2013

Final Project

ITMD 411

GUI Application with DB connectivity to perform CRUD operations along with Java Data Analytics and Serialization

SagarikaMuniraj
IIT, Chicago
5/10/2013

Table of Contents

Project Abstract

This project is creating using the following

- Java for the Service layer
- Swing for the GUI
- MySQL Database

The project is intended to perform the following functionalities

- Read from CSV file and load data into the Database
- Demonstrate the high level CRUD operations by connecting to the Database from the Java layer
- Perform a few Data Analytics in Java – Finding Solstices and Equinoxes
- Serializing and Deserializing data

The project demonstrates easy usability and is user friendly. UI is intuitive even to a newbie user.

System Requirements

- OS – Windows 7
- IDE – NetBeans 7.2.1
- JDK – 1.7.0
- JRE Version 6
- MySQL 5.2.47
- MYSQL Connector - mysql-connector-java-5.1.24-bin

Data Model

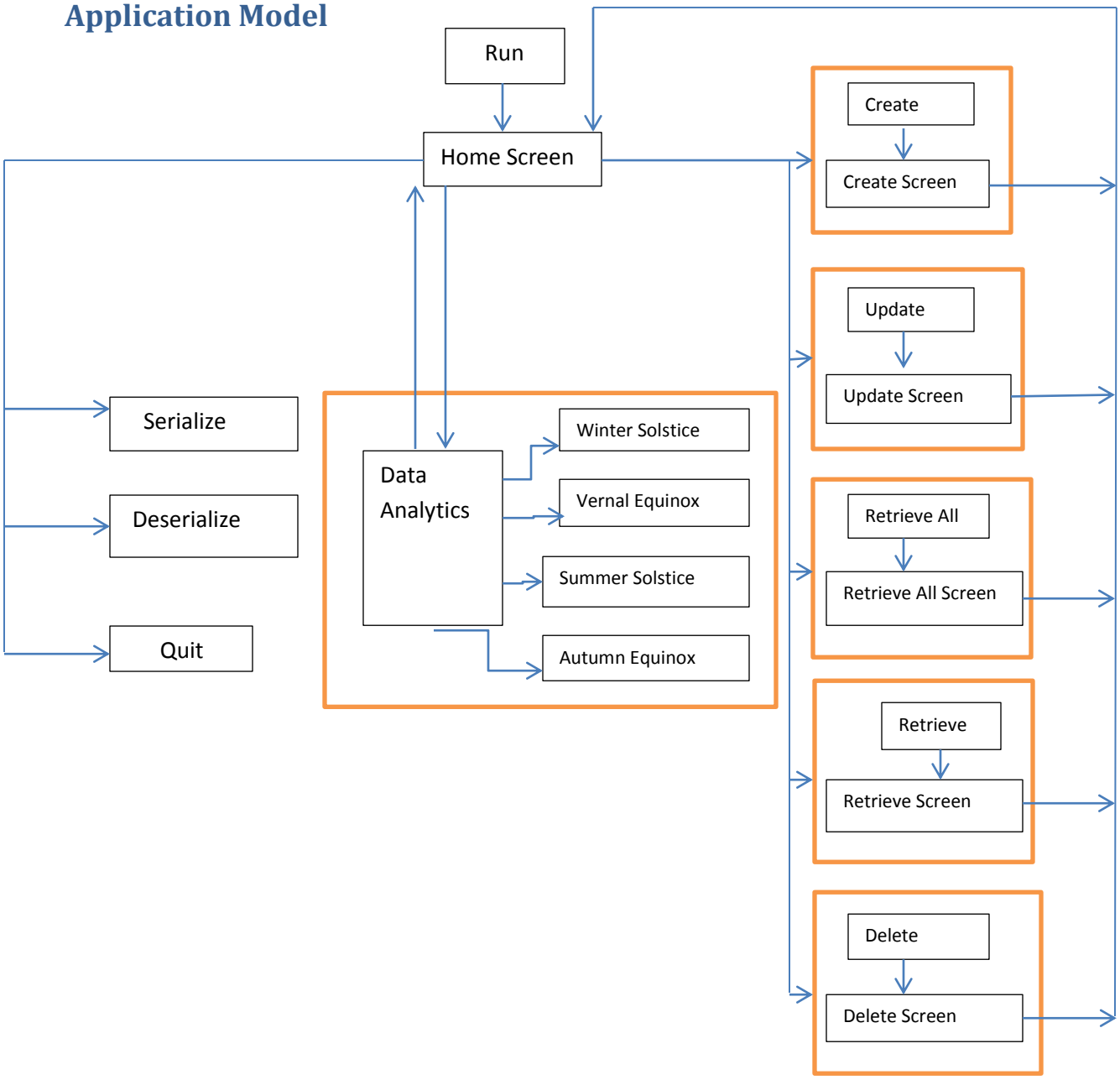
DaylightRecord
sunrise
sunset
daylength
nightlength

It is a simple database table that holds all the values read from the CSV file

- Sunrise - This is a timestamp that holds the date and time of the sunrise in ‘yyyy-mm-dd hh:mm:ss’ format
- Sunset - This is a timestamp that holds the date and time of the sunset in ‘yyyy-mm-dd hh:mm:ss’ format
- Daylength – This is an ‘int’ field that holds the length of day in minutes (sunset-sunrise)
- Nightlength – This is an ‘int’ field that holds the length of the night in minutes (next day’s sunrise – current day’s sunset)

ITMD 411
Sagarika Muniraj
A20295475

Application Model



The Application model of the project is as depicted above.

The User is first presented with a home screen

Home screen consists of the following buttons:

- ReadCSV
- Create – leads to a new create screen
- Update – leads to a new update screen
- Retrieve – leads to Retrieve screen
- Retrieve All – leads to Retrieve All screen
- Delete – Leads to Delete screen
- Serialize
- Deserilaize
- Data Analytics – Leads to Data Analytics Screen
- Quit

Each of the subscreens has a provision to (‘back’ button) take back the user to the main screen.

