202103 基隆市 房屋實價登錄

EEIT29 06 鍾喬安

期中成果報告

2021/06/11

目錄

01　資料來源介紹

01 資料來源

01 資料來源

連結：https://data.gov.tw/dataset/139106

01 資料來源

02 SQL

CreateTable.sql - localhost.Building (sa (55))* - Microsoft SQL Server Management Studio

CreateTable.sql - localhost.master (sa (55)) - Microsoft SQL Server Management Studio

```sql
create table building
(
id int identity (1001,1) constraint building_id_PK primary key,
region char(6) not null,
dealdate char(7) not null,
buildtype char(8) not null,
loc char(25) not null,
area numeric(7,2) not null,
price int not null,
parking char(2) not null,
remark varchar(50)
);
```

	id	region	dealdate	buildtype	loc	area	price	parking	remark
1	1001	仁愛區	1091121	公寓	成功一路61~90號	150.83	6200000	否	預售屋
2	1002	信義區	1091028	華廈	崇法街89巷1~30號	96.56	6300000	否	預售屋
3	1003	信義區	1091211	華廈	崇法街89巷1~30號	99.02	6310000	否	預售屋
4	1004	七堵區	1091230	公寓	百三街151~180號	139.58	7380000	否	無
5	1005	七堵區	1100217	住宅大樓	明德二路1~30號	49.42	2500000	否	無
6	1006	七堵區	1100206	華廈	堵南街1~30號	144.56	8500000	否	無
7	1007	七堵區	1100221	公寓	東新街89巷1~30號	97.07	2600000	否	無
8	1008	七堵區	1100124	公寓	明德一路1~30號	128.44	4500000	否	無
9	1009	七堵區	1091219	住宅大樓	明德二路1~30號	125.71	6000000	否	無
10	1010	七堵區	1091230	公寓	崇孝路42巷31~60號	80.18	6000000	否	無
11	1011	七堵區	1100114	公寓	崇智街35巷1~30號	61.10	4000000	否	無
12	1012	七堵區	1100301	透天厝	六堵一街181~210號	219.66	10900000	否	無
13	1013	七堵區	1100131	公寓	明德二路1~30號	114.24	4220000	否	無
14	1014	七堵區	1100208	華廈	崇禮街61~90號	144.59	9800000	否	無
15	1015	七堵區	1100120	透天厝	麗景一街91~120號	309.99	13800000	否	無
16	1016	七堵區	1100127	華廈	大華一街121~150號	141.64	8800000	否	無
17	1017	七堵區	1100217	透天厝	長安街249巷31~60號	112.20	8800000	否	無
18	1018	七堵區	1100221	公寓	崇禮街1~30號	114.91	6000000	否	無
19	1019	七堵區	1100311	公寓	堵南街1~30號	49.71	3080000	否	無
20	1020	七堵區	1100221	華廈	實踐路151~180號	34.67	2650000	否	無
21	1021	七堵區	1100303	華廈	實踐路151~180號	34.67	2300000	否	無
22	1022	中山區	1100126	公寓	復興路212巷19弄1~30號	76.73	2600000	否	無
23	1023	中山區	1100129	華廈	復興路334巷1~30號	29.94	1800000	否	無
24	1024	中山區	1100124	透天厝	安一路61~90號	168.57	22000000	否	無
25	1025	中山區	1100212	公寓	安一路61~90號	71.34	4250000	否	無
26	1026	中山區	1100212	公寓	安一路61~90號	71.33	12000000	否	無
27	1027	中山區	1100105	公寓	成功二路91~120號	46.99	4500000	否	無
28	1028	中山區	1100224	公寓	德安路42巷1~30號	94.81	2220000	否	無
29	1029	中山區	1100121	公寓	中山三路103巷181~210號	94.46	2220000	否	無
30	1030	中山區	1100221	公寓	復興路271~300號	70.80	2300000	否	無
31	1031	中山區	1091115	透天厝	中華路46巷11弄1~30號	162.24	3300000	否	無
32	1032	中山區	1100205	公寓	復興路1~30號	55.67	1200000	否	無
33	1033	中正區	1091229	公寓	北寧路271~300號	60.26	2000000	否	無
34	1034	中正區	1100206	華廈	調和街226巷1~30號	83.35	2820000	否	無
35	1035	中正區	1100207	住宅大樓	觀海街91~120號	32.25	1600000	否	無
36	1036	中正區	1100114	透天厝	調和街266巷52弄1~30號	230.84	11360000	否	無
37	1037	中正區	1100120	住宅大樓	新豐街481~510號	93.54	3700000	否	無

