Subject: Homework #5 Advanced Database

Farah Shleemon Kamw
Fkamw@kent.edu

1. From the data folder on your repository I got **ChinookDatabase1.4_Sqlite.zip.** I extracted this folder, then I created the chinook database by running this command (sqlite3 data/chinook. db < data/ Chinook_Sqlite.sql) on command prompt (see the Fig. 1). I did the HW4 in Lecture#8 slides. I opened chinook database using sqlite3 commands in command prompt. I saw all the tables and the schema. I did some simple queries to know the number of items in each table (see Fig. 2 and Fig. 3). Then using python I created program to connect to chinook.db, then I created a complex query that joins three tables (Track, Album, and Artist), this query gave me about 3045 items ,then I did aggregation query to filter the result of the previous query to ten items. I found the time of each query. Then I did the optimization into two levels, at the beginning I added only two indexes each one with single columns, in the next level of the optimization I added two indexes one with multiple columns and the other with single column. I used the (Explain query plan) to verify the change. Also I computed the average time and standard deviation of the computations time of implementing the queries with and without indexes 100 times. Then I plotted the result using Line Graph. The results show me that the query with indexes is faster than the query without indexes(see the results in Fig.4 and Fig.5).You can also see my code on my git hub account using this Link
https://github.com/farahshleemon/Advanced-DataBase_HWs/tree/master/HW%234