Architecture and Code Structure

Packages used:

- Domain - Used for the java classes to store Record and its inherited DayLightRecord
- Driver - has the home screen and each of the subscreens
- Connector - Java code to connect to the Database
- Csvimport - Code to read from the csv file and load into the Database

Domain files

- Record.java – abstract class to hold the base data of sunrise and sunset from csv file
- DaylightRecord.java – class extending Record with daylength and nightlength fields

Connector files

- JdbcUtilities.java – Has all the code to connect to the Database

The getConnection() is present here. Please change this to the appropriate userid and passwords.

I have used the default values (root and admin for userid and password).

Csvimport files

- ImportUtilities.java – This has the following functions
  - buildDayLightRecordsList() – to read from the CSV file
  - displayDayLightRecordContents() – to display the read records for developer's testing
  - createAndLoadDatabaseTables() – to create the Database table and load it with the data read from the CSV file
  - sql_exec() – to connect to the database and execute any query
  - getlistofnames() – to get all the days of the year. The user can choose a date from this to update or retrieve

Driver files

- crud.java:
  This is the home screen from which any functionality can be selected
- DataAnalytics.java:
  Performs data analytic functions to find out Winter Solstice, Vernal Equinox, Summer Solstice and Autumnal Equinox
- Record Creation.java:
  Takes user inputs for sunrise and sunset and inserts them as a new row into the database
- RecordUpdate.java:
  User can select a column to update and give a value to update
- RecordRetrieve.java:
  User selects a particular record to view its details
- RecordRetrieveAll.java:
  User clicks to retrieve all data
- RecordDelete.java:
  User selects a particular record to delete

Data

This folder is placed outside the src and contains:

- DayLightRecord.ser – This is used to serialize and deserialize
- sunrise-sunset.csv - This is the input file from which the data is read

Project Insights with Requirements

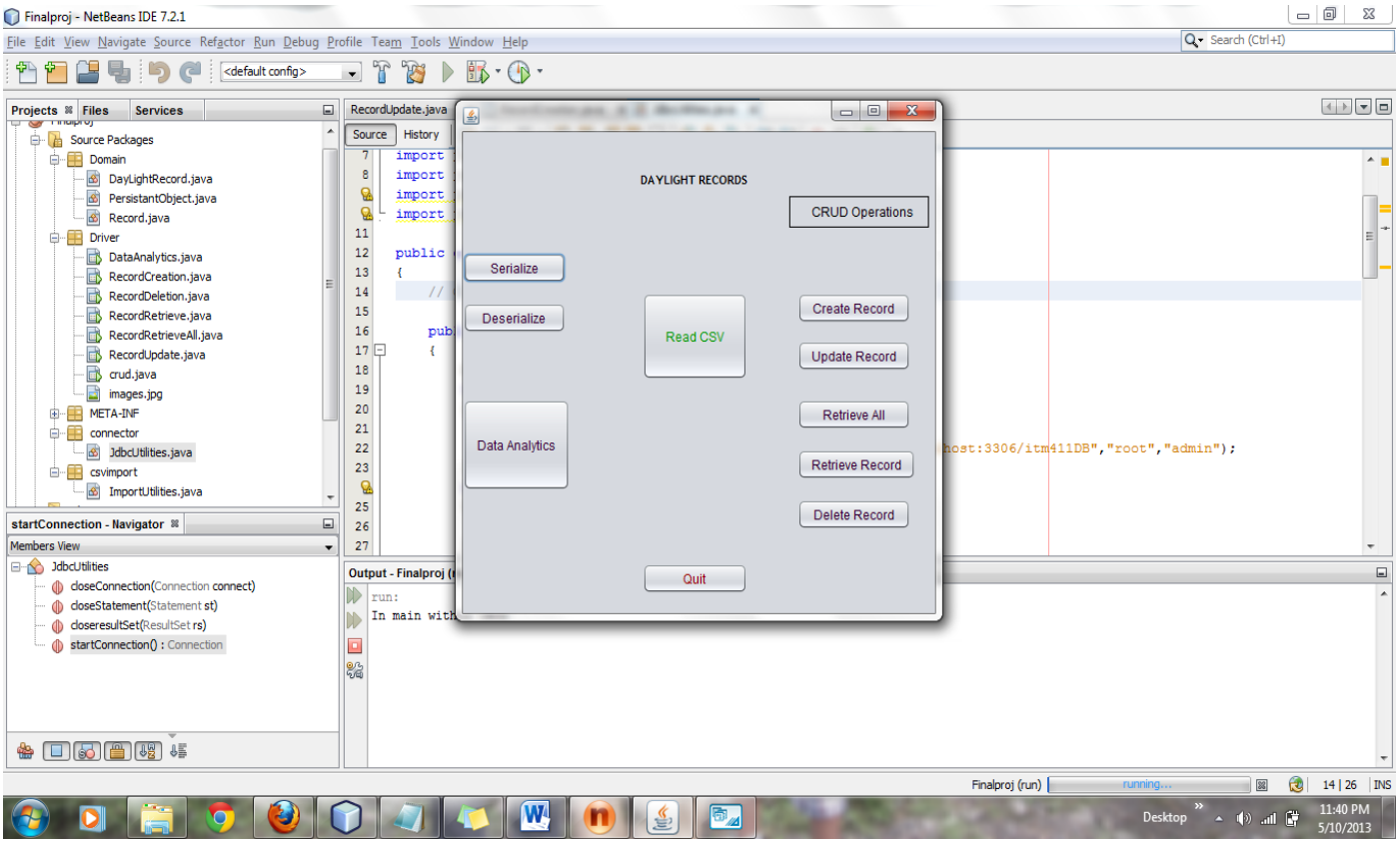
- Ability to read from CSV file and load it into the database table
- The data can be reinitialized at any point of execution of the project which would delete the current data and reset it to the data from CSV
- GUI components are provided for each of the CRUD functionalities
- User can get back to the home screen from any of these CRUD screens
- GUI components are provided for Serializing and Deserializing
- GUI component is provided for Data Analytics that calculates the solstices and equinoxes