已成功執行查詢。

localhost (15.0 RTM) | sa (55) | Building | 00:00:00 | 256 資料列

03 JDBC

CLASS(DAO) 介紹

BuildItem

value object
getter and setter 方法

BuildDAO

基本連結資料庫方法
新增、修改、刪除、查詢

BuildDAOFactory
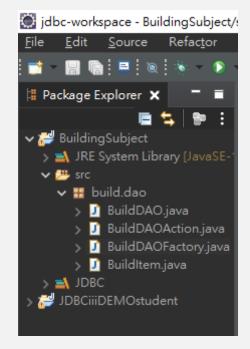
BuildDAOAction

其他邏輯運算
方法的執行

BuildDAO

```java
build/dao/BuildDAO.java - Eclipse IDE
igate   Search   Project   Run   Window   Help

BuildDAO.java ✕
BuildingSubject ▸ src ▸ build.dao ▸ BuildDAO ▸ findByID(int) : BuildItem

35          conn.close();
36          System.out.println("連線關閉");
37      }
38  }
39
40      //新增資料方法
41
42
43●     public void addBuildItem(BuildItem b) throws SQLException  {
44          String sqlStr = "Insert into building Values (?,?,?,?,?,?,?,?)";
45          PreparedStatement preState = conn.prepareStatement(sqlStr);
46          preState.setString(1,b.getBd_region());
47          preState.setString(2,b.getBd_dealdate());
48          preState.setString(3,b.getBd_buildtype());
49          preState.setString(4,b.getBd_loc());
50          preState.setDouble(5,b.getBd_area());
51          preState.setInt(6,b.getBd_price());
52          preState.setString(7,b.getBd_parking());
53          preState.setString(8,b.getBd_remark());
54
55          preState.execute();
56          preState.close();
57      }
58
59
60      //修改方法：利用ID及區域修改房屋面積
61
62●     public Integer updateAreaByIdAndRe(BuildItem b) throws SQLException {
63          String sqlStr = "Update building Set area=? where id=? and region=?";

Console ✕
consoles to display at this time.
```

BuildDAOAction

```java
jdbc-workspace - BuildingSubject/src/build/dao/BuildDAOAction.java - Eclipse IDE
File   Edit   Source   Refactor   Navigate   Search   Project   Run   Window   Help

Package Explorer ✕               *BuildDAOAction.java ✕
                                 BuildingSubject ▸ src ▸ build.dao ▸ BuildDAOAction ▸ storeFileToDB() : void
BuildingSubject
  JRE System Library [JavaSE-
  src
    build.dao              16
      BuildDAO.java        17
      BuildDAOAction.java  18
      BuildDAOFectory.java 19      //  讀取CSV並且寫入DB
      BuildItem.java       20
  JDBC                     21●     public void storeFileToDB() throws SQLException {
    mssql-jdbc-9.2.1.jre11.jar 22
  JDBCiiiDEMOstudent       23          BuildItem bc = new BuildItem();
                           24
                           25          File file = new File("C:\\JDBCfile\\build\\building.csv");
                           26          try (FileInputStream fis = new FileInputStream(file);
                           27              InputStreamReader isr = new InputStreamReader(fis, "UTF8");
                           28              BufferedReader br = new BufferedReader(isr);) {
                           29
                           30              String line = null;
                           31              while ((line = br.readLine()) != null) {
                           32                  String[] coArray = line.split(",");// 用逗號切割
                           33
                           34                  bc.setBd_region(coArray[0]);
                           35                  bc.setBd_dealdate(coArray[1]);
                           36                  bc.setBd_buildtype(coArray[2]);
                           37                  bc.setBd_loc(coArray[3]);
                           38                  bc.setBd_area(Double.parseDouble(coArray[4]));
                           39                  bc.setBd_price(Integer.parseInt(coArray[5]));
                           40                  bc.setBd_parking(coArray[6]);
                           41                  bc.setBd_remark(coArray[7]);
                           42                  bDAO.addBuildItem(bc);
                           43
                           44              }

Console ✕
No consoles to display at this time.
```

BuildDAO

```java
        conn.close();
        System.out.println("連線關閉");
      }
  }

//新增資料方法

public void addBuildItem(BuildItem b) throws SQLException  {
    String sqlStr = "Insert into building Values (?,?,?,?,?,?,?,?)";
    PreparedStatement preState = conn.prepareStatement(sqlStr);
    preState.setString(1,b.getBd_region());
    preState.setString(2,b.getBd_dealdate());
    preState.setString(3,b.getBd_buildtype());
    preState.setString(4,b.getBd_loc());
    preState.setDouble(5,b.getBd_area());
    preState.setInt(6,b.getBd_price());
    preState.setString(7,b.getBd_parking());
    preState.setString(8,b.getBd_remark());

    preState.execute();
    preState.close();
}


//修改方法：利用ID及區域修改房屋面積

public Integer updateAreaByIdAndRe(BuildItem b) throws SQLException {
    String sqlStr = "Update building Set area=? where id=? and region=?";
```

