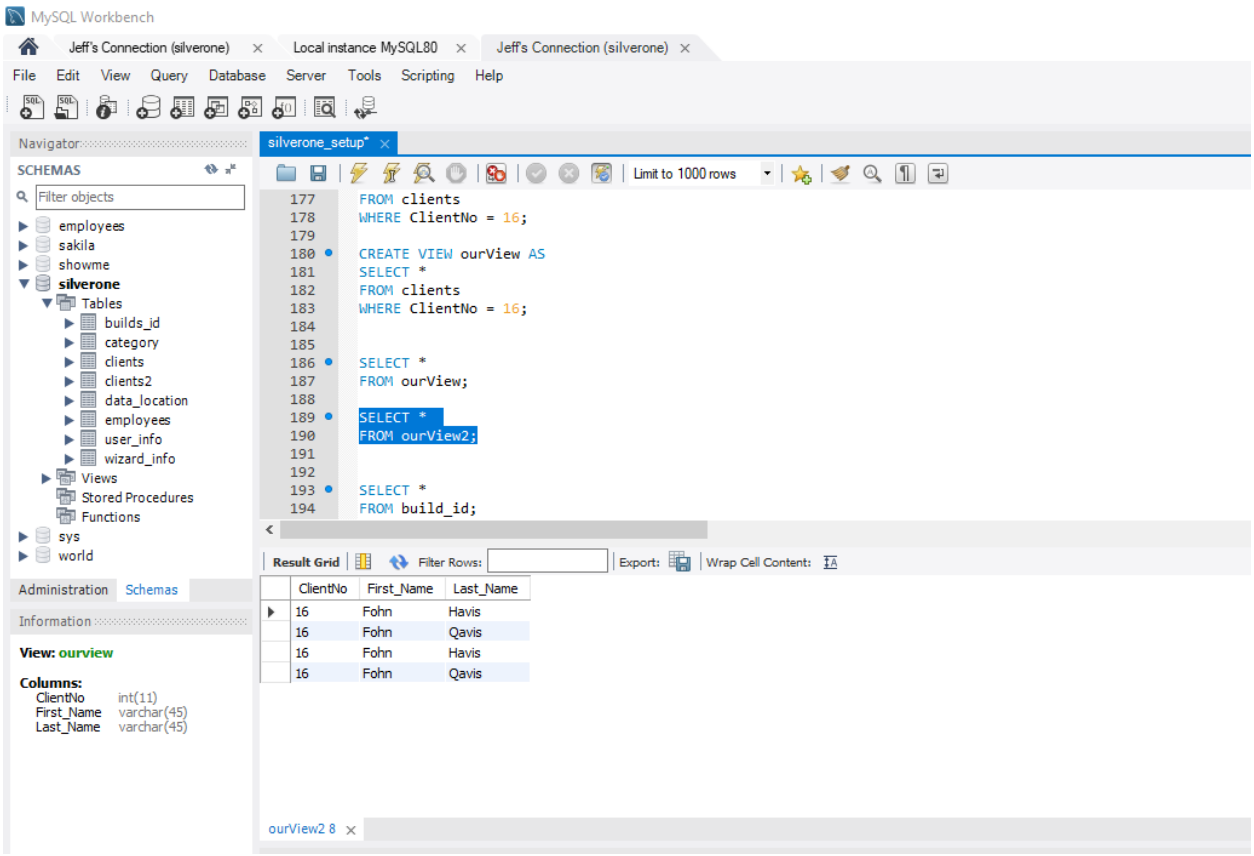
);

```
INSERT INTO Clients(ClientNo, First_Name, Last_Name)
VALUES (16, 'Fohn', 'Havis'), (12,'Rohn', 'Wavis'),
(16,'Fohn', 'Qavis'), (14, 'Gohn', 'Tavis'), (12,'Dillo', 'Tanks');

CREATE VIEW ourView2 AS
SELECT *
FROM clients
WHERE ClientNo = 16;

CREATE VIEW ourView AS
SELECT *
FROM clients
WHERE ClientNo = 16;


SELECT *
FROM ourView;
```



SELECT *

FROM ourView2;



SELECT *

FROM build_id;

SET SQL_SAFE_UPDATES = 0;

INSERT INTO Clients VALUES (NULL, 'Fillo', 'Uanks');

