Problem Set 08 - Inheritance

Complete each task below. Remember to include all header files in the accompanying cpp file. Include the libraries *cctype* and *iomanip*. Furthermore, for all classes, you must define all their special member functions which should be public.

Tasks:

1. Create a header file named `"Person.h"` and define the class *Person* within the namespace *PS8*. The class must contain
   - ☐ a private string field named *firstName*.
   - ☐ a private string field named *lastName*.
   - ☐ a private string field named *middleName*.
   - ☐ a private unsigned int field named *age*.
   - ☐ a public default constructor that assigns `"John"`, `"Smith"`, the empty string, and 1 to *firstName*, *lastName*, *middleName* and *age* respectively.
   - ☐ a public constant getter method for each field.
   - ☐ a public void setter method for each string field that assigns the parameter to the field only if the parameter consists only of letters.
   - ☐ a public void setter method for *age*.
   - ☐ a public string constant method named `ToString()` that takes no parameters. It returns a string in the format

     $$x\ y\ z\ (w)$$

     where x, y, and z are the values of *firstName*, *middleName* and *lastName* respectively in capitalized format and w is the value of *age*.
   - ☐ an ostream operator that returns its outcome in the same format as `ToString()`.

   Afterward, within the accompanying cpp file, within the main function, declare a *Person* object. Modify its fields, and then, display it.

2. Create a header file named `"Student.h"` and define the class *Student* within the namespace *PS8*. The class must publicly inherit *Person* and contain
   - ☐ a private double field named *GPA*.
   - ☐ a public default constructor that assigns 0 to *GPA*.
   - ☐ a public void setter method for *GPA* that assigns the parameter to *GPA* only if it is between 0 and 5 inclusively.
   - ☐ a public constant getter method for *GPA*.
   - ☐ a public string constant method named `ToString()` that takes no parameters. It returns a string in the format

     $$x\ \texttt{'\textbackslash n'GPA:}\ y$$

     where x is the return of `ToString()` from the base class and y is the value of *GPA* with one decimal place.
   - ☐ an ostream operator that returns its outcome in the same format as `ToString()`.

   Afterward, within the accompanying cpp file, within the main function, declare a *Student* object. Modify its fields and base class fields, and then, display it.

3. Create a header file named `"Employee.h"` and define the class *Employee* within the namespace *PS8*. The class must publicly inherit *Person* and contain

   - ☐ a private double field named *salary*.
   - ☐ a private unsigned int field named *hours*.
   - ☐ a public default constructor that assigns 300 and 5 to *salary* and *hours* respectively.
   - ☐ a public void setter method for *salary* that assigns the parameter to *salary* only if it is at least 100.
   - ☐ a public void setter method for *hours* that assigns the parameter to *hours* only if it is between 1 and 112 inclusively.
   - ☐ a public constant getter method for each field.
   - ☐ a public string constant method named `ToString()` that takes no parameters. It returns a string in the format

     $$x\texttt{"\textbackslash n\$ "}y\texttt{ Hours: }z$$

     where x is the return of `ToString()` from the base class, y is the value of *salary* with two decimal place, and z is the value of *hours*.
   - ☐ an ostream operator that returns its outcome in the same format as `ToString()`.

   Afterward, within the accompanying cpp file, within the main function, declare a *Employee* object. Modify its fields and base class fields, and then, display it.