BuildDAOAc

```java
        BufferedReader br = new BufferedReader(isr);) {

        String line = null;
        while ((line = br.readLine()) != null) {
            String[] coArray = line.split(",");// 用逗號切割

            bc.setBd_region(coArray[0]);
            bc.setBd_dealdate(coArray[1]);
            bc.setBd_buildtype(coArray[2]);
            bc.setBd_loc(coArray[3]);
            bc.setBd_area(Double.parseDouble(coArray[4]));
            bc.setBd_price(Integer.parseInt(coArray[5]));
            bc.setBd_parking(coArray[6]);
            bc.setBd_remark(coArray[7]);
            bDAO.addBuildItem(bc);

        }

    } catch (FileNotFoundException e) {
        System.out.println("e.getMessage()" + e.getMessage());
    } catch (UnsupportedEncodingException e) {
        System.out.println("e.getMessage()" + e.getMessage());
    } catch (IOException e1) {
        System.out.println("e1.getMessage()" + e1.getMessage());
    }

}

// 將查詢到的資料顯示
```

BuildDAO

```java
53          preState.setString(8,b.getBd_remark());
54
55          preState.execute();
56          preState.close();
57      }
58
59
60      //修改方法：利用ID及區域修改房屋面積
61
62      public Integer updateAreaByIdAndRe(BuildItem b) throws SQLException {
63          String sqlStr = "Update building Set area=? where id=? and region=?";
64          PreparedStatement preState = conn.prepareStatement(sqlStr);
65          preState.setDouble(1, b.getBd_area());
66          preState.setInt(2, b.getBd_id());
67          preState.setString(3, b.getBd_region());
68
69          int sta = preState.executeUpdate();
70          preState.close();
71          return sta;
72
73      }
74
75
76      //查詢方法：利用ID資料查詢
77
78
79      public BuildItem findByID(int id) throws SQLException {
80          String sqlStr = "Select * from building where id = ?";
81          PreparedStatement preState = conn.prepareStatement(sqlStr);
```

BuildDAOAction

```java
92              }
93
94          }
95
96      }
97
98      // 修改資料：利用ID及區域修改房屋面積
99      public void updateArea() {
100         @SuppressWarnings("resource")
101         Scanner scanner = new Scanner(System.in);
102
103         while (true) {
104             try {
105                 BuildItem update = new BuildItem();
106
107                 System.out.println("請輸入要修改資料的區域名稱，例如:安樂區");
108                 String answer2 = scanner.nextLine();
109                 update.setBd_region(answer2);
110
111                 System.out.println("請輸入要修改資料的四位數字的ID號碼，例如:1100");
112                 int answer = Integer.valueOf(scanner.nextLine());
113                 update.setBd_id(answer);
114
115                 System.out.println("請輸入更改後的面積資料,例如：100.20");
116                 Double answer3 = Double.valueOf(scanner.nextLine());
117                 update.setBd_area(answer3);
118
119                 Integer sta = bDAO.updateAreaByIdAndRe(update);
120
```

BuildDAO

```java
        preState.setString(8,b.getBd_remark());

        preState.execute();
        preState.close();
    }

    //修改方法：利用ID及區域修改房屋面積

    public Integer updateAreaByIdAndRe(BuildItem b) throws SQLException {
        String sqlStr = "Update building Set area=? where id=? and region=?";
        PreparedStatement preState = conn.prepareStatement(sqlStr);
        preState.setDouble(1, b.getBd_area());
        preState.setInt(2, b.getBd_id());
        preState.setString(3, b.getBd_region());

        int sta = preState.executeUpdate();
        preState.close();
        return sta;

    }

    //查詢方法：利用ID資料查詢

    public BuildItem findByID(int id) throws SQLException {
        String sqlStr = "Select * from building where id = ?";
        PreparedStatement preState = conn.prepareStatement(sqlStr);
```