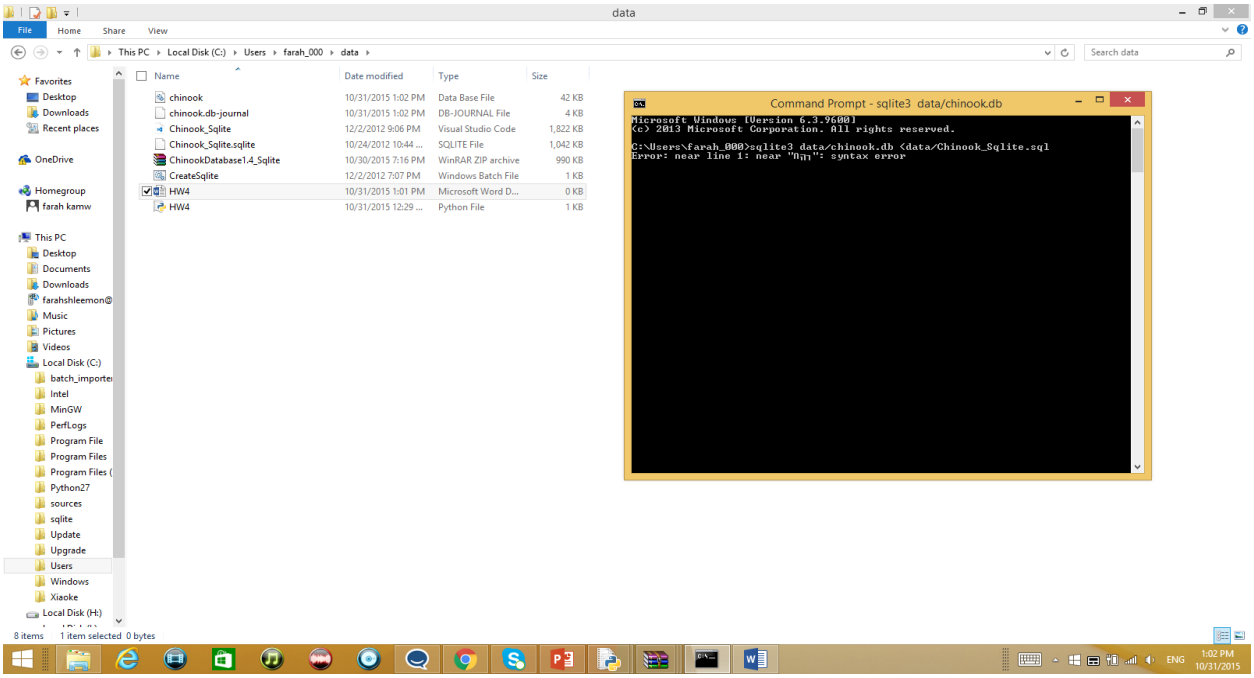

Fig. 1

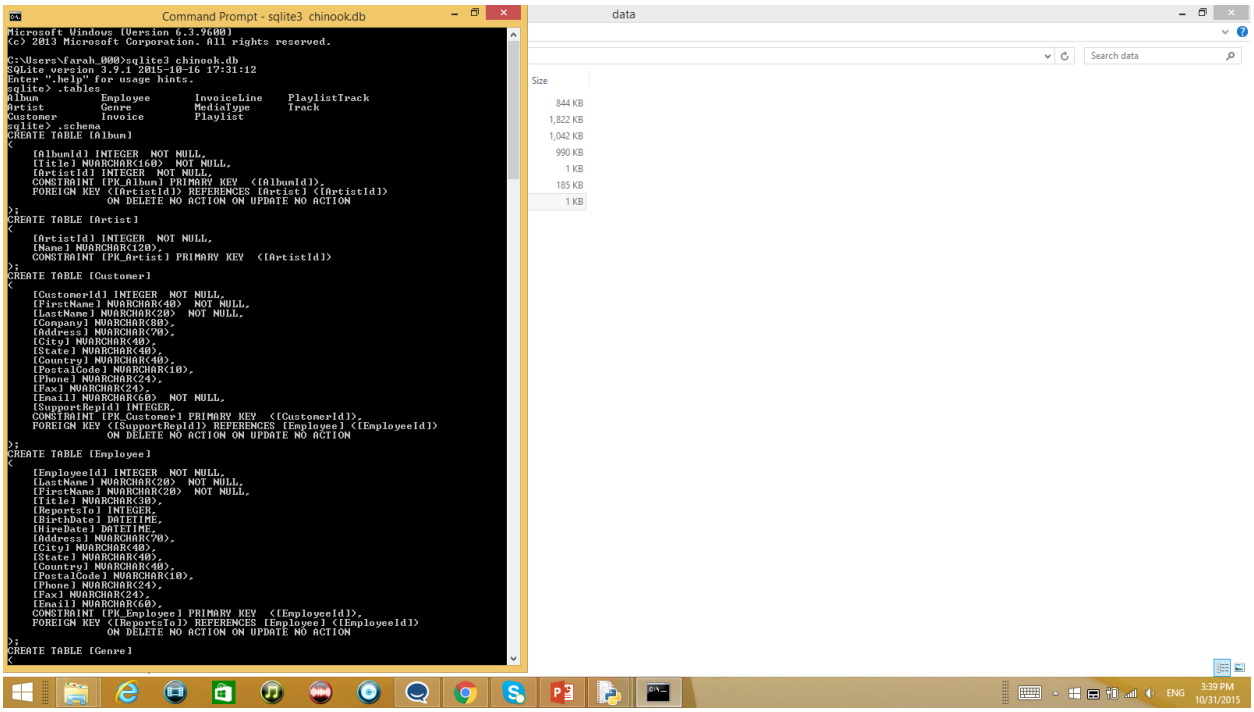
Command Prompt - sqlite3 chinook.db
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.
C:\Users\farah_000>sqlite3 chinook.db
SQLite version 3.9.1 2015-10-16 17:31:12
Enter ".help" for usage hints.
sqlite> .tables
Album Employee InvoiceLine PlaylistTrack
Artist Genre MediaType Track
Customer Invoice Playlist
sqlite> .schema
CREATE TABLE [Album]
(
[AlbumId] INTEGER NOT NULL,
[Title] NVARCHAR(160) NOT NULL,
[ArtistId] INTEGER NOT NULL,
CONSTRAINT [PK_Album] PRIMARY KEY ([AlbumId]),
FOREIGN KEY ([ArtistId]) REFERENCES [Artist] ([ArtistId])
ON DELETE NO ACTION ON UPDATE NO ACTION
);
CREATE TABLE [Artist]
(
[ArtistId] INTEGER NOT NULL,
[Name] NVARCHAR(120),
CONSTRAINT [PK_Artist] PRIMARY KEY ([ArtistId])
);
CREATE TABLE [Customer]
(
[CustomerId] INTEGER NOT NULL,
[FirstName] NVARCHAR(40) NOT NULL,
[LastName] NVARCHAR(20) NOT NULL,
[Company] NVARCHAR(80),
[Address] NVARCHAR(70),
[City] NVARCHAR(40),
