Week 3:Defining simple classes

Learning Materials: Chapter 6
Topic:
   1. Class Definition.
   2. Access Specifier.
   3. Accessing Members of an object.

[RULEs]: **Member variable is always private. Write member functions and make it public if necessary. Add necessary parameters and return necessary values for each function according to its descriptions. Do not change the name of the function.**

Task 1

Create a class named "**Counter**". An object of Counter class keeps track of count. The object also stores the value of increment steps. For example if the increment step is 5, the **count** will be incremented by **5** each time increment is done. Implement the following member functions (task of the function is written after a hyphen):

   ● **setIncrementStep** - it sets the value of Increment Step in the appropriate member variable. **Do not take input from the user. Use appropriate parameter(s).**
   ● **getCount** - it returns the current count value.
   ● **increment** - it increments the count by increment step for that. For example : if the current count is 4 and increment step value is 2 then executing the function will update the count to 6.
   ● **resetCount** - it resets the value of count to 0.

Task 2

Temperature is a measure of the thermal energy of a system. It can be expressed in Celsius $(°C)$, Fahrenheit $(°F)$, or Kelvin (k) scale. Absolute zero $(0 \ (k))$ is the lowest limit of the thermodynamic temperature scale.

Create a class Temperature that stores the temperature value and the unit. The temperature value can be a fractional value but will not store a value lower than absolute zero. The unit can be either Celsius, Fahrenheit, or Kelvin. Use appropriate storage techniques so that it can only store these units. The conversion formula among the units are

$$(°F) = (°C) * 9/5 + 32;$$

$$(k) = (°C) + 273.15$$

Implement the following member functions and select appropriate data type of the parameters that satisfy the provided functionalities:

- **assign** - this function sets the value to the data members to a particular temperature. Do not allow units other than the provided units, and do not allow a value lower than the provided limit. Show appropriate messages.
- **convert** - this function converts the temperature to the specified target unit and returns the converted value.
- **print**() - this member function will display the temperature in its current form. For example: "**The temperature is 100 Celsius.**"

Task 3

Create a class named "**Time**". An object of **Time** class stores the value of **hour, minute and second.** All the data members need to be declared as **private. It should be noted that 60 seconds a min, 60 mins a hour and 24 hour a day. The hour will be reset to 0 when it is 24.**

Implement the following member functions (the task of the function is written after a hyphen):

- int hours() - return the hour value
- int minutes() - return the minute value
- int seconds() - return the second value
- void reset(int h,int m, int s) - reset the time to given hour min sec
- void advance(int h,int m, int s) - the current time will be advanced by h hour, m min and s sec.
- void print() - print the current time stored.