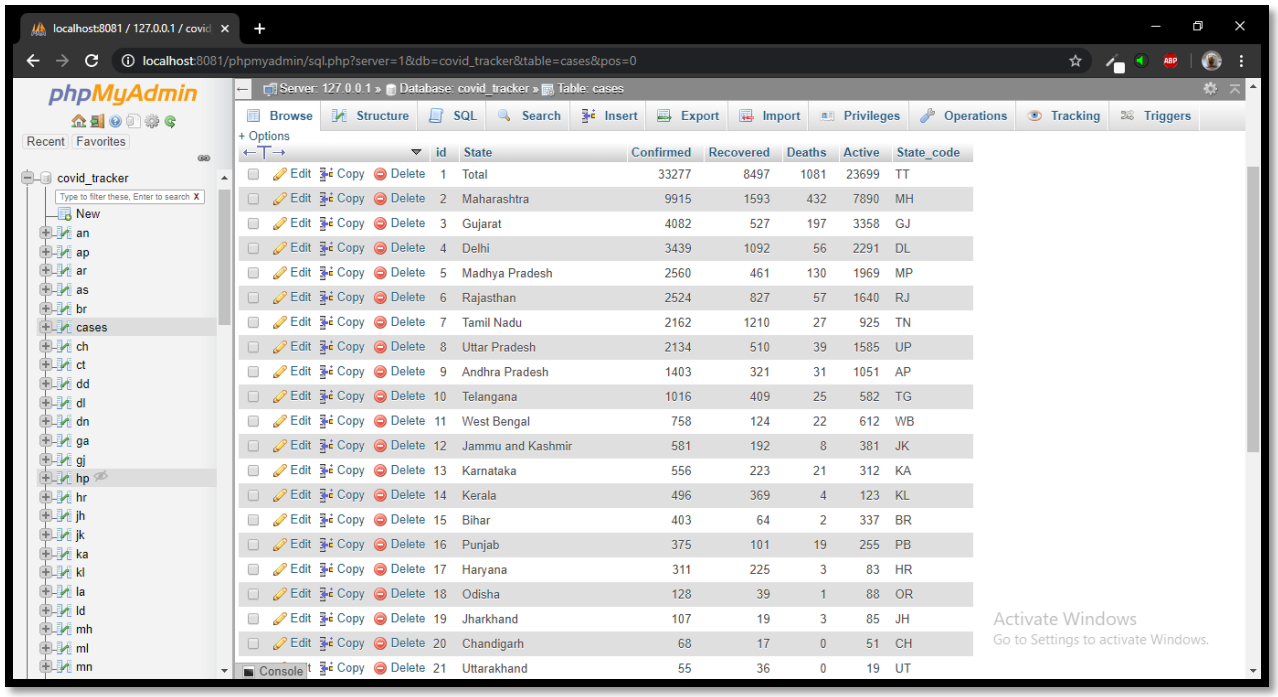
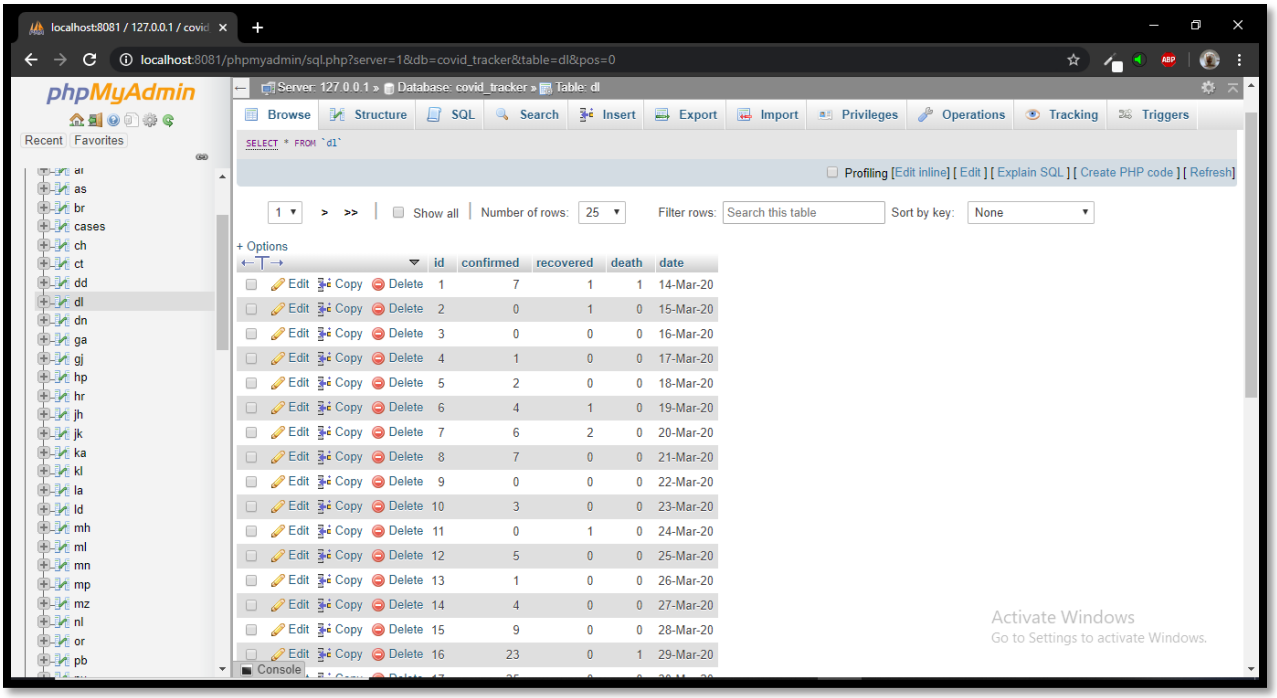
JAVA PROGRAMMING (CSE-1007)
DIGITAL ASSIGNMENT-3

