This lab exercise is an extension of Lab 15a. If you have not completed Lab15a, you should do it now.

Due Date:

You must demonstrate the solution to this lab exercise to the instructor by **Saturday, November 21, 2020**, in order to receive full credit for this work.

Summary of Lab 15a

In Lab 15a, you were given code for the **Employee** class (**`Employee.h`** and **`Employee.cpp`**), and you wrote a class named **ProductionWorker** that is derived from the **Employee** class. That is, the **ProductionWorker** class is a subclass of the **Employee** class.

Programming Assignment: Enhancements to Use Exceptions for Error Reporting

Starting with your solution to Lab 15a, modify the **Employee** and **ProductionWorker** classes to define exception classes and use them in the program to report errors in the user input.

Employee Class

Modify the **Employee** class:

- Add an exception class: **InvalidHireDate**
- Add code to the **Employee** class to check if the hire date **string** object fits the **MM/DD/YYYY** numeric format. One easy way to accomplish this is to use the "square brackets" operator (**`[ ]`**) to access individual characters in the hire date string:

  1. The hire date string should have a length of **`10`**.
  2. The characters at index **`2`** and index **`5`** should be a forward-slash character ('**`/`**').
  3. The characters at index **`0`**, **`1`**, **`3`**, **`4`**, **`6`**, **`7`**, **`8`**, and **`9`** should be in the range of **`0..9`**. (Refer to the **`isdigit()`** function in the **`cctype`** function library . You may need to add a

     ```
     #include <cctype>
     ```

     statement to your program. (Refer to Chapter 10 of the textbook, or the **cplusplus.com** web-site.)

ProductionWorker Class

Modify the **ProductionWorker** class:

- Add two exception classes: **InvalidShift**, and **InvalidPayRate**.
- Add two new test functions:

  `testShift(int shift)` and
  `testPayRate(double rate)`

  These functions should test the validity of the parameter, and throw the appropriate exception if the parameter is incorrect.

- Add a static function for creating a new **ProductionWorker** object:

  ```
  static ProductionWorker *createNewProductionWorker();
  ```

  This function should prompt the user for input of the employee name, hire date, shift, and hourly pay rate, and then dynamically create a **ProductionWorker** object from inside a **`try`** block.

  After the **`try`** block there should be **`catch`** blocks to handle the three types of exceptions: **InvalidHireDate**, **InvalidShift**, and **InvalidPayRate**.

- Add a function to print out the details of a **ProductionWorker** object:

  ```
  void printWorkerData() const;
  ```

Main Program

Modify the "**`c`**" command to call the **`createNewProductionWorker()`** function, and save the address returned in a local variable.

Modify the "**`p`**" command to call the **`printWorkerData`** function**`.`**

Test the program with some valid input values and some invalid values.

Sample Output

In the sample session shown below, the **`bold`** text is what the user entered. In actuality, all text (both input and output) will be displayed in the same font.

Sample Input / Output Session

```
Enter command (or 'h' for help): h
Supported commands:
        c               create a new ProductionWorker object.
        h               print help text.
        p               print ProductionWorker information.
        q               quit (end the program).

Enter command (or 'h' for help): c
Enter name of new employee: George Washington
Enter hire date of new employee: 04/30/1789
Enter shift for new employee (1=day, 2=night): 1
Enter hourly pay rate for new employee: 35.43
Enter command (or 'h' for help): p
Name: George Washington
Employee number: 1
Hire date: 04/30/1789
Shift: Day
Shift number: 1
```

Sample Input / Output Session

```
Pay rate: 35.43
Enter command (or 'h' for help): c
Enter name of new employee: John Adams
Enter hire date of new employee: 3/4/1797
Enter shift for new employee (1=day, 2=night): 1
Enter hourly pay rate for new employee: 50.33
Error: Invalid hire date [3/4/1797]:  Hire date must be MM/DD/YYYY format.
Enter name of new employee: Thomas Jefferson
Enter hire date of new employee: 03/04/1801
Enter shift for new employee (1=day, 2=night): 1
Enter hourly pay rate for new employee: 64.53
Enter command (or 'h' for help): p
Name: Thomas Jefferson
Employee number: 3
Hire date: 03/04/1801
Shift: Day
Shift number: 1
Pay rate: 64.53
Enter command (or 'h' for help): c
Enter name of new employee: James Madison
Enter hire date of new employee: 03/04/1809
Enter shift for new employee (1=day, 2=night): 2
Enter hourly pay rate for new employee: -88.44
Error: Invalid pay rate:  -88.44
Enter name of new employee: James Madison
Enter hire date of new employee: 03/04/1809
Enter shift for new employee (1=day, 2=night): 2
Enter hourly pay rate for new employee: 88.44
Enter command (or 'h' for help): p
Name: James Madison
Employee number: 5
Hire date: 03/04/1809
Shift: Night
Shift number: 2
Pay rate: 88.44
Enter command (or 'h' for help): c
Enter name of new employee: James Monroe
Enter hire date of new employee: 03/04/1817
Enter shift for new employee (1=day, 2=night): 3
Enter hourly pay rate for new employee: 3.44
Error: Invalid shift number:  3
Enter name of new employee: James Monroe
Enter hire date of new employee: 03/04/1817
```

Sample Input / Output Session

```
Enter shift for new employee (1=day, 2=night): 1
Enter hourly pay rate for new employee: 43.44
Enter command (or 'h' for help): p
Name: James Monroe
Employee number: 7
Hire date: 03/04/1817
Shift: Day
Shift number: 1
Pay rate: 43.44
Enter command (or 'h' for help): q
Press any key to continue . . .
```