

```

/*****
                                Log - description
                                -----
    début                      : 23/11/2015
    copyright                   : (C) 2015 par mfallouh_mvirsolv
*****/

//----- Interface de la classe <Log> (fichier Log.h) -----
#if !defined ( LOG_H )
#define LOG_H
#include <string>
using namespace std;
//----- Interfaces utilisées

//----- Constantes
const char SEP = ' ', SEP_DATE_FIN = ']', SEP_REQ = '"', SEP_URL = '/', SEP_INT =
'?', SEP_PVIRG = ';'; // les séparateurs
const string SEP_PARAM = "&";
const string URL_LOCALE = "http://intranet-if.insa-lyon.fr", ERREUR = "ERREUR";
//----- Types

//-----
// Rôle de la classe <Log>
//
// Il s'agit de la classe stockant l'équivalent d'un log.
// On y retrouve toutes les informations excepté le type de requête (GET, POST...)
// et l'adresse cible (pour éviter des redondances).
//
//-----

class Log
{
//----- PUBLIC

public:
//----- Méthodes publiques

//----- Surcharge d'opérateurs
    Log & operator = ( const Log & unLog ); // déclaré mais non défini pour
empêcher son utilisation
    // Mode d'emploi :
    //
    // Contrat :
    //

//----- Constructeurs - destructeur

    Log ( const Log & unLog ); // déclaré mais non défini pour empêcher son
utilisation
    // Mode d'emploi (constructeur de copie) :
    //
    // Contrat :
    //

    Log (const string &log );
    // Mode d'emploi :
    // Initialise tous les attributs à partir de la string en parametre.

```

```

        // Vérifie si la string est du bon format, si non, renvoie un message d'erreur
(sortie erreur).
        //
        // Contrat :
        //

        virtual ~Log ( );
        // Mode d'emploi :
        //
        // Contrat :
        //

//----- PRIVE

protected:
//----- Méthodes protégées

private:
//----- Méthodes privées
        void setErreur();
        // Mode d'emploi :
        // modifie la valeur de chaque attributs à la valeur ERREUR
        // Contrat:
        //
protected:
//----- Attributs protégés
        string ip;                // l'adresse IP à l'origine de la requête
        string logname;           // le nom d'utilisateur du visiteur
        string user;              // le nom d'utilisateur authentifié
        string date;              // la date de la requête, ainsi que le décalage par
rapport à Greenwich
        string statut;            // le retour de la requête HTML
        string taille;            // quantité de données transférées en octets
        string referer;           // adresse depuis laquelle a été émise la requête
        string url;               // URL du referer en entier
        string navigateur;        // infos navigateur
private:
//----- Attributs privés

//----- Classes amies
        friend class Graphe;
        friend class Cible;
//----- Classes privées

//----- Types privés

};

//----- Types dépendants de <Log>

#endif // LOG_H

```