Name: - Harsh Gupta

Registration No. : - 17BCE1152

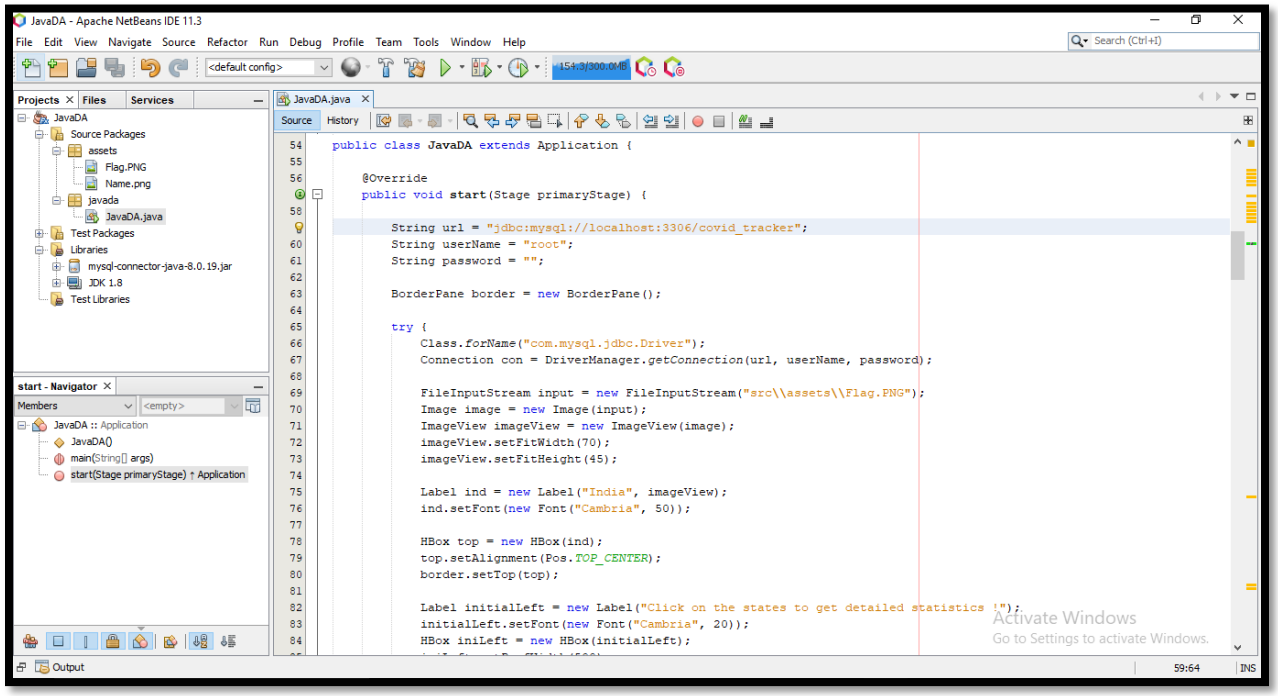
Database Structure





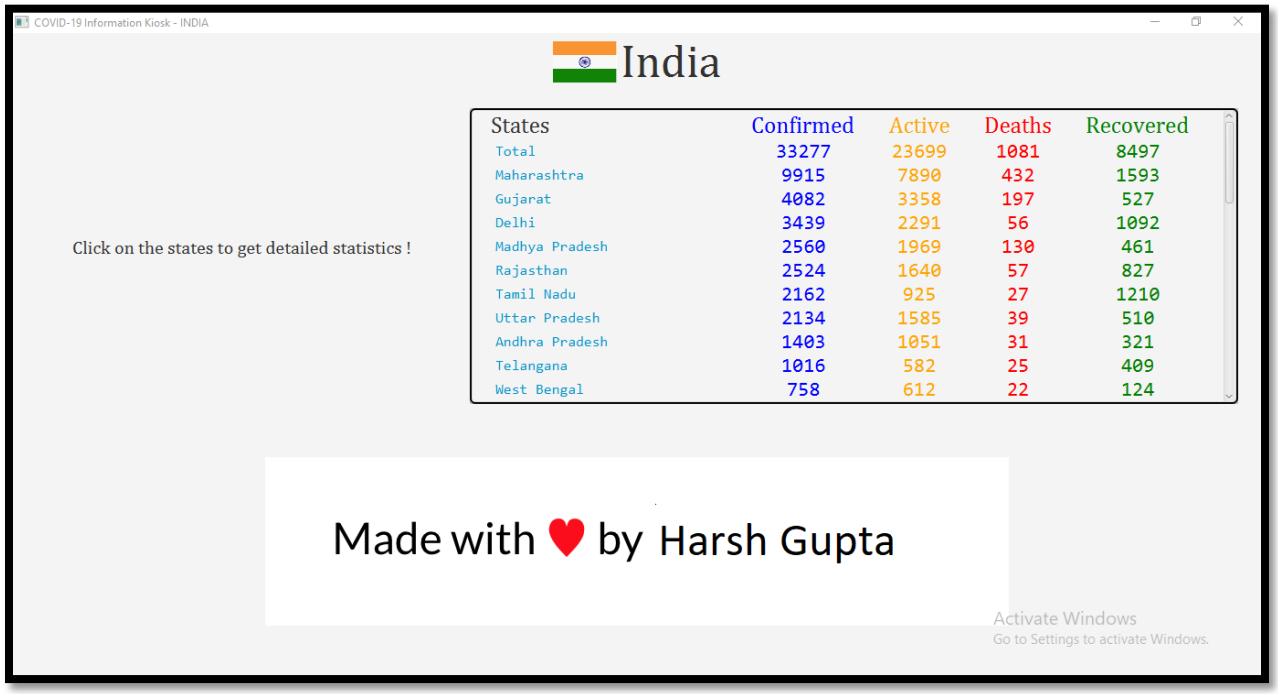
- These are the total stats from 14th March 2020 till 29th April 2020.
- Table `cases` contain the total stats of all states/UT and India as a whole.
- The state code given with each state/UT is used for their tables containing daily stats of the given time period.