[State] NVARCHAR(40),
[Country] NVARCHAR(40),
[PostalCode] NVARCHAR(10),
[Phone] NVARCHAR(24),
[Fax] NVARCHAR(24),
[Email] NVARCHAR(60) NOT NULL,
[SupportRepId] INTEGER,
CONSTRAINT [PK_Customer] PRIMARY KEY ([CustomerId]),
FOREIGN KEY ([SupportRepId]) REFERENCES [Employee] ([EmployeeId])
ON DELETE NO ACTION ON UPDATE NO ACTION
);
CREATE TABLE [Employee]
(
[EmployeeId] INTEGER NOT NULL,
[LastName] NVARCHAR(20) NOT NULL,
[FirstName] NVARCHAR(20) NOT NULL,
[Title] NVARCHAR(30),
[ReportsTo] INTEGER,
[BirthDate] DATETIME,
[HireDate] DATETIME,
[Address] NVARCHAR(70),
[City] NVARCHAR(40),
[State] NVARCHAR(40),
[Country] NVARCHAR(40),
[PostalCode] NVARCHAR(10),
[Phone] NVARCHAR(24),
[Fax] NVARCHAR(24),
[Email] NVARCHAR(60),
CONSTRAINT [PK_Employee] PRIMARY KEY ([EmployeeId]),
FOREIGN KEY ([ReportsTo]) REFERENCES [Employee] ([EmployeeId])
ON DELETE NO ACTION ON UPDATE NO ACTION
);
CREATE TABLE [Genre]
(
data
Search data
Size
844 KB
1,822 KB
1,042 KB
990 KB
1 KB
185 KB
1 KB
ENG
3:39 PM
10/31/2015