Steps for Execution

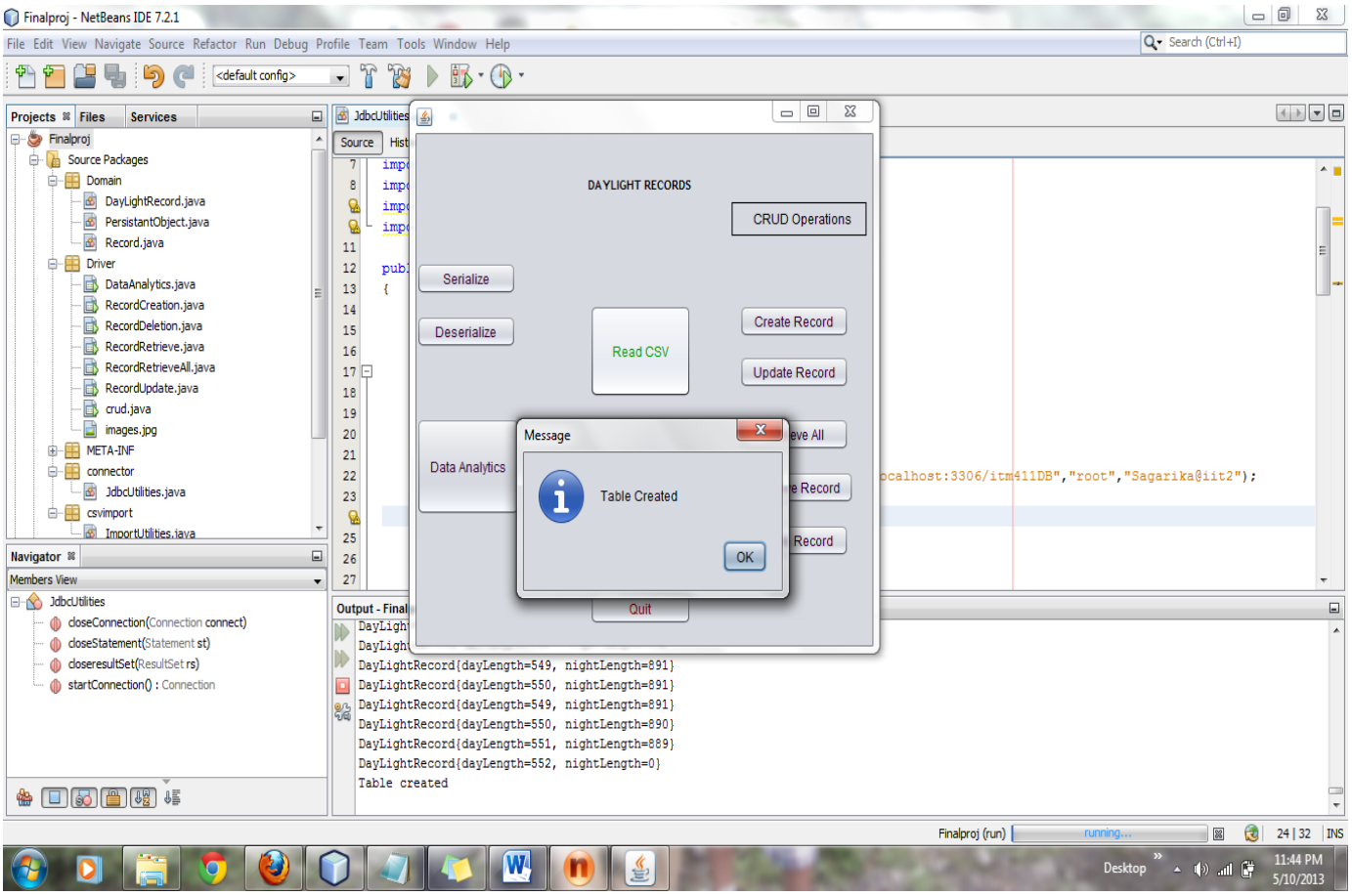
- Open the project in NetBeans IDE
- Build and run
- Check the **Database connection string in connector.JdbcUtilities.java getConnection()** method has the default credentials. Please change these if required
- A HomeScreen is presented from which any of the SubScreens can be chosen
- User can get back to the HomeScreen from any of the SubScreens