Project Structure

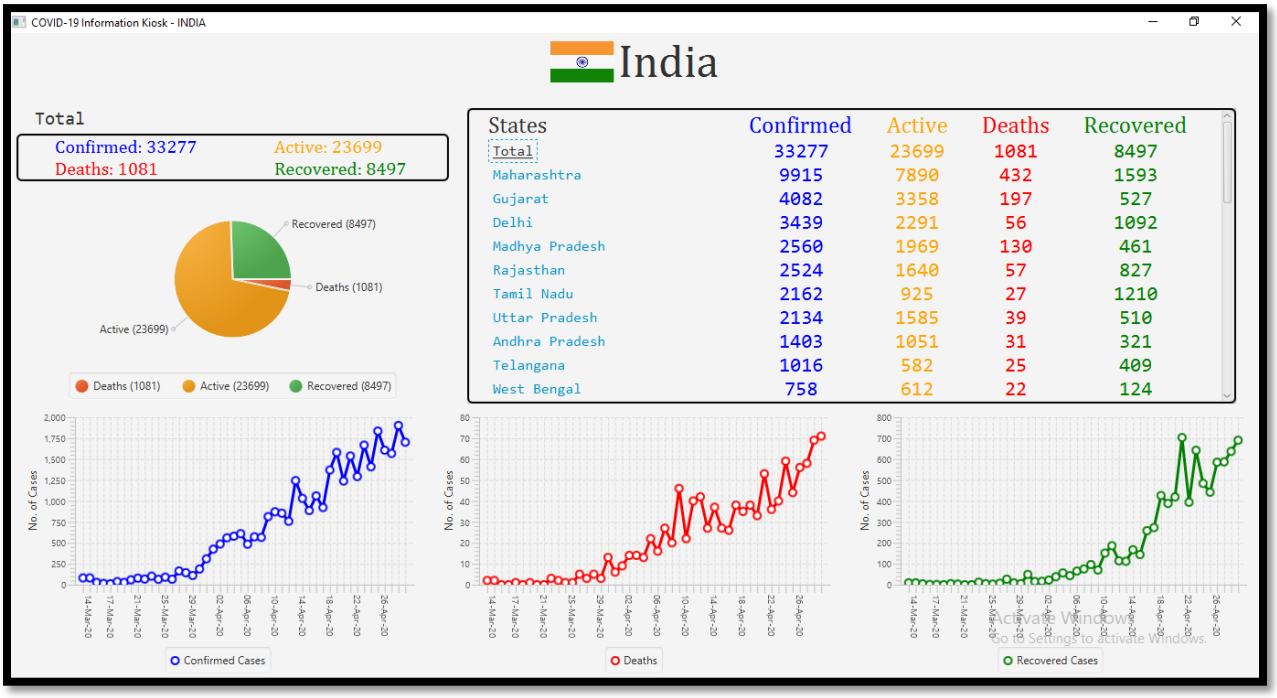


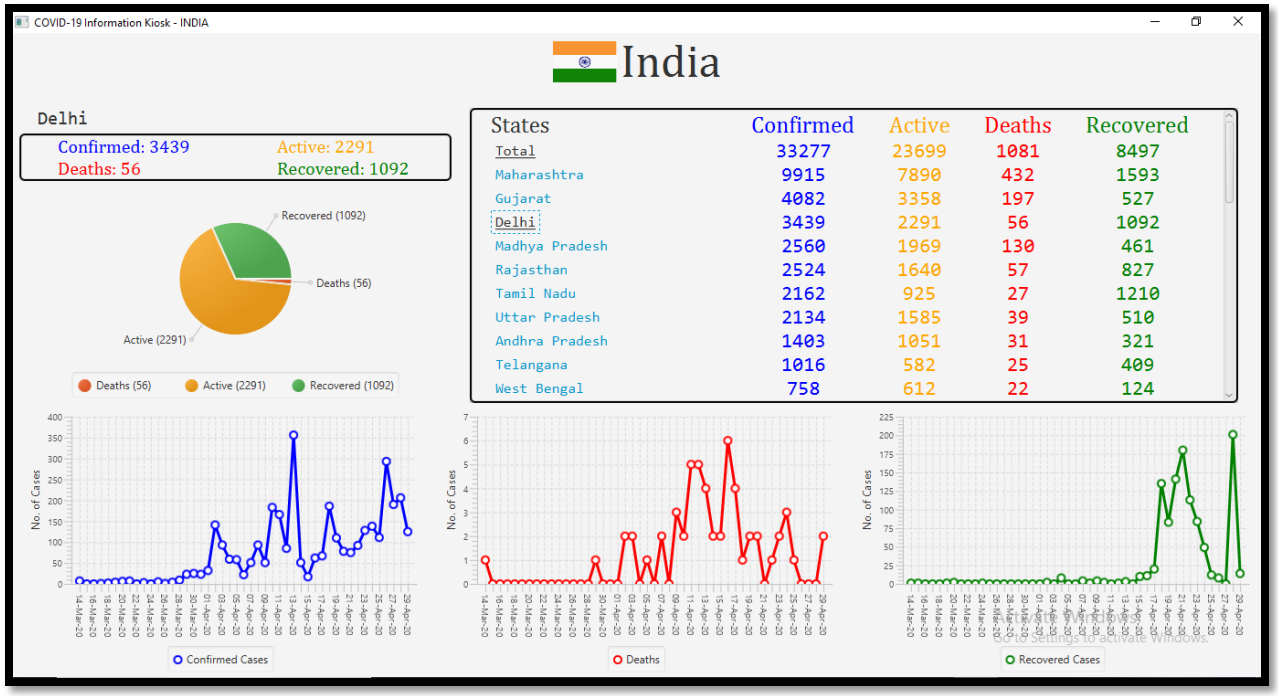
- MySql connecter is added as a library to connect to the MySql database hosted on localhost.

