

#ifndef STUDENT\_H

#define STUDENT\_H

#include <string>

#include vector

/\*\*

\* class student

\*

\*/

class student

{

public:

// Constructors/Destructors

//

/\*\*

\* Empty Constructor

\*/

student ();

/\*\*

\* Empty Destructor

\*/

virtual ~student ();

// Static Public attributes

//

// Public attributes

//

// Public attribute accessor methods

//

// Public attribute accessor methods

//

/\*\*

\*/

void checkid ()

{

}

/\*\*

\*/

void checkpass ()

{

}

protected:

// Static Protected attributes

//

// Protected attributes

//

public:

// Protected attribute accessor methods

//

protected:

public:

// Protected attribute accessor methods

//

protected:

private:

// Static Private attributes

//

// Private attributes

//

integer id;

integer password;

public:

// Private attribute accessor methods

//

private:

public:

// Private attribute accessor methods

//

/\*\*

\* Set the value of id

\* @param new\_var the new value of id

\*/

void setId (integer new\_var) {

id = new\_var;

}

/\*\*

\* Get the value of id

\* @return the value of id

\*/

integer getId () {

return id;

}

/\*\*

\* Set the value of password

\* @param new\_var the new value of password

\*/

void setPassword (integer new\_var) {

password = new\_var;

}

/\*\*

\* Get the value of password

\* @return the value of password

\*/

integer getPassword () {

return password;

}

private:

void initAttributes () ;

};

#endif // STUDENT\_H