

1- Create a class named Student with attributes name and grade. Include a method to check if the student passed (grade  $\geq$  60) or failed.

أنشئ class باسم "Student" مع اسم السمات والدرجة. قم بتضمين طريقة للتحقق مما إذا كان الطالب قد نجح (الدرجة  $\geq$  60) أو رسب.

## Output

```
Alice passed.
```

## Solution

```
// www.gammal.tech

#include<iostream>
#include<string>
using namespace std;

class Student {
public:
    string name;
    int grade;

    void checkPassFail() {
        if (grade  $\geq$  60) {
            cout << name << " passed." << endl;
        } else {
            cout << name << " failed." << endl;
        }
    }
};

int main() {
    Student student1;
    student1.name = "Alice";
    student1.grade = 75;

    student1.checkPassFail();

    return 0;
}
```

2- Create a class named Calculator with attributes num1 and num2. Include methods to perform basic arithmetic operations (addition, subtraction, multiplication, division).

أنشئ class تسمى Calculator.

with attributes num1 num2

تضمن طرق إجراء العمليات الحسابية الأساسية (الجمع، الطرح، الضرب، القسمة).

Output

```
Sum: 15  
Difference: 5  
Product: 50  
Quotient: 2
```

# Solution

```
// www.gammal.tech

#include<iostream>
using namespace std;

class Calculator {
public:
    double num1;
    double num2;

    double add() {
        return num1 + num2;
    }

    double subtract() {
        return num1 - num2;
    }

    double multiply() {
        return num1 * num2;
    }

    double divide() {
        if (num2 != 0) {
            return num1 / num2;
        } else {
            cout << "Cannot divide by zero." << endl;
            return 0;
        }
    }
};

int main() {
    Calculator calc;
    calc.num1 = 10.0;
    calc.num2 = 5.0;

    cout << "Sum: " << calc.add() << endl;
    cout << "Difference: " << calc.subtract() << endl;
    cout << "Product: " << calc.multiply() << endl;
    cout << "Quotient: " << calc.divide() << endl;

    return 0;
}
```

3- Create a class named Employee with attributes name and salary. Include a method to give an annual raise of 10%.

قم بإنشاء class باسم "Employee" with attributes name and salary. قم بتضمين طريقة لمنح زيادة سنوية قدرها 10%.

## Output

```
Annual raise given. New salary: $55000
```

## Solution

```
// www.gammal.tech

#include<iostream>
#include<string>
using namespace std;

class Employee {
public:
    string name;
    double salary;

    void giveRaise() {
        salary *= 1.10; // 10% raise
        cout << "Annual raise given. New salary: $" << salary << endl;
    }
};

int main() {
    Employee emp;
    emp.name = "John";
    emp.salary = 50000.0;

    emp.giveRaise();

    return 0;
}
```

4- Create a class named Time with attributes hours, minutes, and seconds. Include a method to display the time in HH:MM:SS format.

قم بإنشاء class تسمى "Time" with attributes hours, minutes, and seconds. قم بتضمين طريقة لعرض الوقت بتنسيق HH:MM:SS.

## Output

```
Time: 14:30:45
```

## Solution

```
// www.gammal.tech

#include<iostream>
using namespace std;

class Time {
public:
    int hours;
    int minutes;
    int seconds;

    void displayTime() {
        cout << "Time: " << hours << ":" << minutes << ":" << seconds << endl;
    }
};

int main() {
    Time currentTime;
    currentTime.hours = 14;
    currentTime.minutes = 30;
    currentTime.seconds = 45;

    currentTime.displayTime();

    return 0;
}
```

5- Create a class named Square with an attribute sideLength. Include a method to calculate the perimeter of the square.

قم بإنشاء class باسم Square مع سمة SideLength. تضمين طريقة لحساب محيط المربع.