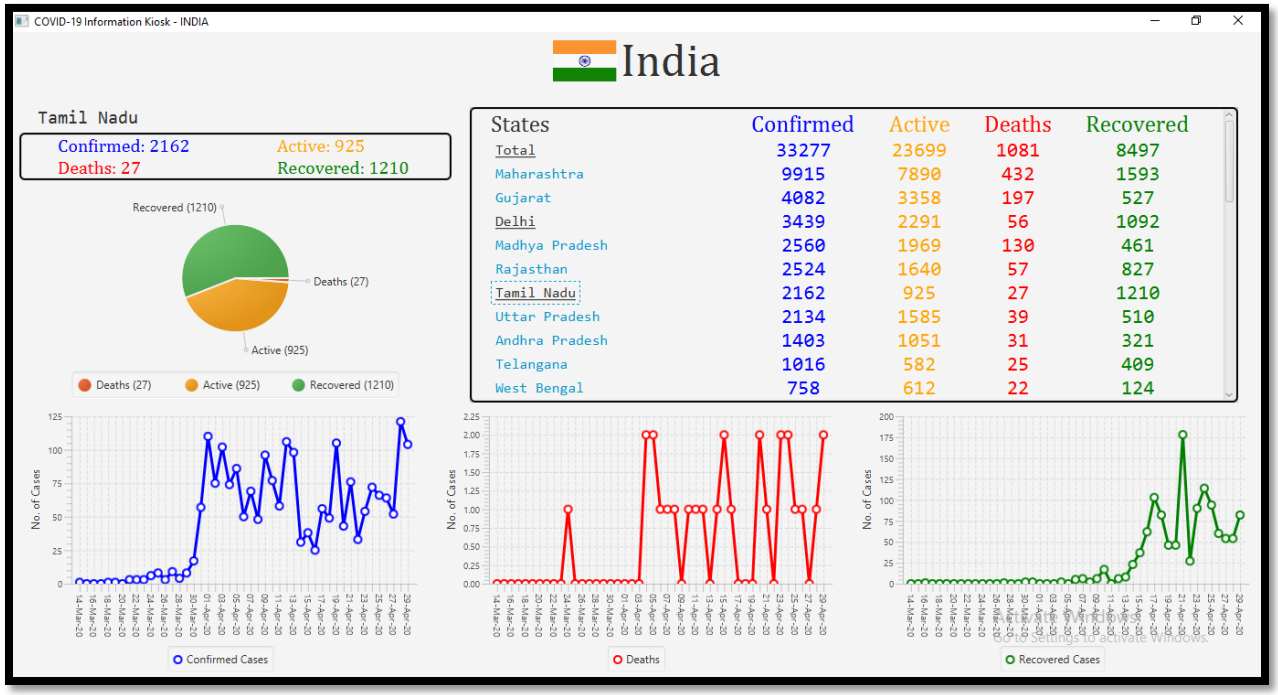
Covid-19 Information Kiosk Screenshots

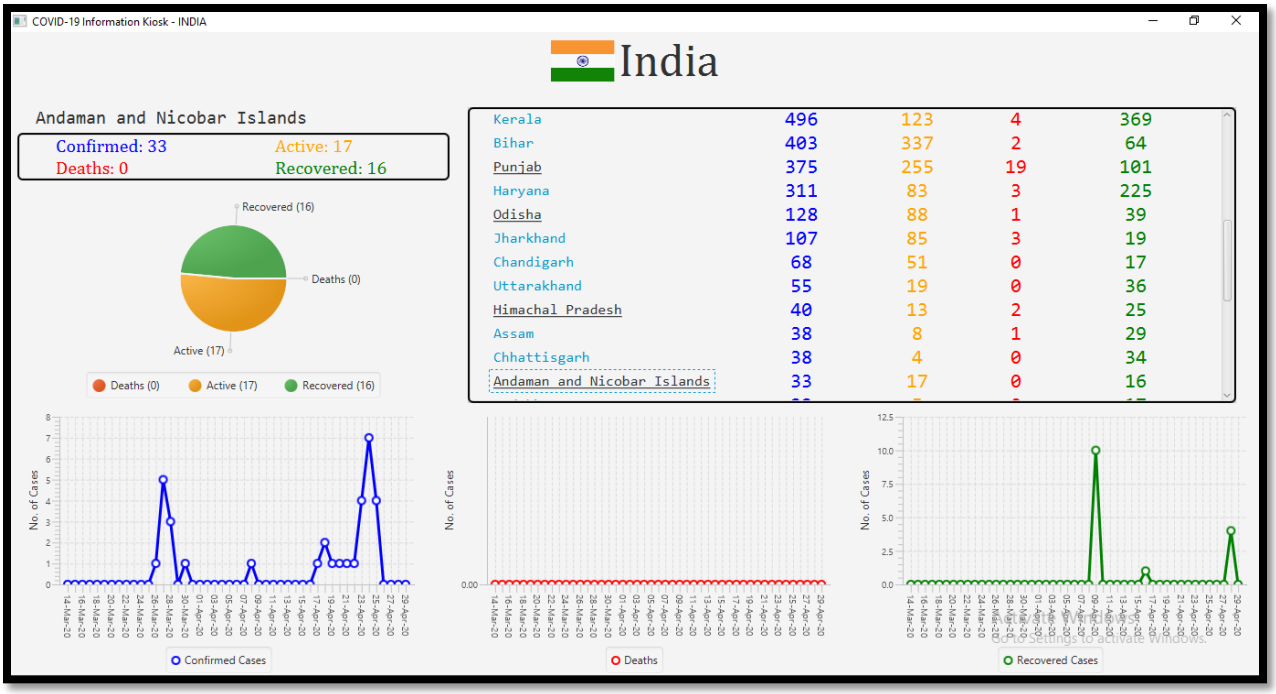


- This is the starting look of the kiosk. The table on the right shows the statistics of Covid-19 cases as of 29th April 2020.
- After clicking on the states of your choice you can see the detailed statistics of those particular states along with vibrant graphs/charts for better visualization.




COVID-19 Information Kiosk - INDIA
India
Delhi
Confirmed: 3439
Active: 2291
Deaths: 56
Recovered: 1092
Recovered (1092)
Deaths (56)
Active (2291)
Deaths (56)
Active (2291)
Recovered (1092)
States
Confirmed
Active
Deaths
Recovered
Total 33277 23699 1081 8497
Maharashtra 9915 7890 432 1593
Gujarat 4082 3358 197 527
Delhi 3439 2291 56 1092
Madhya Pradesh 2560 1969 130 461
Rajasthan 2524 1640 57 827
Tamil Nadu 2162 925 27 1210
Uttar Pradesh 2134 1585 39 510
Andhra Pradesh 1403 1051 31 321
Telangana 1016 582 25 409
West Bengal 758 612 22 124
No. of Cases
Confirmed Cases
Deaths
Recovered Cases


COVID-19 Information Kiosk - INDIA
India
Tamil Nadu
Confirmed: 2162
Active: 925
Deaths: 27
Recovered: 1210
Recovered (1210)
Deaths (27)
Active (925)
Deaths (27)
Active (925)
Recovered (1210)
States
Confirmed
Active
Deaths
Recovered
Total 33277 23699 1081 8497
Maharashtra 9915 7890 432 1593
Gujarat 4082 3358 197 527
Delhi 3439 2291 56 1092
Madhya Pradesh 2560 1969 130 461
Rajasthan 2524 1640 57 827
Tamil Nadu 2162 925 27 1210
Uttar Pradesh 2134 1585 39 510
Andhra Pradesh 1403 1051 31 321
Telangana 1016 582 25 409
West Bengal 758 612 22 124
No. of Cases
Confirmed Cases
Deaths
Recovered Cases



Github Repository Link

https://github.com/Harshg999/Covid-19-Information-Kiosk-India