Fig.2

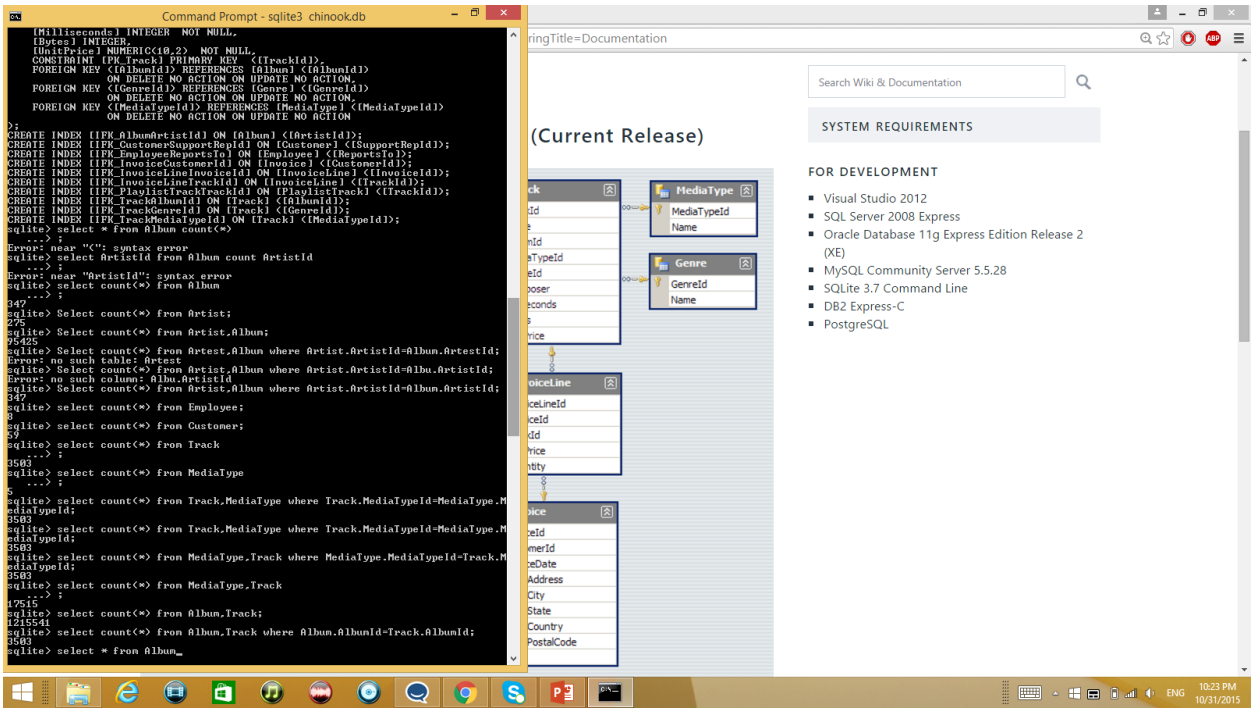
Command Prompt - sqlite3 chinook.db
[Milliseconds] INTEGER NOT NULL,
[Bytes] INTEGER,
[UnitPrice] NUMERIC(10,2) NOT NULL,
CONSTRAINT [PK_Track] PRIMARY KEY ([TrackId]),
FOREIGN KEY ([AlbumId]) REFERENCES [Album] ([AlbumId])
ON DELETE NO ACTION ON UPDATE NO ACTION,
FOREIGN KEY ([GenreId]) REFERENCES [Genre] ([GenreId])
ON DELETE NO ACTION ON UPDATE NO ACTION,
FOREIGN KEY ([MediaTypeId]) REFERENCES [MediaType] ([MediaTypeId])
ON DELETE NO ACTION ON UPDATE NO ACTION
);
CREATE INDEX [IFK_AlbumArtistId] ON [Album] ([ArtistId]);
CREATE INDEX [IFK_CustomerSupportRepId] ON [Customer] ([SupportRepId]);
CREATE INDEX [IFK_EmployeeReportsTo] ON [Employee] ([ReportsTo]);
CREATE INDEX [IFK_InvoiceCustomerId] ON [Invoice] ([CustomerId]);
CREATE INDEX [IFK_InvoiceLineInvoiceId] ON [InvoiceLine] ([InvoiceId]);
CREATE INDEX [IFK_InvoiceLineTrackId] ON [InvoiceLine] ([TrackId]);
CREATE INDEX [IFK_PlaylistTrackTrackId] ON [PlaylistTrack] ([TrackId]);
CREATE INDEX [IFK_TrackAlbumId] ON [Track] ([AlbumId]);
CREATE INDEX [IFK_TrackGenreId] ON [Track] ([GenreId]);
CREATE INDEX [IFK_TrackMediaTypeId] ON [Track] ([MediaTypeId]);
sqlite> select * from Album count(*)
...> ;
Error: near "(": syntax error
sqlite> select ArtistId from Album count ArtistId
...> ;
Error: near "ArtistId": syntax error
sqlite> select count(*) from Album
...> ;
347
sqlite> Select count(*) from Artist;
275