UPDATE Clients

SET ClientNo = 15

WHERE ClientNo = 12;

UPDATE Clients

```
SET ClientNo = 12
WHERE ClientNo = NULL ;

DELETE FROM Clients
WHERE ClientNo = 12;

DROP ourView
DROP Clients;

CREATE TABLE IF NOT EXISTS Clients2 (
ClientNo INT NOT NULL,
First_Name VARCHAR (45) NULL,
Last_Name VARCHAR (45) NULL
);



CREATE VIEW ourView2 AS
SELECT *
FROM clients
WHERE ClientNo = 16;

INSERT INTO Clients2
VALUES (16, 'Fohn', 'Havis'), (12,'Rohn', 'Wavis'),
(16,'Fohn', 'Qavis'), (14, 'Gohn', 'Tavis'), (12,'Dillo', 'Tanks');

CREATE VIEW ourView2 AS
SELECT *
FROM clients
```

```
WHERE ClientNo = 16;

INSERT INTO Clients2
VALUES (16, 'Fohn', 'Havis'), (12,'Rohn', 'Wavis'),
(16,'Fohn', 'Qavis'), (14, 'Gohn', 'Tavis'), (12,'Dillo', 'Tanks');

CREATE VIEW ourView2 AS
SELECT *
FROM clients
WHERE ClientNo = 16;

SELECT *
From builds_id;

SELECT *
FROM wizard_info;
```

Google Drive KODI Addon

Google Drive for KODI

Play all your media from Google Drive including Videos, Music and Pictures.

* Unlimited accounts

* Playback your music and videos. Listing of videos with thumbnails.

* Use Google Drive as a source.

* Subtitles can be assigned automatically if a .str file exists with the same name as the video.

* Export your videos to your library (.strm files). You can export your music too, but kodi won't support it yet. It's a Kodi issue for now.

* Show your photos individually or run a slideshow of them. Listing of pictures with thumbnails.

* Auto-Refreshed slideshow.

* Use of OAuth 2 login. You don't have to write your user/password within the add-on. Use the login process in your browser.

* Extremely fast. Using the Google Drive API

This program is not affiliated with or sponsored by Google.

252 addon.py

@@ -0,0 +1,252 @@

#--

Copyright (C) 2017 Carlos Guzman (cguZZman) carlosguzmang@protonmail.com

#

This file is part of Google Drive for Kodi

#

Google Drive for Kodi is free software: you can redistribute it and/or modify

it under the terms of the GNU General Public License as published by

the Free Software Foundation, either version 3 of the License, or

(at your option) any later version.

#

Cloud Drive Common Module for Kodi is distributed in the hope that it will be useful,

but WITHOUT ANY WARRANTY; without even the implied warranty of

MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the

GNU General Public License for more details.

#

You should have received a copy of the GNU General Public License

along with this program. If not, see <http://www.gnu.org/licenses/>.