Project Code

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package javada;

import java.io.FileInputStream;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.logging.Level;
import java.util.logging.Logger;
```

```
import javafx.application.Application;
import javafx.event.ActionEvent;
import javafx.event.EventHandler;
import javafx.geometry.Insets;
import javafx.geometry.Orientation;
import javafx.geometry.Pos;
import javafx.scene.Node;
import javafx.scene.Scene;
import javafx.scene.chart.CategoryAxis;
import javafx.scene.chart.LineChart;
import javafx.scene.chart.NumberAxis;
import javafx.scene.chart.PieChart;
import javafx.scene.chart.XYChart;
import javafx.scene.control.Button;
import javafx.scene.control.Hyperlink;
import javafx.scene.control.Label;
import javafx.scene.control.ScrollPane;
import javafx.scene.control.Separator;
import javafx.scene.control.TableColumn;
import javafx.scene.control.TableView;
import javafx.scene.control.cell.PropertyValueFactory;
import javafx.scene.image.Image;
import javafx.scene.image.ImageView;
import javafx.scene.layout.AnchorPane;
import javafx.scene.layout.Background;
import javafx.scene.layout.BorderPane;
import javafx.scene.layout.FlowPane;
import javafx.scene.layout.HBox;
import javafx.scene.layout.StackPane;
import javafx.scene.layout.VBox;
import javafx.scene.paint.Color;
import javafx.scene.text.Font;
import javafx.stage.Stage;
```

```
/**
 *
 * @author Harsh Gupta (17BCE1152)
 */
public class JavaDA extends Application {

   @Override
   public void start(Stage primaryStage) {

      String url = "jdbc:mysql://localhost:3306/covid_tracker";
      String userName = "root";
      String password = "";