## Output

```
Perimeter of the square: 24
```

## Solution

```
// www.gammal.tech

#include<iostream>
using namespace std;

class Square {
public:
    double sideLength;

    double calculatePerimeter() {
        return 4 * sideLength;
    }
};

int main() {
    Square square;
    square.sideLength = 6.0;

    cout << "Perimeter of the square: " << square.calculatePerimeter() << endl;

    return 0;
}
```

6- Create a class named Song with attributes title and artist. Include a method to play the song.

قم بإنشاء class باسم أغنية  
with attributes title and artist. قم بتضمين طريقة لتشغيل الأغنية.

## Output

```
Now playing: Shape of You by Ed Sheeran
```

## Solution

```
// www.gammal.tech

#include<iostream>
#include<string>
using namespace std;

class Song {
public:
    string title;
    string artist;

    void play() {
        cout << "Now playing: " << title << " by " << artist << endl;
    }
};

int main() {
    Song mySong;
    mySong.title = "Shape of You";
    mySong.artist = "Ed Sheeran";

    mySong.play();

    return 0;
}
```

7- Create a class named Computer with attributes brand and RAMSize. Include a method to check if the computer meets minimum RAM requirements (e.g., 8GB).

قم بإنشاء class باسم الكمبيوتر with attributes brand and RAMSize  
قم بتضمين طريقة للتحقق مما إذا كان الكمبيوتر يلبي الحد الأدنى من متطلبات  
ذاكرة الوصول العشوائي (على سبيل المثال، 8 جيجابايت).

## Output

```
The computer meets minimum RAM requirements.
```

# Solution

```
// www.gammal.tech

#include<iostream>
#include<string>
using namespace std;

class Computer {
public:
    string brand;
    int RAMSize;

    bool meetsMinimumRAMRequirements() {
        return RAMSize >= 8;
    }
};

int main() {
    Computer myComputer;
    myComputer.brand = "HP";
    myComputer.RAMSize = 12;

    if (myComputer.meetsMinimumRAMRequirements()) {
        cout << "The computer meets minimum RAM requirements." << endl;
    } else {
        cout << "The computer does not meet minimum RAM requirements." << endl;
    }

    return 0;
}
```

8- Create a class named TrafficLight with an attribute currentColor representing the current color of the traffic light. Include a method to change the color to the next one in the sequence (red -> yellow -> green).

قم بإنشاء class باسم TrafficLight باستخدام السمة currentColor التي تمثل اللون الحالي لإشارة المرور. قم بتضمين طريقة لتغيير اللون إلى اللون التالي في التسلسل (أحمر -> أصفر -> أخضر).

## Output

```
Traffic light changed to: yellow
Traffic light changed to: green
Traffic light changed to: red
```



## Solution

```
// www.gammal.tech

#include<iostream>
#include<string>
using namespace std;

class TrafficLight {
public:
    string currentColor;

    void changeColor() {
        if (currentColor == "red") {
            currentColor = "yellow";
        } else if (currentColor == "yellow") {
            currentColor = "green";
        } else {
            currentColor = "red";
        }
        cout << "Traffic light changed to: " << currentColor << endl;
    }
};

int main() {
    TrafficLight myTrafficLight;
    myTrafficLight.currentColor = "red";

    myTrafficLight.changeColor();
    myTrafficLight.changeColor();
    myTrafficLight.changeColor();

    return 0;
}
```

9- Create a program using classes to represent a date (day, month, year). Implement a method setDate in the class to set the date with given parameters and ensure the validity of the date. Also, implement the ++ operator overload to increment the date by one day. Finally, create an object, set a date, print it, increment it, and print again.

إنشاء برنامج باستخدام classes لتمثيل التاريخ (اليوم، الشهر، السنة). قم بتنفيذ طريقة setDate في class لتعيين التاريخ with given parameters المحددة والتأكد من صحة التاريخ. قم أيضاً بتنفيذ التحميل الزائد لعامل التشغيل ++ لزيادة التاريخ بمقدار يوم واحد. أخيراً، قم بإنشاء كائن، وحدد تاريخاً، ثم اطبعه، و قم بزيادته، ثم قم بالطباعة مرة أخرى.

