

C# coding test

Disclaimer

Contents of this document are confidential and should be used only for recruitment purposes. Disclosing it (or any of its parts) to any third party, publishing on internet, etc. is forbidden.

Introduction

We would like you to write some C# application (it should be delivered in form of MS Visual Studio solution) according to instructions given below. Task has been designed in a way that development should take one day. However for your convenience you're given 5 days to prepare your solution and send it back to us. **Thus we expect you to focus on high quality of code by using the best coding standards and principles you know, unit tests etc.**

Test application description

The role of test application is to monitor some *input directory* with some defined frequency and check whether any new *input files* appeared there. If some new *input files* are available data from these files should be loaded into application and presented to the user using some GUI.

Application should fulfill following requirements:

1. *Input directory* should be checked for appearance of new files with some defined frequency (e.g. 5 seconds). It should be possible to change monitoring frequency without code recompilation (application restart is allowed).
2. Path to *input directory* should be configurable (it should be possible to change the directory without code recompilation; restart of the application is allowed).
3. *Input files* that will be loaded into application come in three formats XML, CSV and TXT (you will find sample content of each file format on next pages). For each file format specific loader should be created.
4. All the files contain data for the same entities (which represent some made up trades data) and because of that there should be one type defined to handle data items in the application irrespectively from which file format they've been loaded.
5. Application architecture should provide a possibility of creating additional loaders (a kind of plugins) by other developers. Once new loaders are created it should be possible to use them without a need to recompile application's source code. Set of loaders that will be used by particular instance of application should be configurable.
6. The GUI of an application should be prepared in WPF (preferred) or Windows Forms technology.
7. It should be taken into account that there may be a situation when the number of files that will be loaded at one time and their size may be significantly big. Application should be written in a way that loading of big amounts of data should not cause "freezing" of main window.
8. When application detects that new *input files* have appeared in *input directory* it should load all the new files in parallel and display data from specific file immediately once it's processed (don't wait for all loaders to complete their jobs).

Sample input data

Sample input data for each file type is available on next pages. Input file should contain all the contents given after "Content:" label.

Results delivery

1. Make a ZIP archive of your solution project. All binary files (.DLL,.EXE) should be removed from solution before making the ZIP file.
2. Send it back to person, who provided you this PDF description.

CSV data file

Columns:

Date	Open	High	Low	Close	Volume
------	------	------	-----	-------	--------

Content:

2013-5-20	30.16	30.39	30.02	30.17	1478200
2013-5-17	29.77	30.26	29.77	30.26	2481400
2013-5-16	29.78	29.94	29.55	29.67	1077000
2013-5-15	29.63	29.99	29.63	29.98	928700
2013-5-14	29.53	29.77	29.48	29.6	1065900
2013-5-13	29.33	29.59	29.09	29.51	1005800
2013-5-10	29.89	29.91	29.52	29.83	1831000
2013-5-9	29.65	29.68	29.2	29.3	1693600
2013-5-8	29.65	29.99	29.54	29.72	1009800
2013-5-7	29.46	29.62	29.18	29.49	1390200
2013-5-6	28.8	28.9	28.64	28.78	744800
2013-5-3	28.85	29.02	28.45	28.56	1087800
2013-5-2	28.1	28.33	28	28.24	1472500
2013-5-1	27.94	28.19	27.61	27.7	1006900

Text data file

Columns:

Date	Open	High	Low	Close	Volume
------	------	------	-----	-------	--------

Content:

Date;Open;High;Low;Close;Volume

2013-5-20	30.16	30.39	30.02	30.17	1478200
2013-5-17	29.77	30.26	29.77	30.26	2481400
2013-5-16	29.78	29.94	29.55	29.67	1077000
2013-5-15	29.63	29.99	29.63	29.98	928700
2013-5-14	29.53	29.77	29.48	29.6	1065900
2013-5-13	29.33	29.59	29.09	29.51	1005800
2013-5-10	29.89	29.91	29.52	29.83	1831000
2013-5-9	29.65	29.68	29.2	29.3	1693600
2013-5-8	29.65	29.99	29.54	29.72	1009800
2013-5-7	29.46	29.62	29.18	29.49	1390200
2013-5-6	28.8	28.9	28.64	28.78	744800
2013-5-3	28.85	29.02	28.45	28.56	1087800
2013-5-2	28.1	28.33	28	28.24	1472500
2013-5-1	27.94	28.19	27.61	27.7	1006900

XML data file

Content:

```
<?xml version="1.0" encoding="utf-8" ?>
<values>
  <value date="2013-5-20" open="30.16" high="30.39" low="30.02" close="30.17" volume="1478200" />
  <value date="2013-5-17" open="29.77" high="30.26" low="29.77" close="30.26" volume="2481400" />
  <value date="2013-5-16" open="29.78" high="29.94" low="29.55" close="29.67" volume="1077000" />
  <value date="2013-5-15" open="29.63" high="29.99" low="29.63" close="29.98" volume="928700" />
  <value date="2013-5-14" open="29.53" high="29.77" low="29.48" close="29.6" volume="1065900" />
  <value date="2013-5-13" open="29.33" high="29.59" low="29.09" close="29.51" volume="1005800" />
  <value date="2013-5-10" open="29.89" high="29.91" low="29.52" close="29.83" volume="1831000" />
  <value date="2013-5-9" open="29.65" high="29.68" low="29.2" close="29.3" volume="1693600" />
  <value date="2013-5-8" open="29.65" high="29.99" low="29.54" close="29.72" volume="1009800" />
  <value date="2013-5-7" open="29.46" high="29.62" low="29.18" close="29.49" volume="1390200" />
  <value date="2013-5-6" open="28.8" high="28.9" low="28.64" close="28.78" volume="744800" />
  <value date="2013-5-3" open="28.85" high="29.02" low="28.45" close="28.56" volume="1087800" />
  <value date="2013-5-2" open="28.1" high="28.33" low="28" close="28.24" volume="1472500" />
  <value date="2013-5-1" open="27.94" high="28.19" low="27.61" close="27.7" volume="1006900" />
</values>
```