

Dustin Chang – V00713939
Nov 16, 2015

Assignment 2

GitHub

GitHub Link: https://github.com/dustinchang/Seng330_Assign2

Install

INSTALL.txt File that is included in project:

#Tools needed to build project

- Git
- Google protobuf library

#Instructions to build project

- Download or clone repository files from GitHub
https://github.com/dustinchang/Seng330_Assign2
- Navigate to the src directory
- Run 'make'
- Then you can run ./main_cpp which is where all the execution of prototype class design takes place and the addition of protocol buffer data is
- Then run ./list_portfolio_cpp to list all the stored data in the protocol buffers

Prototype Design Pattern

//Factory to assist prototype instances and create their clones

```
class ObjFactory {
    /*!
    * Static prototype template classes
    */
    static Index* child1;
    static Index* child2;
public:
    /*!
    * Initialization function of derived classes
    */
    static void init() {
        child1 = new Stock("Stock");
        child2 = new MutualFund("MutualFund");
    }
    /*!
    * Function to clone derived class 1
    */
    static Index* getNameVal1() {
        return child1->clone();
    }
};
```

```

    }
    /*!
    * Function to clone derived class 2
    */
    static Index* getNameVal2() {
        return child2->clone();
    }
};

```

Google Protocol Buffers

This is a small sample of part of the Google Protocol Buffers code where it adds what the user wants as an investment id and name, more of the code can be found at the [GitHub link](#).

```

/*!
* Prompts user for interaction and adds the name and id of the investment they
want to add
*/
void PromptForInvestment(invest::Investment* investment) {
    cout << "Enter investment ID number: ";
    int id;
    cin >> id;
    investment->set_id(id);
    cin.ignore(256, '\n');

    cout << "Enter type of investment: ";
    getline(cin, *investment->mutable_iname());
}

```

```

Index* ObjFactory::child1 = 0;
Index* ObjFactory::child2 = 0;

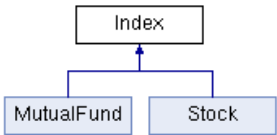
```

Doxygen

Doxygen produces an html viewable page of the comments created throughout the project. A few screenshots can be show below. For the full view, you can clone the [GitHub](#) and view the index file.

Index Class Reference abstract

Inheritance diagram for Index:



Public Member Functions

| |
|--|
| virtual Index * clone ()=0 |
| string getName () |
| int get_ID () |
| virtual ~Index () |

Protected Attributes

| |
|---------------------|
| string iname |
| int id |

Detailed Description

This is the **Index** Prototype class that will be the template for other classes

Constructor & Destructor Documentation

virtual Index::~~Index ()inlinevirtual

Virtual Destructor function for Prototype Template

Member Function Documentation

virtual [Index*](#) Index::clone ()

pure virtual

Template for clone

Implemented in [MutualFund](#), and [Stock](#).

int Index::get_ID ()

inline

Getter method for id

string Index::getName ()

inline

Getter method for iname

Member Data Documentation

int Index::id

protected

string Index::iname

protected

Protected string iname

The documentation for this class was generated from the following file:

- [/Users/dustin/Documents/Seng330_Assign2/src/main.cpp](#)

ObjFactory Class Reference

Static Public Member Functions

| | |
|----------------|----------------|
| static void | init () |
| static Index * | getNameVal1 () |
| static Index * | getNameVal2 () |

Member Function Documentation

| | | |
|--|--------|--------|
| static Index* ObjFactory::getNameVal1 () | inline | static |
| Function to clone derived class 1 | | |
| static Index* ObjFactory::getNameVal2 () | inline | static |
| Function to clone derived class 2 | | |
| static void ObjFactory::init () | inline | static |
| Initialization function of derived classes | | |

▼ Index

 **MutualFund**

 **Stock**

▼ **Message**

 **invest::Investment**

 **invest::Portfolio**

 **ObjFactory**

 **invest::StaticDescriptorInitializer_invest_2eproto**