sqlite> Select count(*) from Artist,Album;
95425
sqlite> Select count(*) from Artest,Album where Artist.ArtistId=Album.ArtestId;
Error: no such table: Artest
sqlite> Select count(*) from Artist,Album where Artist.ArtistId=Albu.ArtistId;
Error: no such column: Albu.ArtistId
sqlite> Select count(*) from Artist,Album where Artist.ArtistId=Album.ArtistId;
347
sqlite> select count(*) from Employee;
8
sqlite> select count(*) from Customer;
59
sqlite> select count(*) from Track
...> ;
3503
sqlite> select count(*) from MediaType
...> ;
5
sqlite> select count(*) from Track,MediaType where Track.MediaTypeId=MediaType.MediaTypeId;
3503
sqlite> select count(*) from Track,MediaType where Track.MediaTypeId=MediaType.MediaTypeId;
3503
sqlite> select count(*) from MediaType,Track where MediaType.MediaTypeId=Track.MediaTypeId;
3503
sqlite> select count(*) from MediaType,Track
...> ;
17515
sqlite> select count(*) from Album,Track;
1215541
sqlite> select count(*) from Album,Track where Album.AlbumId=Track.AlbumId;
3503
sqlite> select * from Album_
ringTitle=Documentation
Search Wiki & Documentation
(Current Release)
SYSTEM REQUIREMENTS
FOR DEVELOPMENT
Visual Studio 2012
SQL Server 2008 Express
Oracle Database 11g Express Edition Release 2 (XE)
MySQL Community Server 5.5.28
SQLite 3.7 Command Line
DB2 Express-C
PostgreSQL
MediaType
MediaTypeId
Name
Genre
GenreId
Name
Address
City
State
Country
PostalCode
ENG
10:23 PM
10/31/2015