# Output

```
Date is correct
31/12/2022
1/1/2023
```

# Solution

```
// www.gammal.tech

#include<iostream>
using namespace std;

class Gammal_Tech_member {
private:
    int day, month, year;
public:
    bool setDate(int d, int m, int y) {
        if (d >= 1 && d <= 31)
            day = d;
        else
            return false;

        if (m >= 1 && m <= 12)
            month = m;
        else
            return false;

        if (y >= 2020)
            year = y;
        else
            return false;

        return true;
    }

    void operator++() {
        if (day < 31)
            day++;
        else {
            day = 1;
            if (month < 12)
                month++;
            else {
                month = 1;
                year++;
            }
        }
    }

    void print() {
        cout << day << "/" << month << "/" << year << endl;
    }
};

int main() {
    Gammal_Tech_member omar;

    if (omar.setDate(31, 12, 2022) == false)
        cout << "Date is incorrect\n";
    else
        cout << "Date is correct\n";

    omar.print(); // Print the date before increment
    ++omar;       // Increment the date
    omar.print(); // Print the date after increment

    return 0;
}
```

10- create a program to represent a date using classes. Implement methods to set the date, increment the date by one day using the ++ operator, and display the date. Additionally, create a method to check if a given year is a leap year. In the main program, create an object, set a date, increment it, check if the year is a leap year, and print the date.

إنشاء برنامج لتمثيل تاريخ باستخدام **classes**. قم بتنفيذ طرق لتعيين التاريخ وزيادة التاريخ بيوم واحد باستخدام عامل التشغيل ++ وعرض التاريخ. بالإضافة إلى ذلك، قم بإنشاء أسلوب للتحقق مما إذا كانت سنة معينة سنة كبيسة. في البرنامج الرئيسي، قم بإنشاء **object** وقم بتعيين تاريخ، وزيادته، وتحقق مما إذا كانت السنة سنة كبيسة، ثم قم بطباعة التاريخ.

## Input & Output

```
Date is correct
31/12/2022
1/1/2023
Enter a year to check for leap year: 2024
2024 is a leap year.
```

## Solution

```

// www.gammal.tech
#include<iostream>
using namespace std;

class Gammal_Tech_member {
private:
    int day, month, year;

public:
    bool setDate(int d, int m, int y) {
        if (d >= 1 && d <= 31)
            day = d;
        else
            return false;

        if (m >= 1 && m <= 12)
            month = m;
        else
            return false;

        if (y >= 2020)
            year = y;
        else
            return false;

        return true;
    }

    void operator++() {
        if (day < daysInMonth(month, year))
            day++;
        else {
            day = 1;
            if (month < 12)
                month++;
            else {
                month = 1;
                year++;
            }
        }
    }

    void print() {
        cout << day << "/" << month << "/" << year << endl;
    }

    bool isLeapYear(int y) {
        if ((y % 4 == 0 && y % 100 != 0) || (y % 400 == 0))
            return true;
        else
            return false;
    }

    int daysInMonth(int m, int y) {
        int daysInMonth[] = {0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};
        if (m == 2 && isLeapYear(y))
            return 29;
        else
            return daysInMonth[m];
    }
};

int main() {
    Gammal_Tech_member omar;

    if (omar.setDate(31, 12, 2022) == false)
        cout << "Date is incorrect\n";
    else
        cout << "Date is correct\n";

    omar.print(); // Print the date before increment
    ++omar;       // Increment the date
    omar.print(); // Print the date after increment

    int yearToCheck;
    cout << "Enter a year to check for leap year: ";
    cin >> yearToCheck;

    if (omar.isLeapYear(yearToCheck))
        cout << yearToCheck << " is a leap year.\n";
    else
        cout << yearToCheck << " is not a leap year.\n";

    return 0;
}

```