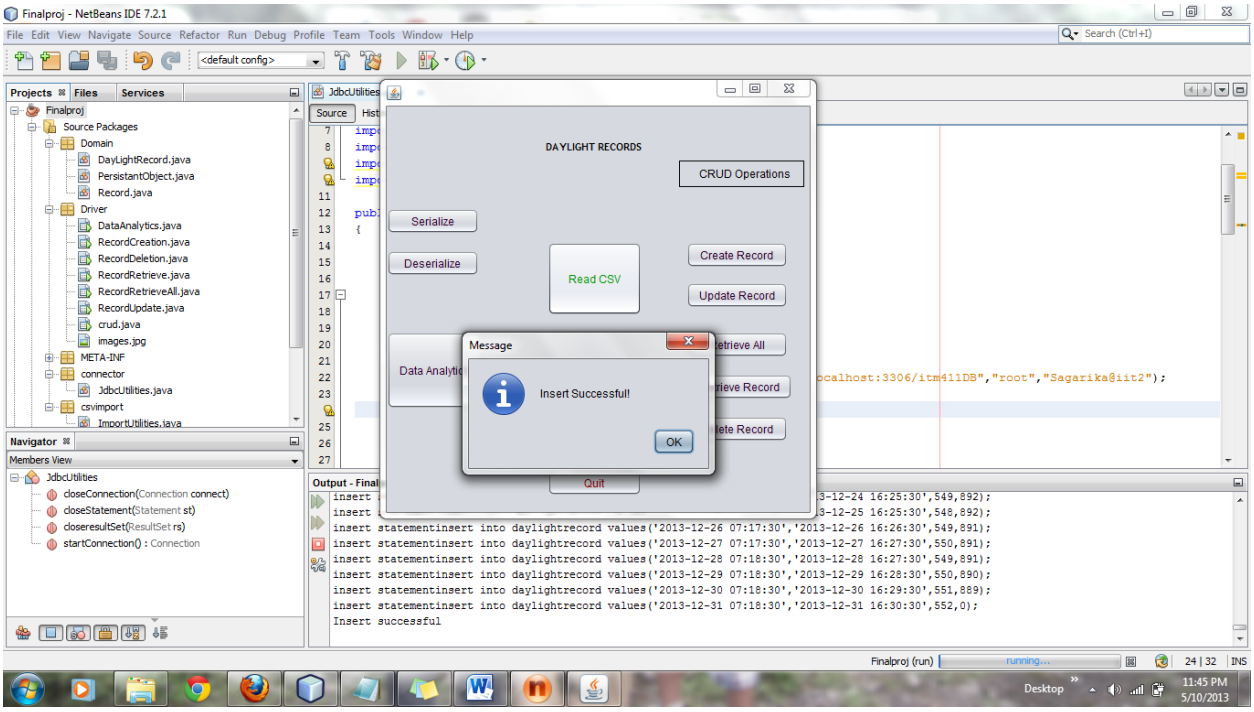
Expected Results and Screen Captures

1. Home Screen

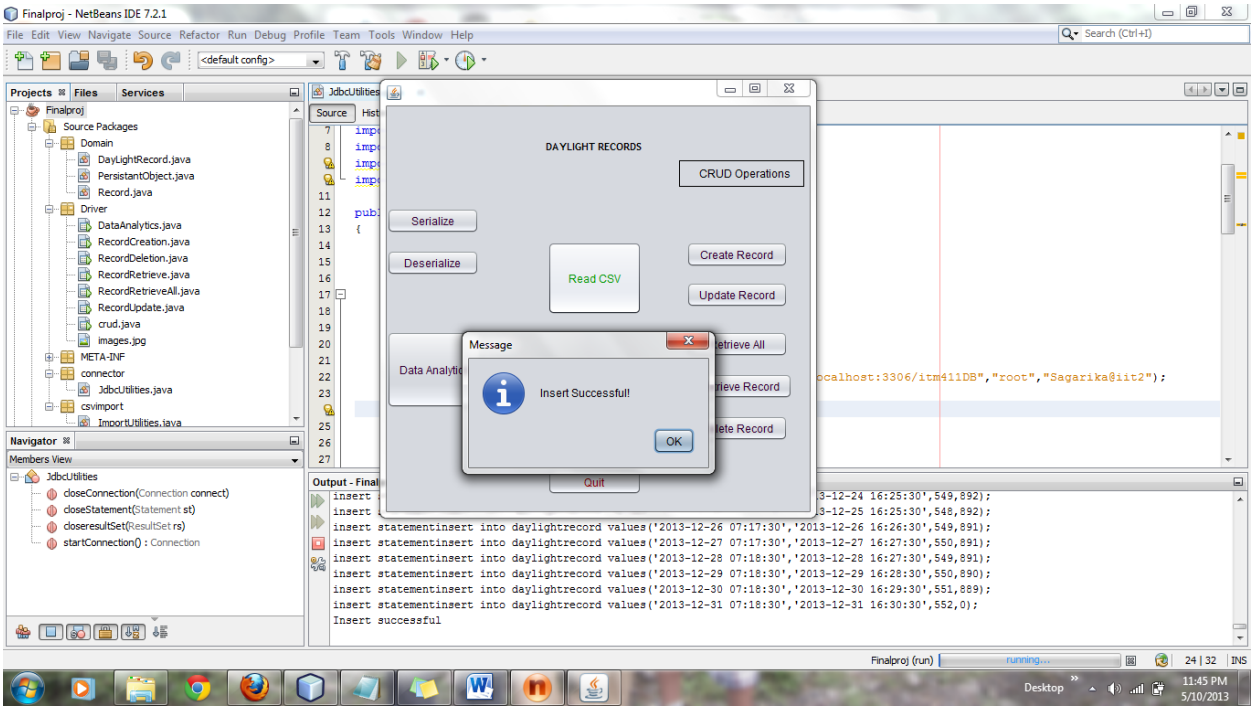

Finalproj - NetBeans IDE 7.2.1
DAYLIGHT RECORDS
CRUD Operations
Serialize
Deserialize
Read CSV
Data Analytics
Create Record
Update Record
Retrieve All
Retrieve Record
Delete Record
Quit