#--

import datetime

```
import urllib

from clouddrive.common.cache.simplecache import SimpleCache
from clouddrive.common.ui.addon import CloudDriveAddon
from clouddrive.common.utils import Utils
from resources.lib.provider.googledrive import GoogleDrive
from clouddrive.common.ui.logger import Logger
from urllib2 import HTTPError


class GoogleDriveAddon(CloudDriveAddon):
_provider = GoogleDrive()
_parameters = {'spaces': 'drive', 'prettyPrint': 'false'}
_file_fileds =
'id,name,mimeType,description,hasThumbnail,thumbnailLink,modifiedTime,owners(permissionId),parents,size,imageMediaMetadata(width),videoMediaMetadata'
_cache = None
_child_count_supported = False
_change_token = None

def __init__(self):
self._cache = SimpleCache()
super(GoogleDriveAddon, self).__init__()

def get_provider(self):
return self._provider

def get_my_files_menu_name(self):
return self._addon.getLocalizedString(32013)
```

```
                        def get_custom_drive_folders(self, driveid):
                        self._account_manager.load()
                        drive_folders = []
                        drive_folders.append({'name' : self._common_addon.getLocalizedString(32058),
'path' : 'sharedWithMe'})
                        if self._content_type == 'image':
                        drive_folders.append({'name' : self._addon.getLocalizedString(32007), 'path' :
'photos'})
                        drive_folders.append({'name' : self._addon.getLocalizedString(32014), 'path' :
'starred'})
                        return drive_folders


                        def new_change_token_slideshow(self, change_token, driveid,
item_driveid=None, item_id=None, path=None):
                        self._provider.configure(self._account_manager, driveid)
                        if not change_token:
                        response = self._provider.get('/changes/startPageToken', parameters =
self._parameters)
                        self._change_token = Utils.get_safe_value(response, 'startPageToken')
                        change_token = 1
                        else:
                        Logger.notice('checking change with %s and %s' % (self._change_token,
change_token))
                        page_token = self._change_token
                        while page_token:
                        self._parameters['pageToken'] = page_token
                        self._parameters['fields'] =
'nextPageToken,newStartPageToken,changes(file(id,name,parents))'
                        Logger.notice('%s' % (self._parameters))
                        response = self._provider.get('/changes', parameters = self._parameters)
```

```
                    Logger.notice(response)
                    if self.cancel_operation():
                    return
                    self._change_token = Utils.get_safe_value(response, 'newStartPageToken',
self._change_token)
                    changes = Utils.get_safe_value(response, 'changes', [])
                    for change in changes:
                    f = Utils.get_safe_value(change, 'file', {})
                    Logger.notice(f)
                    parents = Utils.get_safe_value(f, 'parents', [])
                    parents.append(f['id'])
                    if item_id in parents:
                    return change_token + 1
                    page_token = Utils.get_safe_value(response, 'nextPageToken')
                    return change_token


                    def get_folder_items(self, driveid, item_driveid=None, item_id=None,
path=None, on_items_page_completed=None):
                    self._provider.configure(self._account_manager, driveid)
                    item_driveid = Utils.default(item_driveid, driveid)
                    self._parameters['fields'] = 'files(%s),nextPageToken' % self._file_fileds
                    if item_id:
                    self._parameters['q'] = '\'%s\' in parents' % item_id
                    elif path == 'sharedWithMe' or path == 'starred':
                    self._parameters['q'] = path
                    elif path == 'photos':
                    self._parameters['spaces'] = path
                    else:
                    if path == '/':
```

```
                self._parameters['q'] = '\'root\' in parents'
                else:
                item = self.get_item_by_path(path)
                self._parameters['q'] = '\'%s\' in parents' % item['id']
                files = self._provider.get('/files', parameters = self._parameters)
                if self.cancel_operation():
                return
                return self.process_files(driveid, files, on_items_page_completed)

                def search(self, query, driveid, item_driveid=None, item_id=None,
on_items_page_completed=None):
                self._provider.configure(self._account_manager, driveid)
                item_driveid = Utils.default(item_driveid, driveid)
                self._parameters['fields'] = 'files(%s)' % self._file_fileds
                query = 'fullText contains \'%s\'' % query
                if item_id:
                query += ' and \'%s\' in parents' % item_id
                self._parameters['q'] = query
                files = self._provider.get('/files', parameters = self._parameters)
                if self.cancel_operation():
                return
                return self.process_files(driveid, files, on_items_page_completed)

                def process_files(self, driveid, files, on_items_page_completed=None):
                items = []
                for f in files['files']:
                item = self._extract_item(f)
                cache_key = self._addonid+'-drive-'+driveid+'-item_driveid-'+item['drive_id']+'-
item_id-'+item['id']+'-path-None'
```

```
                    self._cache.set(cache_key, f, expiration=datetime.timedelta(minutes=1))
                    items.append(item)
                    if on_items_page_completed:
