

COMP 3111: Software Engineering

HKeInvestData.cs Documentation

TO USE THE METHODS IN HKEINVESTDATA.CS

1. Add the following directives to your code file.

```
using System.Data;  
using System.Data.SqlClient;  
using HKeInvestWebApplication.Code_File;
```

2. Declare an instance of the HKeInvestData class, such as

```
HKeInvestData myHKeInvestData = new HKeInvestData();
```

RETRIEVING DATA FROM TABLES

```
public DataTable getData(string sql)
```

Parameters: sql - the SQL SELECT statement

Return type: a DataTable containing the result of the SQL SELECT statement

null - an error occurred in processing the SQL SELECT statement

Example usage: myHKeInvestData.getData("select [name] from [Person]");

RETRIEVING AGGREGATE VALUES FROM TABLES

```
public decimal getAggregateValue(string sql)
```

Parameters: sql - the aggregate SQL SELECT statement

The aggregate functions are: avg, count, max, min, sum, stdev, var

Return type: the aggregate value

-1 - an error occurred in processing the aggregate SQL SELECT statement

Example usage: myHKeInvestData.getAggregateValue("select count(*) from [Person]");

UPDATING DATA IN TABLES

```
public SqlTransaction beginTransaction()
```

```
public void commitTransaction(SqlTransaction trans)
```

```
public void setData(string sql, SqlTransaction trans)
```

Parameters: sql - the SQL INSERT, UPDATE or DELETE statement

trans - a transaction created by a beginTransaction() statement

Example usage:

```
SqlTransaction trans = myHKeInvestData.BeginTransaction();  
myHKeInvestData.setData("update [Person] set [FirstName] =  
    'John' where [lastName]='Chan'", trans);  
myHKeInvestData.commitTransaction(trans);
```

Note: When updating multiple tables at one time, place all update statements between a BeginTransaction statement and a CommitTransaction statement.