**#Solution-1:**

#include <iostream>

#include <string>

using namespace std;

class GradeBook {

private:

string courseName; // course name for this GradeBook

string instructorName; // course instructor's name

public:

explicit GradeBook(string course, string instructor)

: courseName(course), instructorName(instructor) {}

void setInstructorName(string instructor) {

instructorName = instructor;

}

string getInstructorName() const {

return instructorName;

}

void displayMessage() const {

cout << "Welcome to the grade book for " << getCourseName() << "!" << endl;

cout << "This course is presented by: " << getInstructorName() << endl;

}

string getCourseName() const {

return courseName;

}

};

int main() {

// Create a GradeBook object with course name and instructor name

GradeBook myGradeBook("Introduction to Computer Science", "Prof. Smith");

// Display initial values

cout << "Initial course name is: " << myGradeBook.getCourseName() << endl;

// Change instructor's name

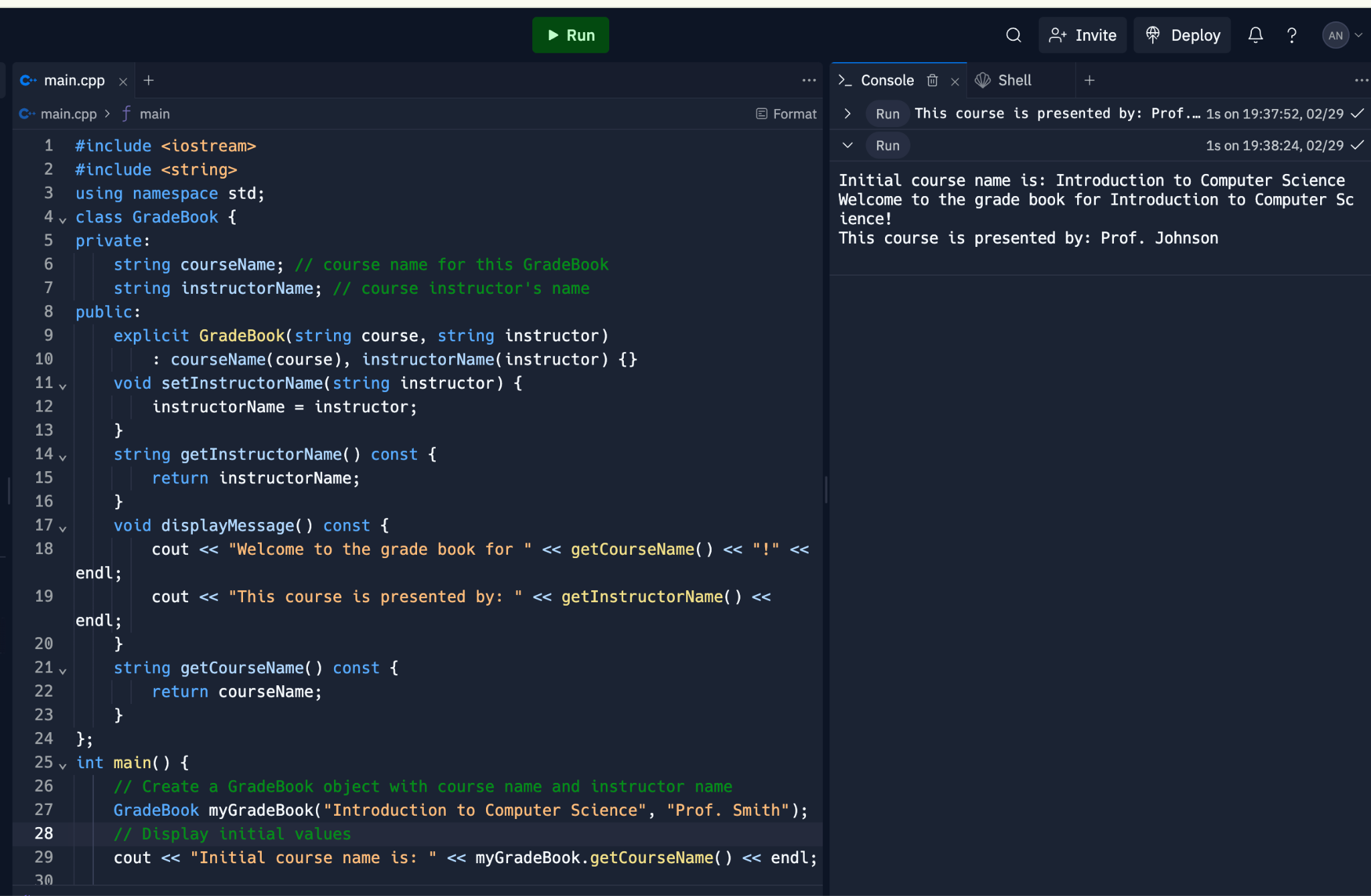
myGradeBook.setInstructorName("Prof. Johnson");