2. Read CSV

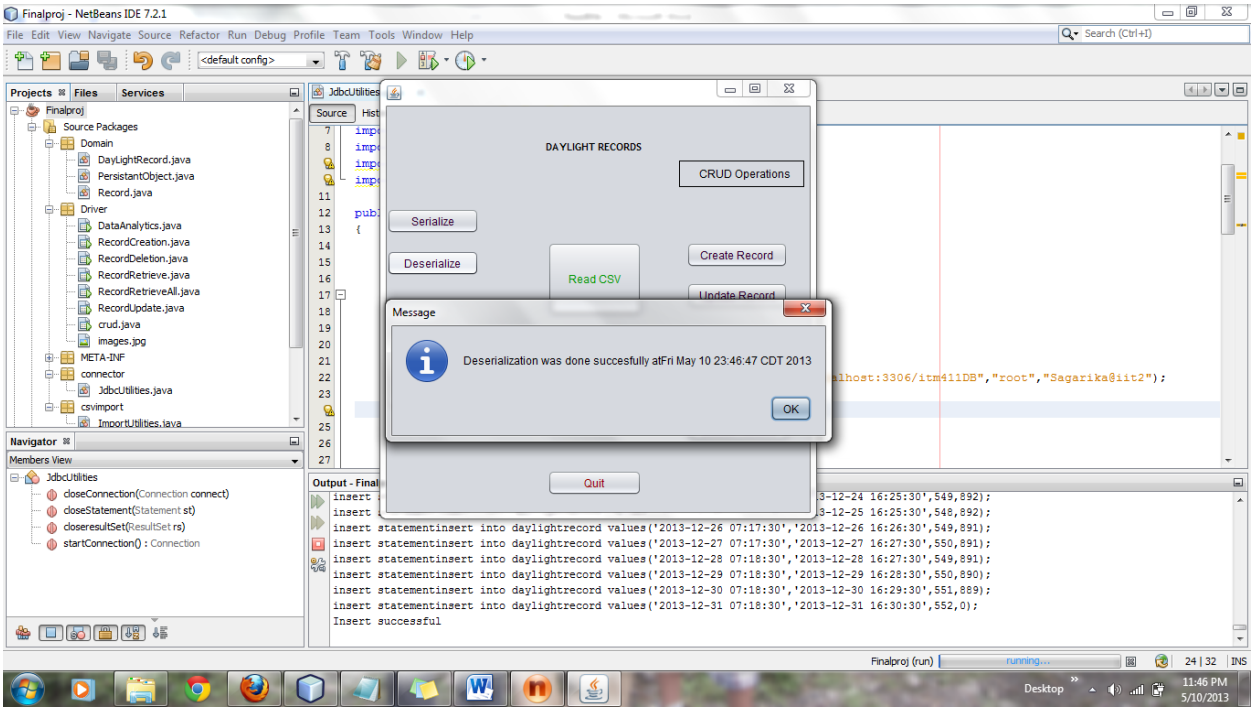

Finalproj - NetBeans IDE 7.2.1
DAYLIGHT RECORDS
CRUD Operations
Serialize
Deserialize
Read CSV
Create Record
Update Record
Data Analytics
Message
Table Created
OK
Quit
DayLightRecord{dayLength=549, nightLength=891}
DayLightRecord{dayLength=550, nightLength=891}
DayLightRecord{dayLength=549, nightLength=891}
DayLightRecord{dayLength=550, nightLength=890}
DayLightRecord{dayLength=551, nightLength=889}
DayLightRecord{dayLength=552, nightLength=0}
Table created

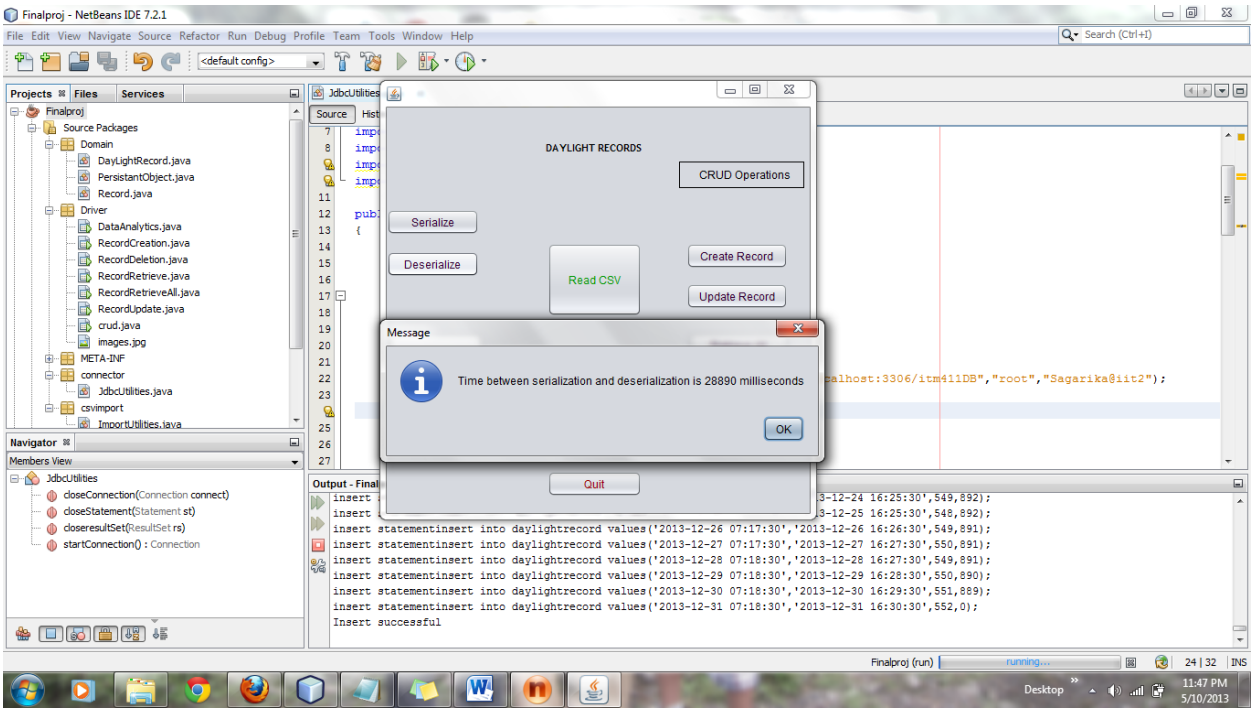

Finalproj - NetBeans IDE 7.2.1
DAYLIGHT RECORDS
CRUD Operations
Serialize
Deserialize
Read CSV
Create Record
Update Record
Message
Insert Successful!
OK
Quit
Insert successful