      BorderPane border = new BorderPane();

      try {
         Class.forName("com.mysql.jdbc.Driver");
         Connection con = DriverManager.getConnection(url, userName,
password);

         FileInputStream input = new
FileInputStream("src\\assets\\Flag.PNG");
         Image image = new Image(input);
         ImageView imageView = new ImageView(image);
         imageView.setFitWidth(70);
         imageView.setFitHeight(45);

         Label ind = new Label("India", imageView);
         ind.setFont(new Font("Cambria", 50));

         HBox top = new HBox(ind);
         top.setAlignment(Pos.TOP_CENTER);
         border.setTop(top);
```

```
        Label initialLeft = new Label("Click on the states to get detailed
statistics !");
        initialLeft.setFont(new Font("Cambria", 20));
        HBox iniLeft = new HBox(initialLeft);
        iniLeft.setPrefWidth(500);
        iniLeft.setAlignment(Pos.CENTER);
        border.setLeft(iniLeft);

        FileInputStream harsh = new
FileInputStream("src\\assets\\Name.PNG");
        Image imgName = new Image(harsh);
        ImageView imageName = new ImageView(imgName);

        HBox bottomCredit = new HBox(imageName);
        bottomCredit.setPrefHeight(300);
        bottomCredit.setAlignment(Pos.CENTER);
        border.setBottom(bottomCredit);

        Label states = new Label("States");
        states.setFont(new Font("Cambria", 25));

        Label confirmed = new Label("Confirmed");
        confirmed.setFont(new Font("Cambria", 25));
        confirmed.setTextFill(Color.BLUE);

        Label active = new Label("Active");
        active.setFont(new Font("Cambria", 25));
        active.setTextFill(Color.ORANGE);

        Label deaths = new Label("Deaths");
        deaths.setFont(new Font("Cambria", 25));
        deaths.setTextFill(Color.RED);

        Label recovered = new Label("Recovered");
```

```
recovered.setFont(new Font("Cambria", 25));
recovered.setTextFill(Color.GREEN);

//HBox h1 = new HBox(states, confirmed, active, deaths, recovered);
VBox v1 = new VBox(states);
VBox v2 = new VBox(confirmed);
VBox v3 = new VBox(recovered);
VBox v4 = new VBox(deaths);
VBox v5 = new VBox(active);

Statement stmt = con.createStatement();
ResultSet rs = stmt.executeQuery("select * from cases");
while (rs.next()) {
   String col2 = rs.getString(2);
   String col3 = rs.getString(3).toLowerCase();
   String col4 = rs.getString(4).toLowerCase();
   String col5 = rs.getString(5).toLowerCase();
   String col6 = rs.getString(6).toLowerCase();
   String col7 = rs.getString(7).toLowerCase();

   Hyperlink hy1 = new Hyperlink(rs.getString(2));
   hy1.setOnAction(e -> {

      Label c1 = new Label("Confirmed: " + col3);
      c1.setFont(new Font("Cambria", 20));
      c1.setTextFill(Color.BLUE);

      Label a1 = new Label("Active: " + col6);
      a1.setFont(new Font("Cambria", 20));
      a1.setTextFill(Color.ORANGE);

      Label d1 = new Label("Deaths: " + col5);
      d1.setFont(new Font("Cambria", 20));
      d1.setTextFill(Color.RED);
```

```
        Label r1 = new Label("Recovered: " + col4);
        r1.setFont(new Font("Cambria", 20));
        r1.setTextFill(Color.GREEN);

        VBox display1 = new VBox(c1, d1);
        display1.setPrefWidth(240);
        VBox display2 = new VBox(a1, r1);

        HBox displayBox = new HBox(display1, display2);

        String displaybox_border = "-fx-border-color: black;" + "-fx-border-
width: 2;" + "-fx-border-insets: 5 20 0 20;" + "-fx-border-radius: 5;";
        displayBox.setStyle(displaybox_border);

        displayBox.setPadding(new Insets(0, 30, 0, 40));

        PieChart pieChart = new PieChart();

        PieChart.Data slice2 = new PieChart.Data("Active (" + col6 + ")",
Integer.parseInt(col6));
        PieChart.Data slice1 = new PieChart.Data("Deaths (" + col5 + ")",
Integer.parseInt(col5));
        PieChart.Data slice3 = new PieChart.Data("Recovered (" + col4 + ")",
Integer.parseInt(col4));

        pieChart.getData().add(slice1);
        pieChart.getData().add(slice2);
        pieChart.getData().add(slice3);

        Label name = new Label(col2);

        name.setFont(new Font("Consolas", 20));
        name.setPadding(new Insets(0, 0, 0, 40));
```

