Class insurance type

Dalam dunia asuransi terdapat beberapa jenis. Ada asuransi yang hanya menanggung kerugian akibat gempa bumi saja ada juga yang hanya karena kebakaran saja dan ada juga yang menanggung semua jenis kerugian. Oleh karena itu data asuransi harus dilengkapi dengan jenis perlindungan yang disetujui oleh pihak asuransi dan klientnya.

This is the coding :

Write this in common class

public class InsuranceTypeData

{

public string Id { get; set; }

public string TypeName { get; set; }

public string Explain { get; set; }

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using MySql.Data.MySqlClient;

using System.Windows.Forms;

namespace Core\_Inventaris

{

public class InsuranceType

{

// variable definition

Connection Connetion = new Connection();

MySqlDataReader Reader;

public List<InsuranceTypeData> getData()

{

// function for get all insurance data from database server

Reader = Connetion.selectTable("select \* from insurancetype order by id");

List<InsuranceTypeData> InsDatas = new List<InsuranceTypeData>();

while (Reader.Read())

{

InsuranceTypeData InsData = new InsuranceTypeData();

InsData.Id = Reader.GetString(0);

InsData.TypeName = Reader.GetString(1);

InsData.Explain = Reader.GetString(2);

InsDatas.Add(InsData);

}

Reader.Close();

return InsDatas;

}

public bool toServer()

{

// function for chekking connection to database server

List<string> Database = Connetion.readSever();

try

{

if (Connetion.connection(Database[0], Database[1], Database[2]))

{

return true;

}

else

{

return false;

}

}

catch(Exception e)

{

MessageBox.Show(e.Message);

return false;

}

}

public string deleteData(string ID)

{

// delete query

Reader = Connetion.selectTable("Delete from insurancetype where id='" + ID + "'");

Reader.Close();

return "Deleted!";

}

public string insertData(string ID, string Typename, string Explaination)

{

// insert data query

Reader = Connetion.selectTable("insert into insurancetype (id,Typename,Explaination) values ('" + ID + "','" +

Typename + "','" + Explaination + "')");

Reader.Close();

return "Data Saved!";

}

public string updatetData(string ID, string Typename, string Explaination)

{

// update data query

Reader = Connetion.selectTable("update insurancetype set typeName = '" + Typename + "', explaination='" + Explaination

+ "' where id = '" + ID + "'");

Reader.Close();

return "Data Update!";

}

}

}