Fig. 3

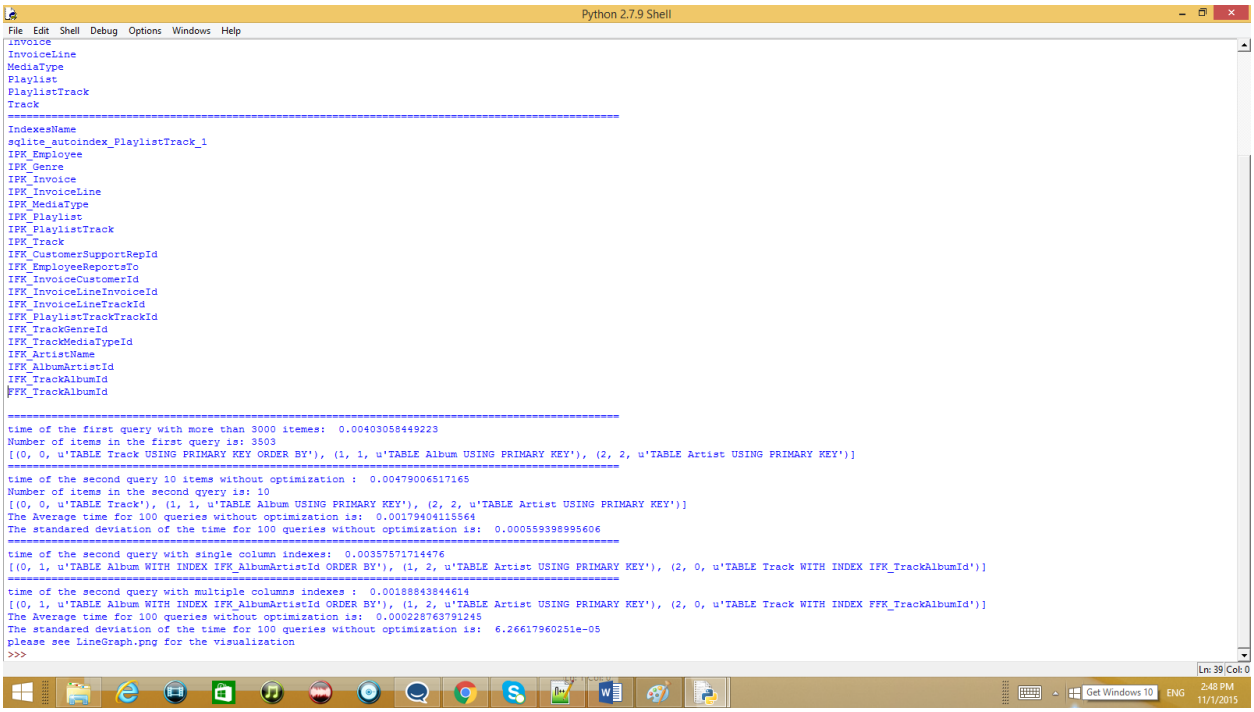

Fig.4

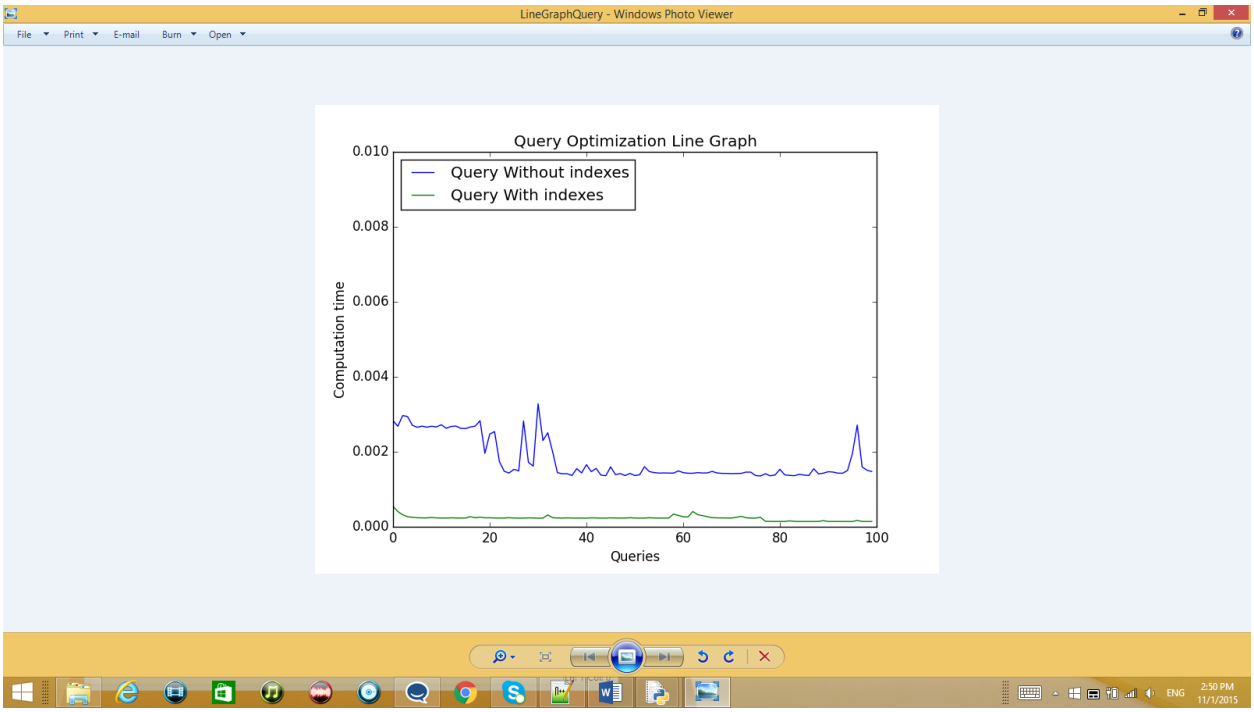

Fig.5

2. Using (git clone https://github.com/gregdelozier/advanced_database_class.git) I cloned your Advanced Database class repository. Then from lecture_8 folder I got (example. Peewee) folder then I run app.py code on my machine (see the figures 6, 7, 8, and 9). Also, I read about the tiny DB.

