CS 271
11-07-17

Student.h

```
class Student {

   private:

      string firstName;
      string lastName;
      int studentID;

   public:

      Student( );
      Student( string, string, int );
      string getFirstName( );
      string getLastName( );
      int getStudentID( );
      Student& setFirstName( string );
      Student& setLastName( string );
      Student& setStudentID( int );

      // returning a Student reference allows chaining like this
      // johndoe.setFirstName("James").setLastName.( "Dawson" );

}; // end class definition
```

Student.cpp

```
Student::Student( ) {

   firstName = "";
   lastName = "";
   studentID = 0;
}

Student::Student( string fname, string lname, int id ) {

   firstName = fname;
   lastName = lname;
   studentID = id;
}

Student& Student::setFirstName( string name ) {

   firstName = name;
   return *this;

}

string Student::getFirstName( ) {

   return firstName;

}


// other accessors and mutators go here
```

UndergraduateStudent.h

```
// let's derive UndergraduateStudent from class Student

class UndergraduateStudent : public Student
{
   private:
      double gpa;
      string major;

   public:

      UndergraduateStudent( );
      UndergraduateStudent( string, string, int, double, string );
      double getGPA( ) const;
      string getMajor( ) const;
      UndergraduateStudent& setGPA( double );
      UndergraduateStudent& setMajor ( string );

};
```

UndergraduateStudent.cpp

```
// Constructor with 5 parameters
// 3 of the parameters are passed to the Student constructor

UndergraduateStudent::UndergraduateStudent ( string fn, string ln, int id, double g,
  string m )  : Student ( fn, ln, id ) {

    gpa = g;
    major = m;
}

// other function definitions go here



--------------------------------

// in a main function

UndergraduateStudent garrett( "first", "last", 8000, 4.0, "CS" );
```