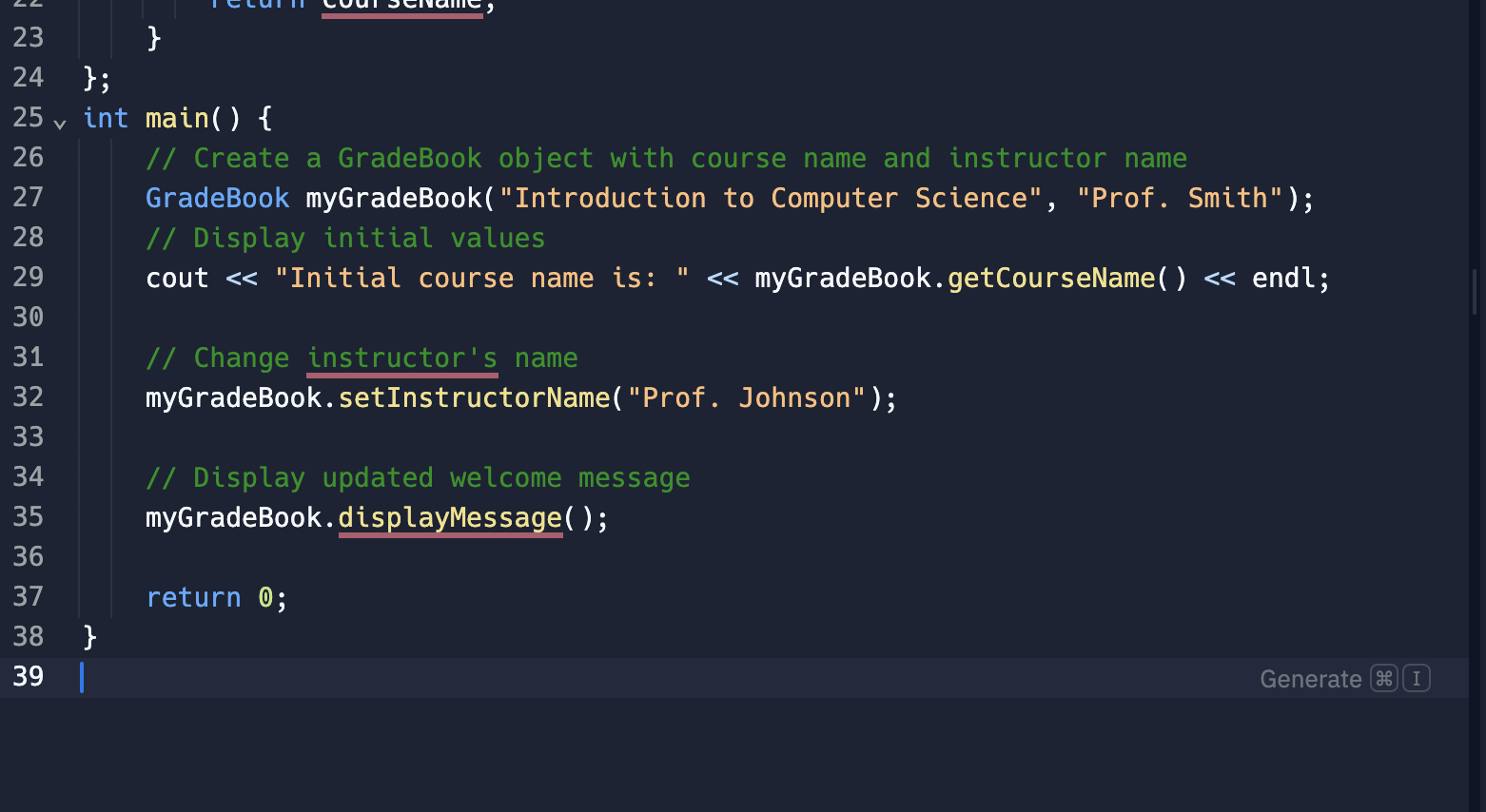
// Display updated welcome message

myGradeBook.displayMessage();

return 0;

}





**#Solution-2:**

#include <iostream>

using namespace std;

class Date {

private:

int month;

int day;

int year;

public:

// Constructor with three parameters

Date(int m, int d, int y) {

// Ensure month value is in the range 1-12

if (m >= 1 && m <= 12)

month = m;

else

month = 1; // Set month to 1 if invalid value provided

day = d;

year = y;

}

void setMonth(int m) {

if (m >= 1 && m <= 12)

month = m;

else

cout << "Invalid month value. Setting month to 1." << endl;