                    on_items_page_completed(items)
                    if 'nextPageToken' in files:
                    self._parameters['pageToken'] = files['nextPageToken']
                    next_files = self._provider.get('/files', parameters = self._parameters)
                    if self.cancel_operation():
                    return
                    items.extend(self.process_files(driveid, next_files, on_items_page_completed))
                    return items

                    def _extract_item(self, f, include_download_info=False):
                    size = long('%s' % Utils.get_safe_value(f, 'size', 0))
                    item = {
                    'id': f['id'],
                    'name': f['name'],
                    'name_extension' : Utils.get_extension(f['name']),
                    'drive_id' : Utils.get_safe_value(Utils.get_safe_value(f, 'owners', [{}])[0],
'permissionId'),
                    'mimetype' : Utils.get_safe_value(f, 'mimeType'),
                    'last_modified_date' : Utils.get_safe_value(f,'modifiedTime'),
                    'size': size,
                    'description': Utils.get_safe_value(f, 'description', '')
                    }
                    if item['mimetype'] == 'application/vnd.google-apps.folder':
                    item['folder'] = {
                    'child_count' : 0
                    }
```

```
if 'videoMediaMetadata' in f:
video = f['videoMediaMetadata']
item['video'] = {
'width' : Utils.get_safe_value(video, 'width'),
'height' : Utils.get_safe_value(video, 'height'),
'duration' : long('%s' % Utils.get_safe_value(video, 'durationMillis', 0)) / 1000
}
if 'audio' in f:
audio = f['audio']
item['audio'] = {
'tracknumber' : Utils.get_safe_value(audio, 'track'),
'discnumber' : Utils.get_safe_value(audio, 'disc'),
'duration' : int(Utils.get_safe_value(audio, 'duration') or '0') / 1000,
'year' : Utils.get_safe_value(audio, 'year'),
'genre' : Utils.get_safe_value(audio, 'genre'),
'album': Utils.get_safe_value(audio, 'album'),
'artist': Utils.get_safe_value(audio, 'artist'),
'title': Utils.get_safe_value(audio, 'title')
}
if 'imageMediaMetadata' in f:
item['image'] = {
'size' : size
}
if 'hasThumbnail' in f and f['hasThumbnail']:
item['thumbnail'] = Utils.get_safe_value(f, 'thumbnailLink')
if include_download_info:
item['download_info'] = {
'url' : self._provider._get_api_url() + '/files/%s?alt=media&access_token=%s' %
(f['id'], self._provider.get_access_tokens()['access_token'])
```

```
}
return item

def get_item_by_path(self, path, include_download_info=False):
if path[:1] == '/':
path = path[1:]
if path[-1:] == '/':
path = path[:-1]
Logger.notice('get_item_by_path: %s' % path)
parts = path.split('/')
parent = 'root'
current_path = ''
item = None
self._parameters['fields'] = 'files(%s)' % self._file_fileds
for part in parts:
part = urllib.unquote(part)
current_path += '/%s' % part
self._parameters['q'] = '\'%s\' in parents and name = \'%s\'' % (parent, part)
files = self._provider.get('/files', parameters = self._parameters)
if (len(files['files']) > 0):
for f in files['files']:
item = self._extract_item(f, include_download_info)
parent = item['id']
cache_key = self._addonid+'-drive-None-item_driveid-None-item_id-None-path-
'+current_path
self._cache.set(cache_key, f, expiration=datetime.timedelta(minutes=1))
break
else:
item = None
```

```
                        break

                        if not item:

                        raise HTTPError(path, 404, 'Not found', None, None)

                        return item


                        def get_item(self, driveid, item_driveid=None, item_id=None, path=None,
find_subtitles=False, include_download_info=False):

                        self._provider.configure(self._account_manager, driveid)

                        item_driveid = Utils.default(item_driveid, driveid)

                        cache_key = self._addonid+'-drive-'+driveid+'-item_driveid-
'+Utils.str(item_driveid)+'-item_id-'+Utils.str(item_id)+'-path-'+Utils.str(path)

                        f = self._cache.get(cache_key)

                        if f:

                        item = self._extract_item(f, include_download_info)

                        else:

                        cache_key = self._addonid+'-drive-None-item_driveid-None-item_id-None-path-
'+path

                        f = self._cache.get(cache_key)

                        if f:

                        item = self._extract_item(f, include_download_info)

                        else:

                        self._parameters['fields'] = self._file_fileds

                        if not item_id and path == '/':

                        item_id = 'root'

                        if item_id:

                        f = self._provider.get('/files/%s' % item_id, parameters = self._parameters)

                        self._cache.set(cache_key, f, expiration=datetime.timedelta(seconds=59))

                        item = self._extract_item(f, include_download_info)

                        else:

                        item = self.get_item_by_path(path, include_download_info)
```

```
                if find_subtitles:
                subtitles = []
                self._parameters['q'] = 'name contains \'%s\'' %
urllib.quote(Utils.remove_extension(item['name']))
                files = self._provider.get('/files', parameters = self._parameters)
                for f in files['files']:
                subtitle = self._extract_item(f, include_download_info)
                if subtitle['name_extension'] == 'srt' or subtitle['name_extension'] == 'sub' or
subtitle['name_extension'] == 'sbv':
                subtitles.append(subtitle)
                if subtitles:
                item['subtitles'] = subtitles
                return item


                if __name__ == '__main__':
                GoogleDriveAddon().route()
```

