```
/*************************************************************************
                          DataManager  -  description
                             -------------------
    debut                : 23/11/2015
    copyright            : (C) 2015 by Edern Haumont & Nicolas Six
*************************************************************************/


//-------- Interface of the class DataManager (file DataManager.h) -------
#ifndef TP_CPP_DATAMANAGER_H
#define TP_CPP_DATAMANAGER_H

//------------------------------------------------------------ Includes
#include <iostream>
#include <string>
#include <fstream>
#include <map>
#include <vector>
#include <sstream>

#include "LogOtherInfos.h"
#include "GraphGenerator.h"

//------------------------------------------------------------ Constants
const int DATA_TAB_SIZE = 2;

//------------------------------------------------------------------Typedefs
typedef std::vector< LogOtherInfos > dataHourLevel;
typedef std::map< std::string, dataHourLevel* > dataDestinationLevel;
typedef std::map< std::string, dataDestinationLevel* > dataFromLevel;

typedef std::pair< std::string, int> pageAndHits;

//-----------------------------------------------------------------------
// The class DataManager is the main class of the program. It contains all
// data structures, creates, manages and make calculations on them.
//-----------------------------------------------------------------------
class DataManager {

//------------------------------------------------------------------ PUBLIC
public:


//------------------------------------------------------- Public methods
    int LoadLogFile(const std::string &logFilePath);
    // User guide :
    // This method is used to load a logfile in the program.
    // The parameter logFilePath is the path to the file.

    int Request(const bool optionT=false,const int tHour=-1,const bool optionE=false,const bool
optionG=false,const std::string &outputFile="");
    // User guide :
    // Generate the outputs of the program.
    // The parameters correspond to user options, outputFile is the path
    // to a generated .dot file

//------------------------------------------- Constructors - destructors
    DataManager();
    // Use at the beginning of the program
    virtual ~DataManager();
    // Use at the end of the program

//------------------------------------------------------- Private methods
private:
    int add(const std::string &referrer,const std::string &destination, unsigned hour, unsigned int
httpCode,const LogOtherInfos &other);


    unsigned transformToTabIndex(int httpCode) const;

    static bool compareDateAndHits(const pageAndHits &A, const pageAndHits &B);
    bool isNotExcludedDocument(const std::string &pagePath) const;
```

```
    /* return: false if the extension is in the list of excluded extension
     */

//------------------------------------------------------- Private atributs

    GraphGenerator * graph;
    dataFromLevel * data[DATA_TAB_SIZE];
    std::vector< std::string > excludedExtension;

};


#endif //TP_CPP_DATAMANAGER_H
```