}

void setDay(int d) {

day = d;

}

void setYear(int y) {

year = y;

}

// Get functions for data members

int getMonth() const {

return month;

}

int getDay() const {

return day;

}

int getYear() const {

return year;

}

// Display date in the format: MM/DD/YYYY

void displayDate() const {

cout << month << "/" << day << "/" << year << endl;

}

};

int main() {

Date myDate(2, 15, 2024);

cout << "Initial date: ";

myDate.displayDate();

myDate.setMonth(12);

myDate.setDay(31);

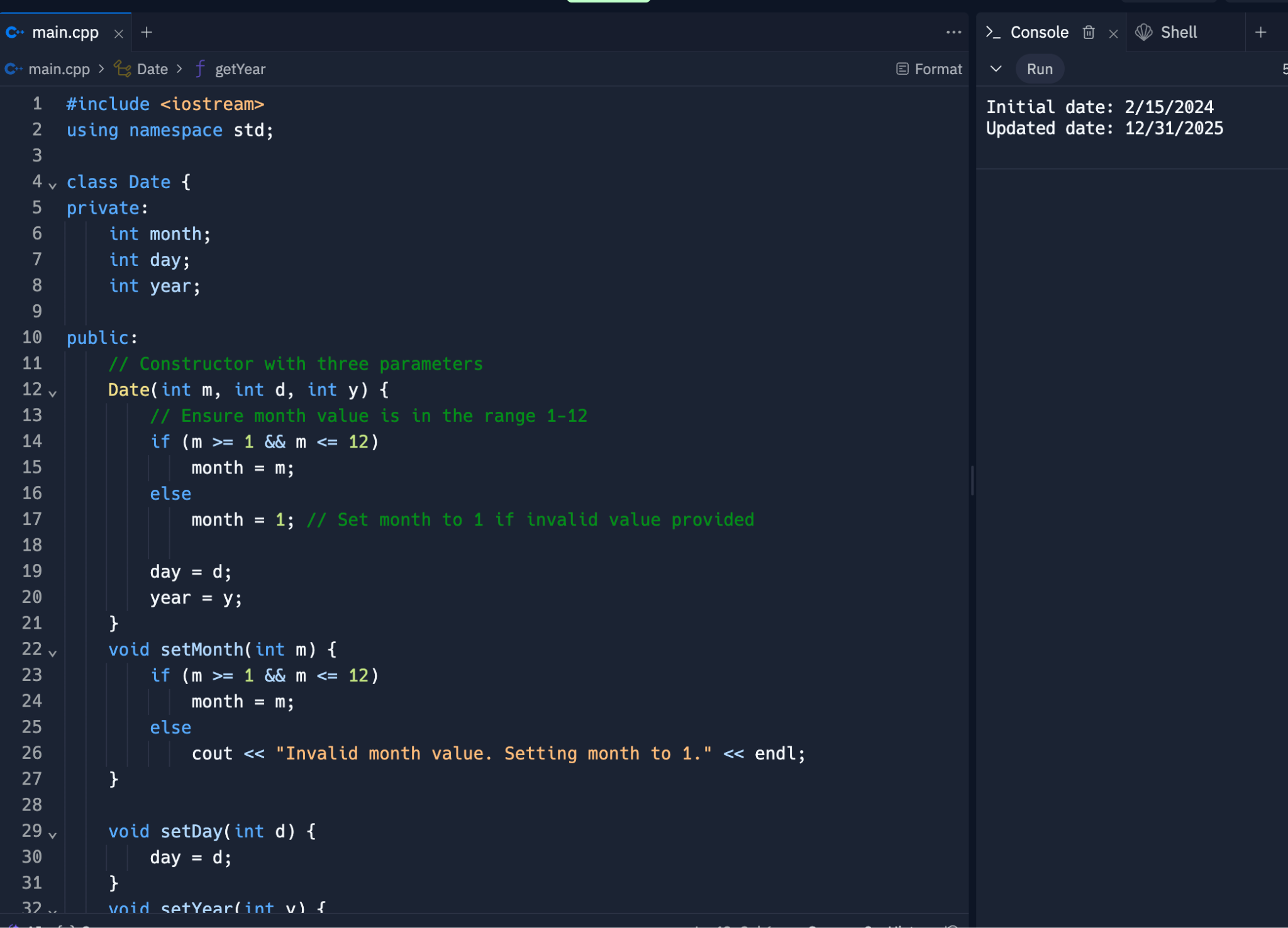
myDate.setYear(2025);