38 addon.xml

```
        @@ -0,0 +1,38 @@
                <?xml version="1.0" encoding="UTF-8"?>
                <addon id="plugin.googledrive" name="Google Drive" version="1.0.0" provider-
name="Carlos Guzman (cguZZman)">
                <requires>
                <import addon="xbmc.python" version="2.25.0" />
                <import addon="script.module.clouddrive.common"/>
                </requires>
                <extension point="xbmc.python.pluginsource" library="addon.py">
                <provides>image audio video</provides>
                </extension>
```

<extension point="xbmc.service" library="service.py" start="login" />

<extension point="xbmc.addon.metadata">

<platform>all</platform>

<summary lang="en_GB">Google Drive for KODI</summary>

<description lang="en_GB">

Play all your media from Google Drive including Videos, Music and Pictures.

- Unlimited number of personal or business accounts.

- Search over your drive.

- Auto-Refreshed slideshow.

- Export your videos to your library (.strm files)

- Use Google Drive as a source

- This program is not affiliated with or sponsored by Google.

</description>

<summary lang="he_IL">בונן Google Drive עבור קודי של מיקרוסופט</summary>

<description lang="he_IL">

הפעל את כל המדיה שלך מ- Google Drive ותמונות, מוסיקה וידאו בולל.

- מספר בלתי מוגבל של חשבונות אישיים או עסקיים.

- חיפוש ומציאת הבונן שלך.

- ריענון מצגת אוטומטית.

- ייצוא קטעי הווידאו לספרייה שלך (קבצי .strm)

- שימוש ב-Google Drive במקור

- תוכנית זו אינה מזוהה עם או בחסות מיקרוסופט.

</description>

<license>GNU GENERAL PUBLIC LICENSE. Version 3, 29 June 2007</license>

<source>https://github.com/cguZZman/plugin.googledrive</source>

<forum>https://github.com/cguZZman/plugin.googledrive/issues</forum>

<website>https://addons.kodi.tv/show/plugin.googledrive</website>

</extension>

</addon>

build_id	NEEDED
id	FK
wizard_name	PK
build_name	PK
age_of_build	Null
network_number	Null
network_location	FK

employees	NEEDED
emp_id	FK
name_f	PK
name_l	PK
dept	PK
date_of_birth	Null
gender	FK
salary	FK
location	FK