3. Serialize

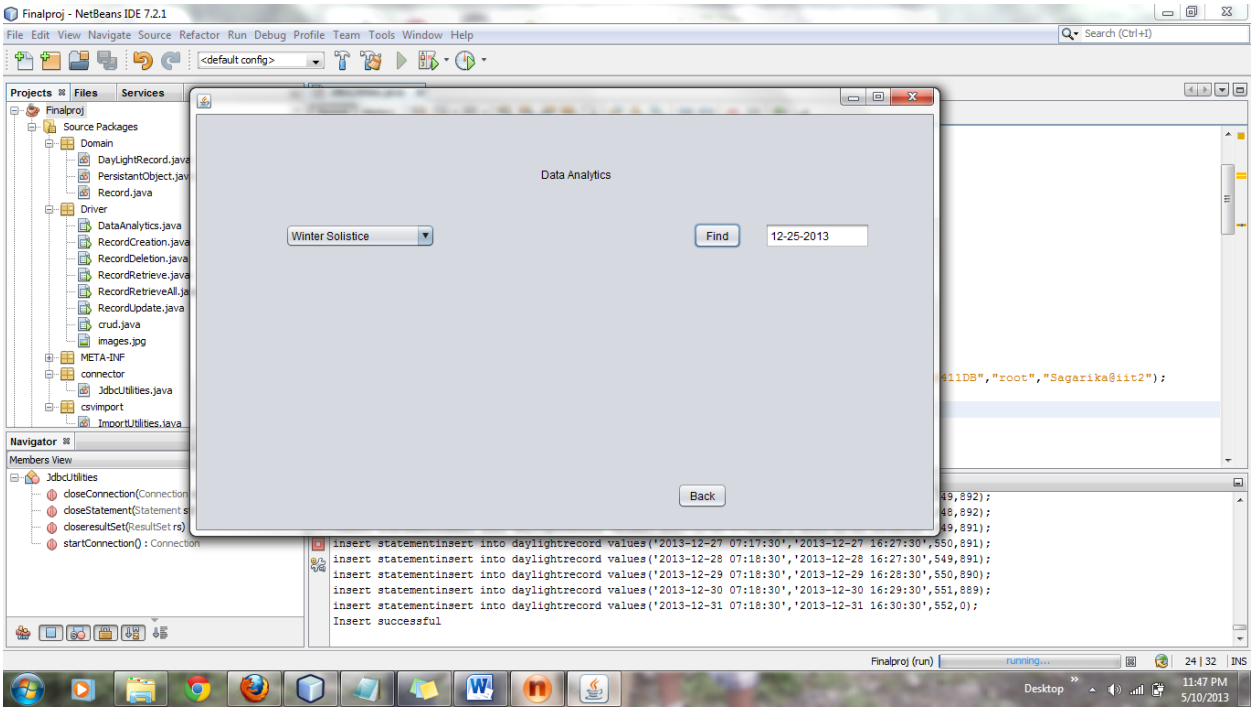


4. Deserialize

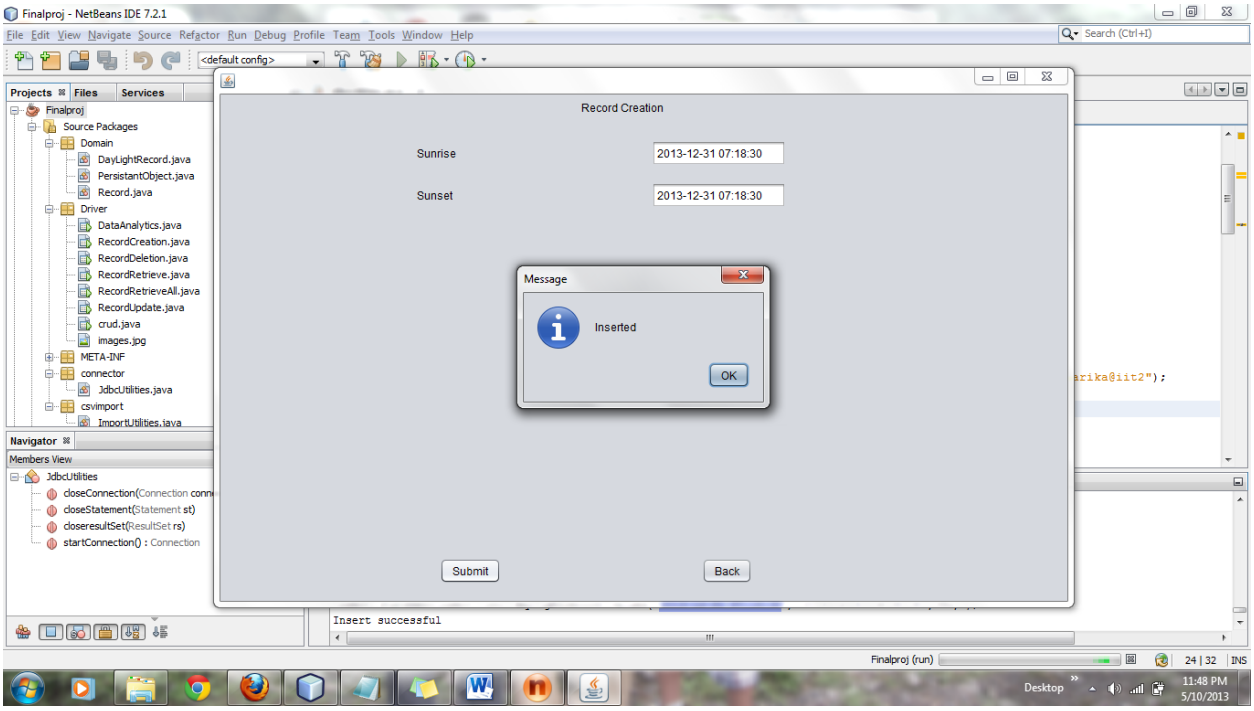