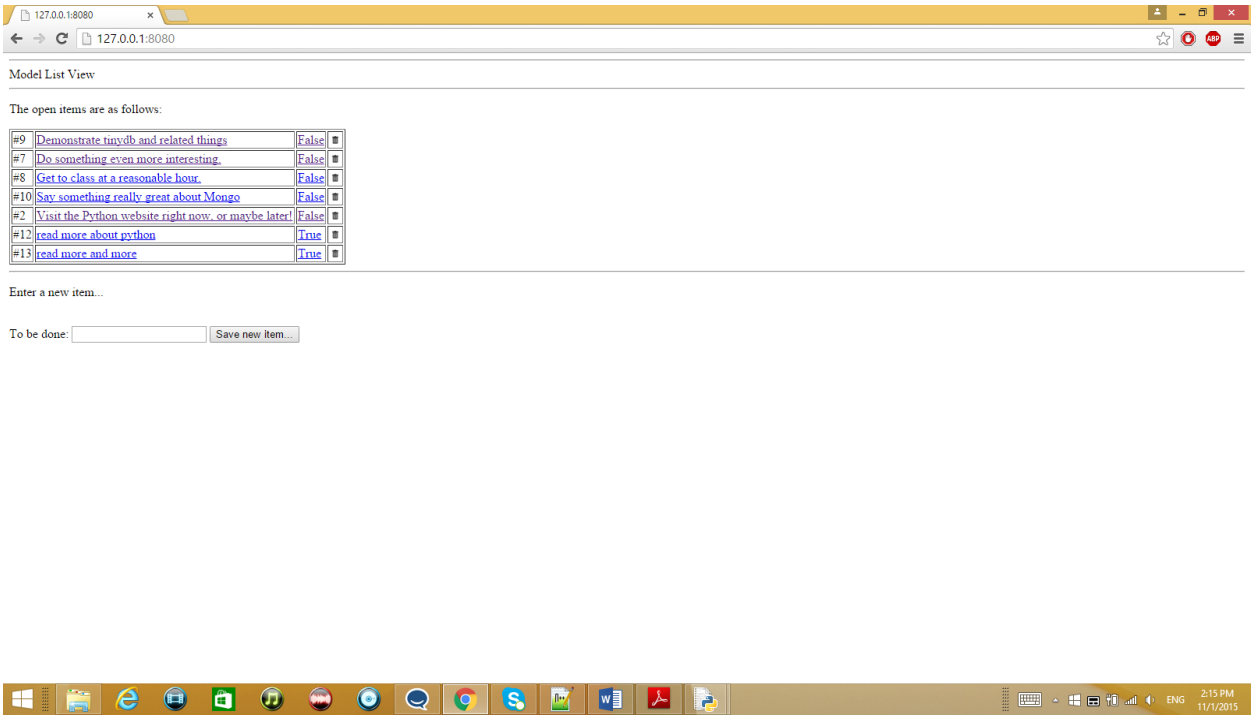

Fig.6

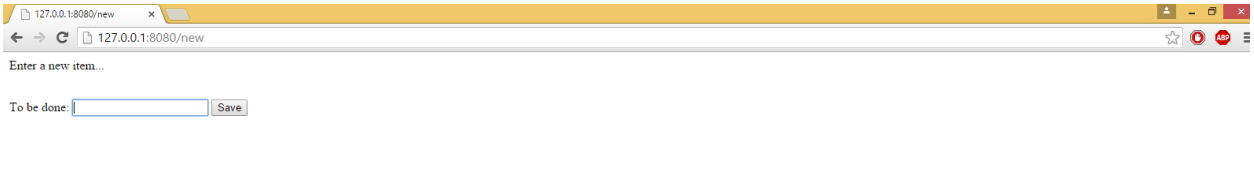

Fig.7

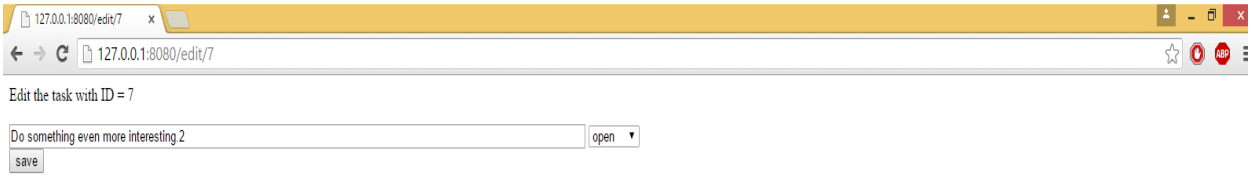

Fig.8

Fig.9