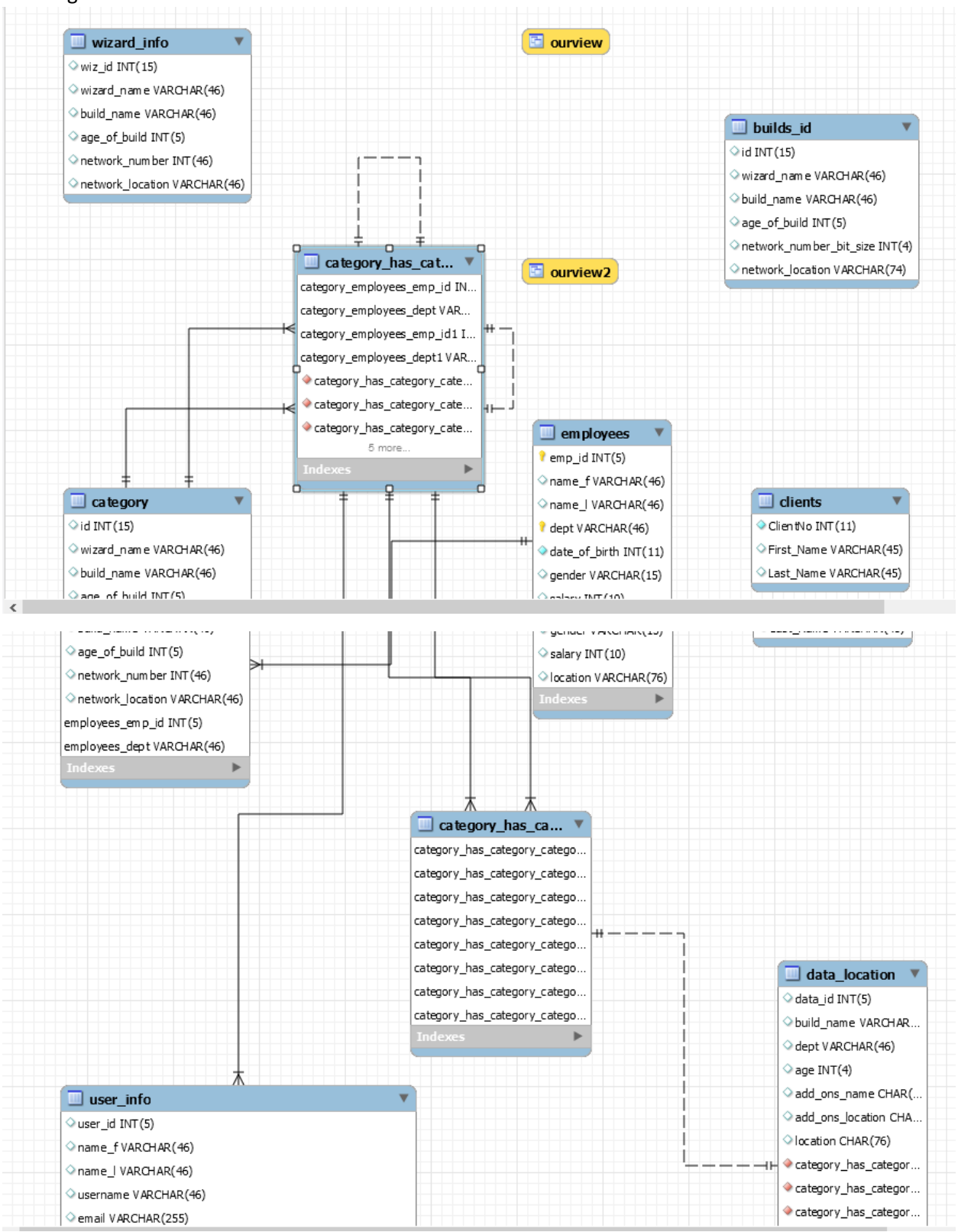
wizard_info	NEEDED
id	FK
wizard_name	PK
build_name	PK
age_of_build	Null
network_number	Null
network_location	FK

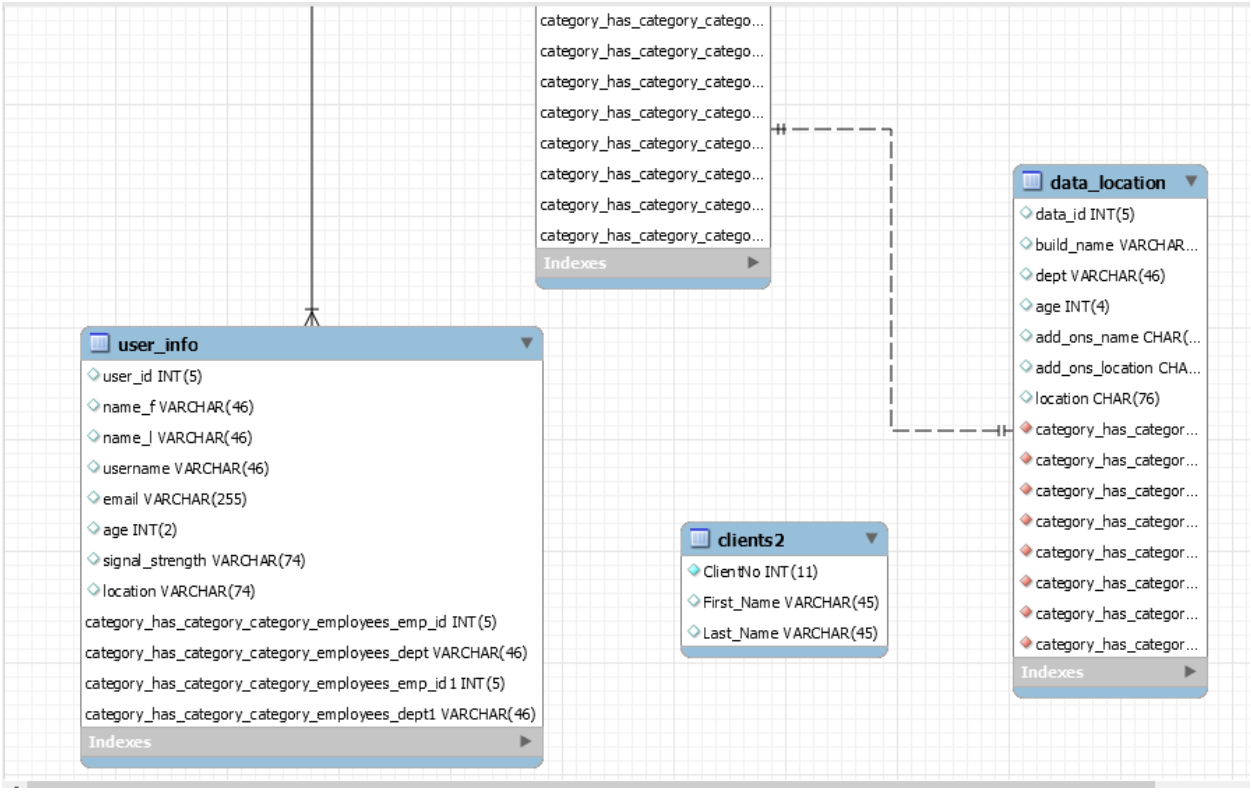
user_info	NEEDED
user_id	FK
name_f	PK
name_l	PK
username	Null
email	Null
age	FK
signal_strength	Null
Location	FK