BuildDAOAction

```java
            Integer sta = bDAO.updateAreaByIdAndRe(update);

            if (sta >= 1) {
                System.out.println("已更新完成");
            } else {
                System.out.println("更新失敗,資料輸入錯誤");
            }

            System.out.println("請問是否還有要繼續更新資料?(1.繼續 2.退出)");

            int anYN = Integer.valueOf(scanner.nextLine());

            if (anYN == 1) {
                continue;
            } else {
                break;
            }

        } catch (SQLException e) {
            System.out.println("錯誤訊息: 錯誤! 請重新輸入 ");
            e.printStackTrace();
        } catch (NullPointerException e) {
            System.out.println("錯誤訊息: 錯誤! 請重新輸入 ");
        } catch (NumberFormatException e) {
            System.out.println("錯誤訊息: 錯誤! 請重新輸入 ");
        }
    }
}
```

BuildDAO

```java
//查詢方法：利用ID資料查詢

public BuildItem findByID(int id) throws SQLException {
    String sqlStr = "Select * from building where id = ?";
    PreparedStatement preState = conn.prepareStatement(sqlStr);
    preState.setInt(1, id);
    ResultSet rs = preState.executeQuery();

    BuildItem b = null;

    if(rs.next()) {

        b = new BuildItem();
        b.setBd_id(rs.getInt("id"));
        b.setBd_region(rs.getString("region"));
        b.setBd_dealdate(rs.getString("dealdate"));
        b.setBd_buildtype(rs.getString("buildtype"));
        b.setBd_loc(rs.getString("loc"));
        b.setBd_area(rs.getDouble("area"));
        b.setBd_price(rs.getInt("price"));
        b.setBd_parking(rs.getString("parking"));
        b.setBd_remark(rs.getString("remark"));

    }

    rs.close();
    preState.close();
    return b;
```

BuildDAOAction

```java
// 將查詢到的資料顯示

public void queryData() throws SQLException {
    @SuppressWarnings("resource")
    Scanner scanner = new Scanner(System.in);

    while (true) {
        try {
            System.out.println("請輸入四位數字的ID號碼");
            int answer = Integer.valueOf(scanner.nextLine());
            BuildItem bq = bDAO.findByID(answer);
            System.out.println("id:" + bq.getBd_id());
            System.out.println("region:" + bq.getBd_region());
            System.out.println("dealdate:" + bq.getBd_dealdate());
            System.out.println("buildingtype:" + bq.getBd_buildtype());
            System.out.println("loc:" + bq.getBd_loc());
            System.out.println("area:" + bq.getBd_area());
            System.out.println("price:" + bq.getBd_price());
            System.out.println("parking:" + bq.getBd_parking());
            System.out.println("remark:" + bq.getBd_remark());
            System.out.println("請問是否還有要繼續查詢?(1.繼續 2.退出)");

            int anYN = Integer.valueOf(scanner.nextLine());

            if (anYN == 1) {
                continue;
            } else {
                break;
            }
        }
```

BuildDAO

```java
//查詢方法：利用ID資料查詢

public BuildItem findByID(int id) throws SQLException {
    String sqlStr = "Select * from building where id = ?";
    PreparedStatement preState = conn.prepareStatement(sqlStr);
    preState.setInt(1, id);
    ResultSet rs = preState.executeQuery();

    BuildItem b = null;

    if(rs.next()) {

        b = new BuildItem();
        b.setBd_id(rs.getInt("id"));
        b.setBd_region(rs.getString("region"));
        b.setBd_dealdate(rs.getString("dealdate"));
        b.setBd_buildtype(rs.getString("buildtype"));
        b.setBd_loc(rs.getString("loc"));
        b.setBd_area(rs.getDouble("area"));
        b.setBd_price(rs.getInt("price"));
        b.setBd_parking(rs.getString("parking"));
        b.setBd_remark(rs.getString("remark"));

    }

    rs.close();
    preState.close();
    return b;
```

BuildDAOAction

```java
        System.out.println("dealdate:" + bq.getBd_dealdate());
        System.out.println("buildingtype:" + bq.getBd_buildtype());
        System.out.println("loc:" + bq.getBd_loc());
        System.out.println("area:" + bq.getBd_area());
        System.out.println("price:" + bq.getBd_price());
        System.out.println("parking:" + bq.getBd_parking());
        System.out.println("remark:" + bq.getBd_remark());
        System.out.println("請問是否還有要繼續查詢?(1.繼續 2.退出)");

        int anYN = Integer.valueOf(scanner.nextLine());

        if (anYN == 1) {
            continue;
        } else {
            break;
        }

    } catch (SQLException e) {
        System.out.println("錯誤訊息：錯誤！請重新輸入 ");
        e.printStackTrace();
    } catch (NullPointerException e) {
        System.out.println("錯誤訊息：錯誤！請重新輸入 ");
    } catch (NumberFormatException e) {
        System.out.println("錯誤訊息：錯誤！請重新輸入 ");
    }

}

}
```