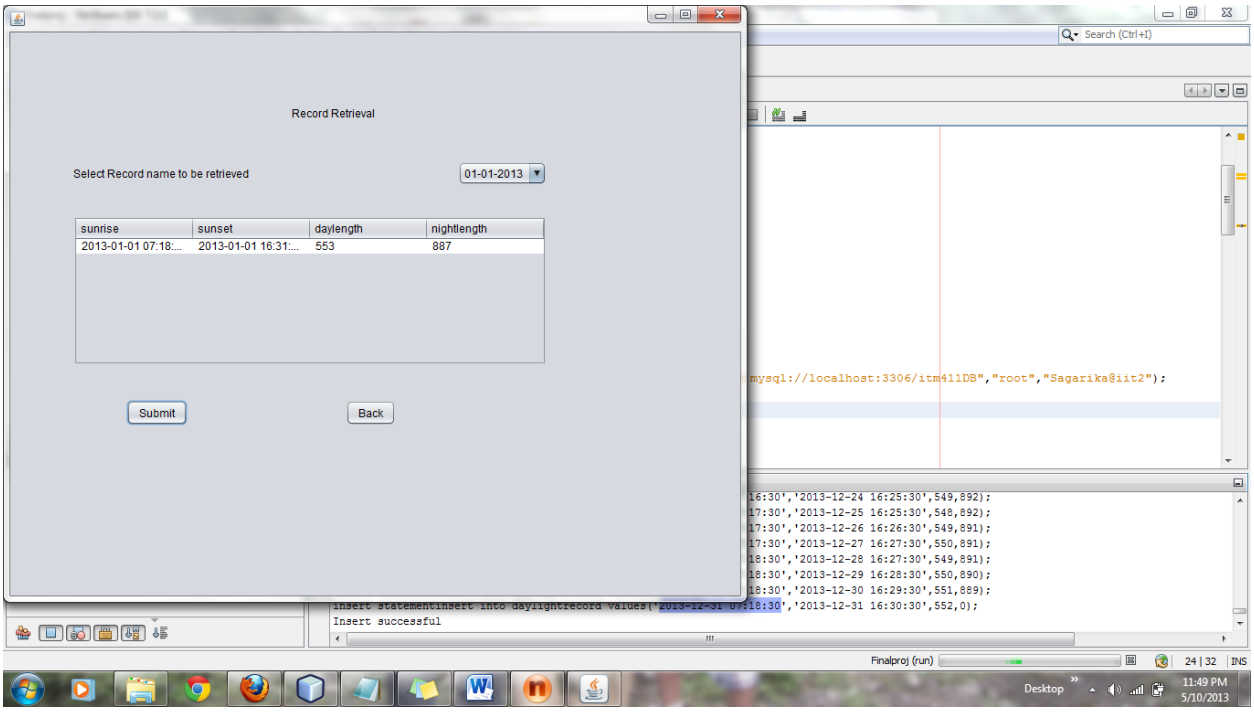
5. Data Analytics



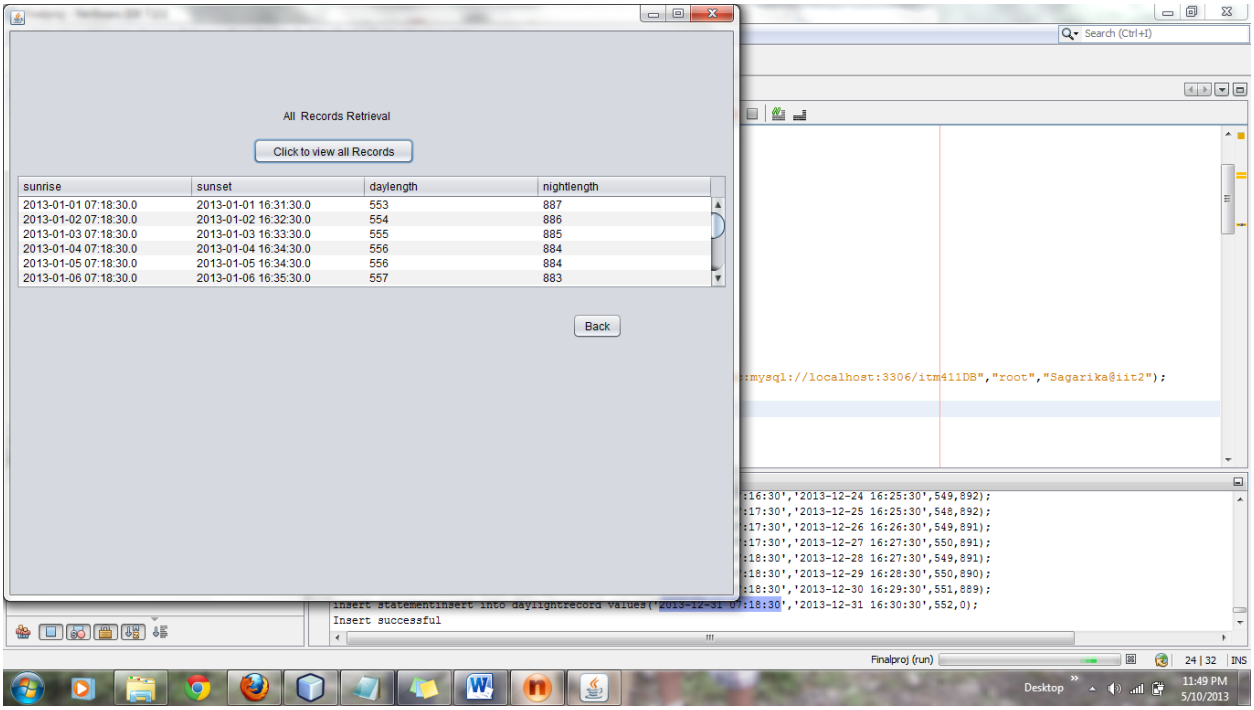
6. Create Record



