PROBLEM STATEMENT

Read the Problem Statement File

INSTALLATION

Go to root directory of the project and build the project using the following command

```
..\SoliumStockOption> gradle build
```

Once the build is successful, drop the input files in the following folder

```
..\ SoliumStockOption\build\libs\
```

Run the application using the following command

```
>java -jar SoliumStockOption.jar < sample.def
```

HIGH LEVEL TECHNICAL EXPLANATION

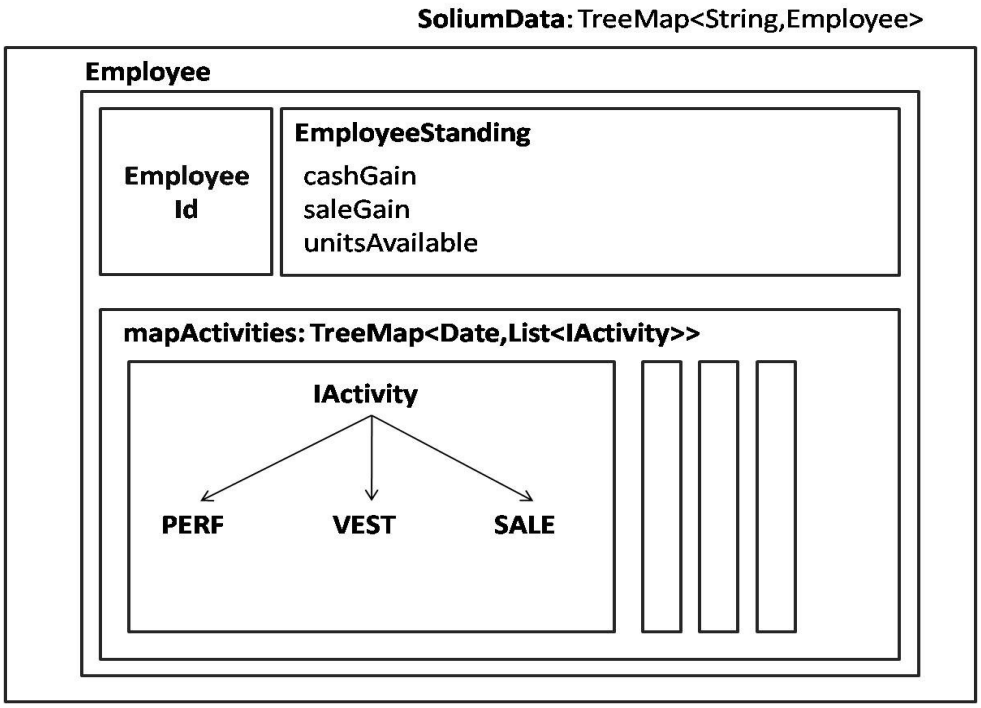
The whole application is run as an instance of SoliumStock class that consists of the basic functions of

1) readInput – reads and parses the input
2) computeGain – required conputation to calculate the gain – both sale gain and overall vest gain
3) displayOutput – Displays the output

The data structure used is SoliumData. This will contain the data for all the employees along with all the transactions. There will only be a single instance of SoliumData (Singleton) so that there is only one copy of data that can maintain all the necessary data.

The SoliumData consists of a sorted map between EmployeeId and Employee Data structure. Employee consists of the following

1) EmployeeId
2) EmployeeStanding
3) Sorted map between a date and the list of activities for that employee



Each Line of Input that has any transaction (like vesting) corresponds to an activity – Each activity that has one of the following operations

1) PERF
2) SALE
3) VEST

And each comma delimited item is considered to be an activity component. Examples of activity components are EmployeeDate, VestDate, Number of Units, GrantPrice, PerformanceMultiplier. Common components are available in class GeneralActivity and all the activities implements an Interface IActivity. This will allow any new type of activity that can be introduced into the system.

The final line of the input file is put in a ComputeParams class that has a date and a price. Multiple instances can be created so that in case the program is to be enhanced to find the maximum gain among a list of given options, there can be multiple instances created of ComputeParams.