刪除資料
_BuildDAO、BuildDAOAction

BuildDAO

```java
            return b;
        }

    //刪除資料方法

    public void deleteMemberById(int id) throws SQLException {
        String sqlStr = "Delete from building where id=?";
        PreparedStatement preState = conn.prepareStatement(sqlStr);

        preState.setInt(1, id);
        preState.execute();
        preState.close();
    }

    //查詢全部資料

    public void querryALL() throws SQLException {
        String sqlStr = "Select * from building";
        Statement stmt = conn.createStatement();
        ResultSet rs =stmt.executeQuery(sqlStr);
        System.out.println("ID"+","+"區域"+ ","+"交易日期"+ ","+"房屋型態"+ ","+"路段"+
                ","+"面積"+","+"價格"+","+"是否含車位"+ ","+"備註");

        while(rs.next()) {
```

BuildDAOAction

```java
            System.out.println("請輸入要使用的選項：1.查詢總表 2.修改資料 3.查詢單筆資料 4.退出程式");
            int answer = Integer.valueOf(scanner.nextLine());
            if (answer == 1) {
                bDAO.querryALL();// 查詢全部資料表的資料
            } else if (answer == 2) {
                ac.updateArea();// 更新資料
            } else if (answer == 3) {
                ac.queryData();// 將查詢到的資料顯示
            } else if (answer == 4) {
                break;
            } else {
                System.out.println("錯誤!請重新輸入，請輸入對應的阿拉伯數字，例如：1");
            }

            System.out.println("請問是否要繼續執行其他程式，1.是 2.否");
            int answer2 = Integer.valueOf(scanner.nextLine());
            if (answer2 == 1) {
                continue;
            } else {
                break;
            }
        }

        bDAO.deleteMemberById(1005);// 刪除資料

        bDAO.closeConn();
    }
}
```

main()方法中

BuildDAO

```
117          preState.execute();
118          preState.close();
119      }
120
121      //查詢全部資料
122
123      public   void  querryALL() throws SQLException {
124          String sqlStr = "Select * from building";
125          Statement stmt = conn.createStatement();
126          ResultSet rs =stmt.executeQuery(sqlStr);
127          System.out.println("ID"+","+"區域"+ ","  +"交易日期"+ ","  +"房屋型態"+ "," +"路段"+
128                  ","+"面積"+","+"價格"+","+"是否含車位"+ ","  +"備註");
129
130          while(rs.next()) {
131
132
133
134
135              System.out.println(rs.getInt(1)+ "," +rs.getString(2)+ "," + rs.getString(3) + "," + rs.getString(4) + "," +rs.getString(5)+
136                  ","+rs.getDouble(6)+","+rs.getInt(7)+","+rs.getString(8)+ "," +rs.getString(9));
137          }
138      }
139
140
141
142
143
144
145
```

BuildDAOAction

```
145                      }
144            }
145        }
146
147        // 主程式
148°       public static void main(String[] args) throws SQLException {
149            BuildDAOAction ac = new BuildDAOAction();
150
151            bDAO.createConn();//開連線
152
153            ac.storeFileToDB();// 讀取CSV並且寫入DB
154
155            @SuppressWarnings("resource") // 註解掉警告
156            Scanner scanner = new Scanner(System.in);
157
158            while (true) {
159                System.out.println("請輸入要使用的選項：1.查詢總表 2.修改資料 3.查詢單筆資料 4.退出程式");
160                int answer = Integer.valueOf(scanner.nextLine());    main()方法中
161                if (answer == 1) {
162                    bDAO.querryALL();// 查詢全部資料表的資料
163                } else if (answer == 2) {
164                    ac.updateArea();// 更新資料
165                } else if (answer == 3) {
166                    ac.queryData();// 將查詢到的資料顯示
167                } else if (answer == 4) {
168                    break;
169                } else {
170                    System.out.println("錯誤!請重新輸入，請輸入對應的阿拉伯數字，例如：1");
171                }
```

```java
package build.dao;

public class BuildDAOFactory {



    private static final BuildDAO buildDAO = createBuildDAO();

    private static BuildDAO createBuildDAO() {
        BuildDAO bDAO = new BuildDAO();
        return bDAO;
    }

    public static BuildDAO getBuildDAO() {
        return buildDAO;
    }

}
```

04 cmd使用者輸入

java -cp build.jar;lib/*;. build.dao.BuildDAOAction

簡報結束
謝謝大家