```
        VBox left = new VBox(name, displayBox, pieChart);
        left.setPadding(new Insets(20, 0, 0, -13));
        left.setPrefWidth(500);
        border.setLeft(left);

        CategoryAxis xAxis1 = new CategoryAxis();
        CategoryAxis xAxis2 = new CategoryAxis();
        CategoryAxis xAxis3 = new CategoryAxis();

        NumberAxis yAxis1 = new NumberAxis();
        NumberAxis yAxis2 = new NumberAxis();
        NumberAxis yAxis3 = new NumberAxis();
        yAxis1.setLabel("No. of Cases");
        yAxis2.setLabel("No. of Cases");
        yAxis3.setLabel("No. of Cases");

        LineChart<String, Number> lineChart1 = new LineChart<>(xAxis1,
yAxis1);
        LineChart<String, Number> lineChart2 = new LineChart<>(xAxis2,
yAxis2);
        LineChart<String, Number> lineChart3 = new LineChart<>(xAxis3,
yAxis3);

        XYChart.Series dataSeries1 = new XYChart.Series();
        XYChart.Series dataSeries2 = new XYChart.Series();
        XYChart.Series dataSeries3 = new XYChart.Series();

        dataSeries1.setName("Confirmed Cases");
        dataSeries2.setName("Deaths");
        dataSeries3.setName("Recovered Cases");

        try {
```

```
        String query = "select * from `" + col7 + "`";
        ResultSet rs2 = stmt.executeQuery(query);

        while (rs2.next()) {
          int col22 = rs2.getInt(2);
          int col33 = rs2.getInt(3);
          int col44 = rs2.getInt(4);
          String col55 = rs2.getString(5);

          dataSeries1.getData().add(new XYChart.Data(col55, col22));

          dataSeries2.getData().add(new XYChart.Data(col55, col44));
          dataSeries3.getData().add(new XYChart.Data(col55, col33));
        }
      } catch (Exception ex) {
        ex.printStackTrace();
      }
      lineChart1.getData().add(dataSeries1);
      lineChart1.setStyle("CHART_COLOR_1: blue ;");
      lineChart2.getData().add(dataSeries2);
      lineChart2.setStyle("CHART_COLOR_1: red ;");
      lineChart3.getData().add(dataSeries3);
      lineChart3.setStyle("CHART_COLOR_1: green ;");

      HBox bottom = new HBox(lineChart1, lineChart2, lineChart3);
      bottom.setPrefHeight(300);
      border.setBottom(bottom);
    });

    v1.getChildren().add(hy1);
    v1.setAlignment(Pos.CENTER_LEFT);
    //hy1.setAlignment(Pos.CENTER);

    hy1.setFont(new Font("Consolas", 16));
```

```
        Label c3 = new Label(col3);
        c3.setFont(new Font("Consolas", 22));
        c3.setTextFill(Color.BLUE);

        Label c4 = new Label(col4);
        c4.setFont(new Font("Consolas", 22));
        c4.setTextFill(Color.GREEN);

        Label c5 = new Label(col5);
        c5.setFont(new Font("Consolas", 22));
        c5.setTextFill(Color.RED);

        Label c6 = new Label(col6);
        c6.setFont(new Font("Consolas", 22));
        c6.setTextFill(Color.ORANGE);

        v2.getChildren().add(c3);
        v2.setAlignment(Pos.TOP_CENTER);
        v3.getChildren().add(c4);
        v3.setAlignment(Pos.TOP_CENTER);
        v4.getChildren().add(c5);
        v4.setAlignment(Pos.TOP_CENTER);
        v5.getChildren().add(c6);
        v5.setAlignment(Pos.TOP_CENTER);

        HBox h2 = new HBox(v1, v2, v5, v4, v3);
        h2.setPadding(new Insets(0, 0, 0, 20));

        h2.setSpacing(37);

        ScrollPane sp = new ScrollPane(h2);
        String centerList = "-fx-border-color: black;" + "-fx-border-width: 2;"
+ "-fx-border-radius: 5;";
        sp.setStyle(centerList);
```

```
            VBox cen = new VBox(sp);
            cen.setPadding(new Insets(20, 0, 0, 0));

            border.setCenter(cen);

          }
        } catch (Exception e) {
          e.printStackTrace();
        }

        Label test = new Label();
        test.setPrefWidth(25);
        border.setRight(test);
        Scene scene = new Scene(border, 1350, 700);

        primaryStage.setTitle("COVID-19 Information Kiosk - INDIA");
        primaryStage.setScene(scene);
        primaryStage.show();
    }

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        launch(args);
    }

}
```