

דוגמה למחלקה (כולל הפתרון) :

יש לבנות מחלקה המייצגת שעון דיגיטלי . נתוני השעון (שדות) : מספר שעות, מספר דקות ומספר שניות.

יש לספק למחלקה את הפעולות הבאות (שיטות/מתודות/פונקציות) :

1. מתודה המקבלת 3 פרמטרים המהווים את מרכיבי הזמן ומעדכנת בהתאם את שדות האובייקט (יש לבדוק את חוקיות הפרמטרים).

2. מתודה המדפיסה את נתוני השעון בפורמט הבא : HH:MM:SS AM/PM

3. מתודות ממשק לכל שדה פרטי של המחלקה (set/get).

4. מתודה המקדמת ומעדכנת שעון לשנייה אחת קדימה.

הערה: אסור להוסיף שדות נוספים אך מותר להוסיף מתודות.

יש לבנות תכנית ראשית (הדרייבר) ובה ליצור אובייקטים מהמחלקה הנ"ל ולהפעיל את כלל המתודות על מנת לבדוק האם כל אחת מהמתודות מתנהגות כצפוי.

הפתרון :

DigitalClock.h

```
#include <iostream>
#include <iomanip>
using namespace std;
class DigitalClock{
private:
    int hours;      // 0-23
    int minutes;    // 0-59
    int seconds;    //0-59
public:
    void SetDigitalClock(int,int,int);           // set all clock's data
    void ShowClock();                           // show clock's data
    void AddSecond();                           // add one second and update object
    // -----interface methods for each private fields
    // set the clock's parts
    void SetHours(int); // set number of hours
    void SetMinutes(int); // set number of minutes
    void SetSeconds(int); // set number of seconds
    // get the clock's parts - declaration and implementation, because the methods are
    very simple and short
    int GetHours() { return hours; } // get number of hours
    int GetMinutes() { return minutes; } // get number of minutes
    int GetSeconds() { return seconds; } // get number of seconds
    //-----
};
```

DigitalClock.cpp

```
#include "DigitalClock.h"

void DigitalClock::SetDigitalClock(int h,int m,int s)
```

```

{
    SetHours(h);
    SetMinutes(m);
    SetSeconds(s);
}

void DigitalClock::SetHours(int _hours)
{
    if(_hours>=0 && _hours<24)
        hours=_hours;
    else
        hours=0; // or using short if :        hours=(_hours>=0 &&
        _hours<24)?_hours:0;
}
void DigitalClock::SetMinutes(int _minutes)
{
    if(_minutes>=0 && _minutes<60)
        minutes=_minutes;
    else
        minutes=0; // or using short if :        minutes=(_minutes>=0 &&
        _minutes<24)?_minutes:0;
}
void DigitalClock::SetSeconds(int _seconds)
{
    if(_seconds>=0 && _seconds<60)
        seconds=_seconds;
    else
        seconds=0; // or using short if :        seconds=(_seconds>=0 &&
        _seconds<24)?_seconds:0;
}
void DigitalClock::ShowClock()
{
    // Print digital clock in the following format:    HH:MM:SS AM/PM
    // Show hours
    if(hours==0 || hours==12)
        cout<<" 12:";
    else cout<< hours%12<<":";
    // Show minutes
    if(minutes<10)
        cout<<"0";
    cout<<minutes<<":";
    //Show seconds
    if(seconds<10)
        cout<<"0";
    cout<<seconds;
    //Show AM/PM
    if(hours<12)
        cout<< " AM ";
    else
        cout<< " PM ";
    cout<<endl;
    // using manipulators - must add #include <iomanip>
    /*cout<<( ( hours==0 || hours==12) ? 12: hours%12)<<':'
        <<setfill('0')<<setw(2)<<minutes<<':'
        <<setw(2)<<seconds
        <<( hours<12?" AM":" PM")<<endl;*/
}
void DigitalClock::AddSecond()
{
    seconds++;
    if(seconds==60)
    {
        minutes++;
        seconds=0;
    }
}

```

```

        if(minutes==60)
        {
            hours++;
            minutes=0;
        }
        if(hours==24)
            hours=0;
    }

```

driver.cpp

```

#include "DigitalClock.h"
int main()
{
    //driver for DigitalClockClass : create objects and test methods
    DigitalClock clock;
    clock.SetDigitalClock(11,5,60);
    clock.ShowClock();
    clock.SetHours(23);
    clock.ShowClock();
    cout<<"Count of minutes : "<<clock.GetMinutes()<<endl;
    clock.SetDigitalClock(23,59,59);
    clock.AddSecond();
    clock.ShowClock();
    //////////////////////////////////////
}

```