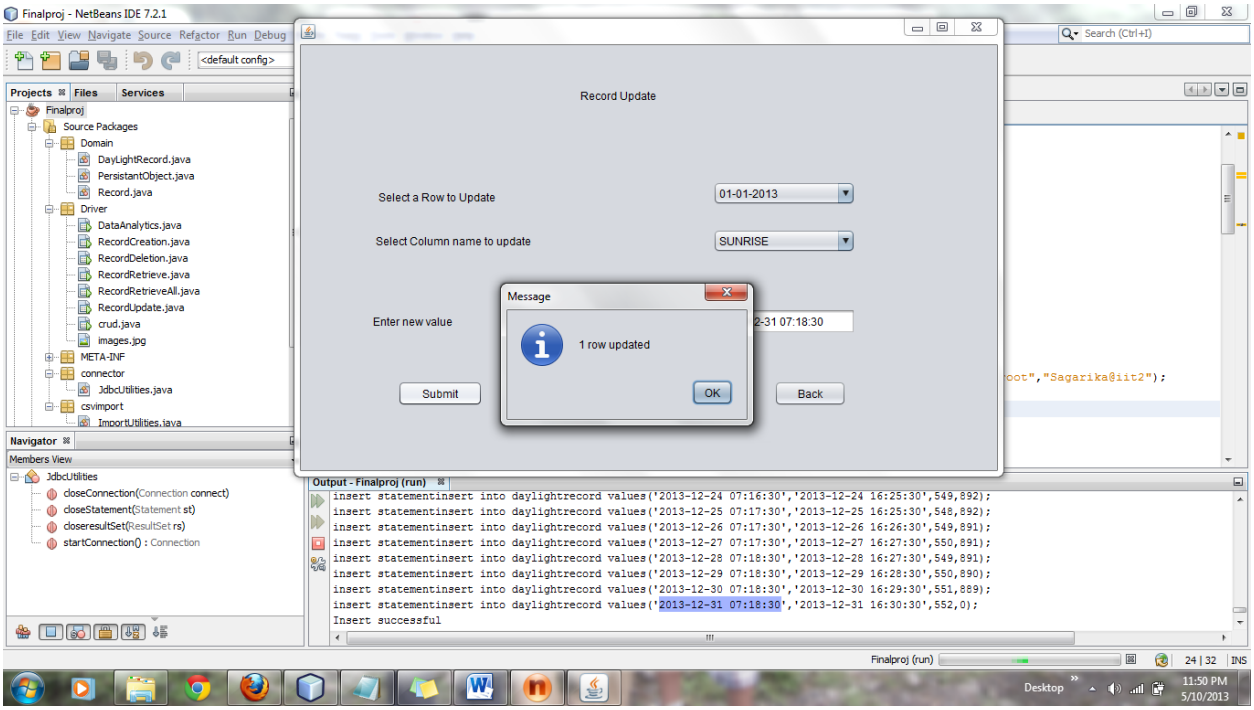
7. Retrieve



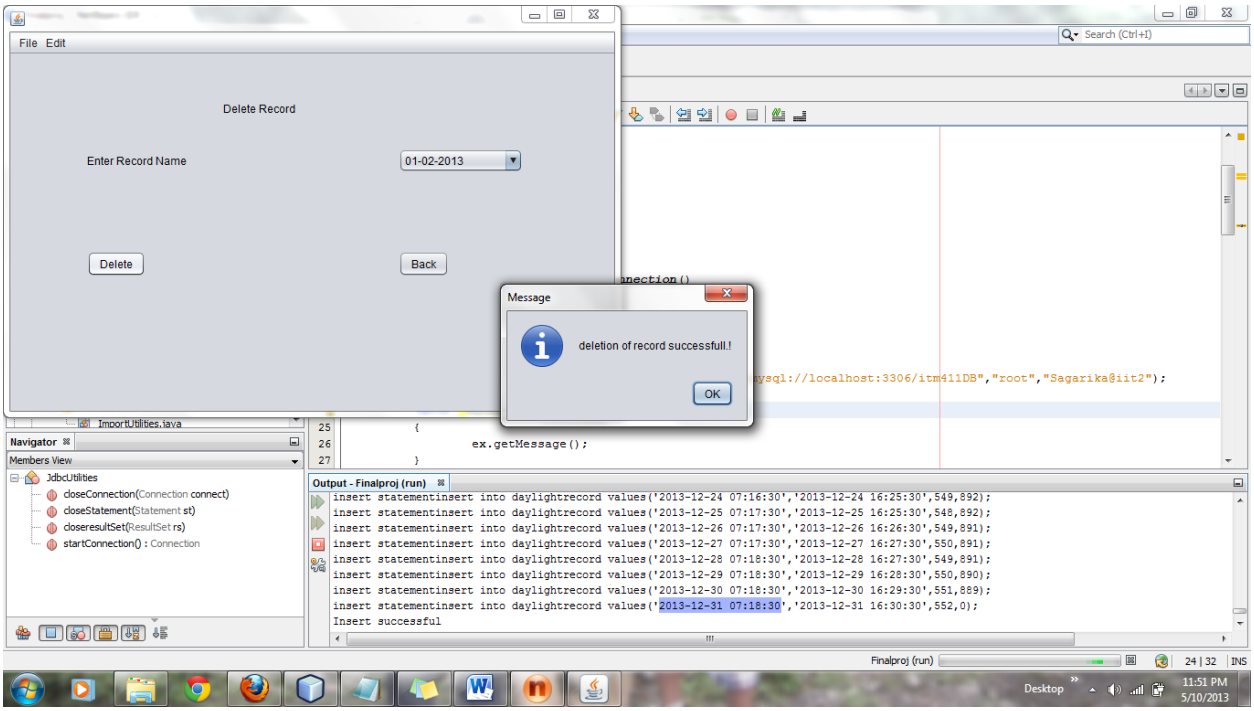
8. Retrieve All



9. Update



10. Delete



11. Exit