category	NEEDED
id	FK
wizard_name	PK
build_name	PK
age_of_build	Null
network_number	Null
network_location	FK

data_location	NEEDED
data_id	FK
build_name	PK
dept	PK
add_ons_name	Null
add_ons_loation	Null
location	FK

E-R Diagram


wizard_info
wiz_id INT(15)
wizard_name VARCHAR(46)
build_name VARCHAR(46)
age_of_build INT(5)
network_number INT(46)
network_location VARCHAR(46)
ourview
builds_id
id INT(15)
wizard_name VARCHAR(46)
build_name VARCHAR(46)
age_of_build INT(5)
network_number_bit_size INT(4)
network_location VARCHAR(74)
category_has_cat...
category_employees_emp_id IN...
category_employees_dept VAR...
category_employees_emp_id1 I...
category_employees_dept1 VAR...
category_has_category_cate...
category_has_category_cate...
category_has_category_cate...
5 more...
Indexes
ourview2
employees
emp_id INT(5)
name_f VARCHAR(46)
name_l VARCHAR(46)
dept VARCHAR(46)
date_of_birth INT(11)
gender VARCHAR(15)
category
id INT(15)
wizard_name VARCHAR(46)
build_name VARCHAR(46)
clients
ClientNo INT(11)
First_Name VARCHAR(45)
Last_Name VARCHAR(45)
age_of_build INT(5)
network_number INT(46)
network_location VARCHAR(46)
employees_emp_id INT(5)
employees_dept VARCHAR(46)
Indexes
salary INT(10)
location VARCHAR(76)
Indexes
category_has_ca...
category_has_category_catego...
category_has_category_catego...
category_has_category_catego...
category_has_category_catego...
category_has_category_catego...
category_has_category_catego...
category_has_category_catego...
category_has_category_catego...
Indexes
data_location
data_id INT(5)
build_name VARCHAR...
dept VARCHAR(46)
age INT(4)
add_ons_name CHAR(...
add_ons_location CHA...
location CHAR(76)
category_has_categor...
category_has_categor...
category_has_categor...
user_info
user_id INT(5)
name_f VARCHAR(46)
name_l VARCHAR(46)
username VARCHAR(46)
email VARCHAR(255)



Scheme

build_id

id	wizard_name	build_name	age_of_build	network_number	network_location

categorey

id	wizard_name	build_name	age_of_build	network_number	network_location

user_info

user_id	name_f	name_l	username	email	age	strength signal	location

(int(5), varchar(46), varchar(46), varchar(46), varchar(255),int(2), int(10), varchar(46)

data_location

data_id	name_f	dept	age	salary	location

int(5), varchar(46), varchar(46), int(2), int(10),

wizard_info

wiz_id	wizard_name	build_name	age_of_build	network_number	network_location

int(15), varchar(46), varchar(46), int(5), int(46), varchar(46)