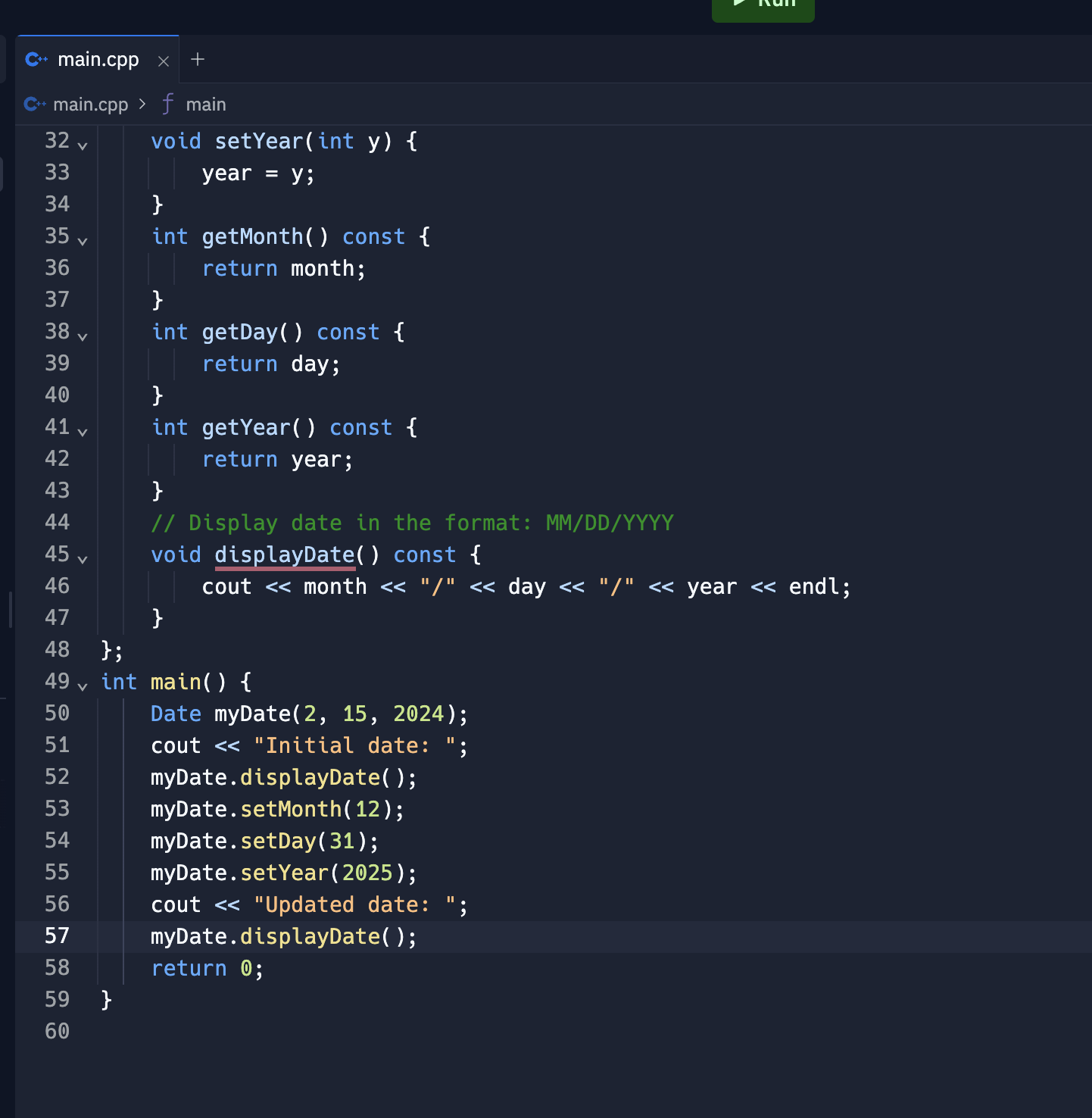
cout << "Updated date: ";

myDate.displayDate();

return 0;

}





**#Solution-3:**

#include <iostream>

#include <string>

using namespace std;

class HeartRates {

private:

string firstName;

string lastName;

int birthMonth;

int birthDay;

int birthYear;

public:

// Constructor with parameters for first name, last name, and birth date

HeartRates(string fName, string lName, int month, int day, int year)

: firstName(fName), lastName(lName), birthMonth(month), birthDay(day), birthYear(year) {}

void setFirstName(string fName) {

firstName = fName;

}

void setLastName(string lName) {

lastName = lName;

}

void setBirthMonth(int month) {

birthMonth = month;

}

void setBirthDay(int day) {

birthDay = day;

}

void setBirthYear(int year) {

birthYear = year;

}

// Get functions for data members

string getFirstName() const {

return firstName;

}

string getLastName() const {

return lastName;

}

int getBirthMonth() const {

return birthMonth;

}

int getBirthDay() const {

return birthDay;

}

int getBirthYear() const {

return birthYear;

}

// Calculate and return person's age in years

int getAge() const {

// Assume current year is 2024 (you can modify this based on actual current year)

int currentYear = 2024;

return currentYear - birthYear;

}

// Calculate and return person's maximum heart rate

int getMaxHeartRate() const {

return 220 - getAge();

}

// Calculate and return person's target heart rate range

string getTargetHeartRate() const {

int maxRate = getMaxHeartRate();

int lowerRange = 0.5 \* maxRate;

int upperRange = 0.85 \* maxRate;

return to\_string(lowerRange) + " BPM - " + to\_string(upperRange) + " BPM";

}

};

int main() {

// Prompt user for information

string fName, lName;

int birthMonth, birthDay, birthYear;

cout << "Enter your first name: ";

cin >> fName;

cout << "Enter your last name: ";

cin >> lName;

cout << "Enter your birth month (1-12): ";

cin >> birthMonth;

cout << "Enter your birth day: ";

cin >> birthDay;

cout << "Enter your birth year: ";

cin >> birthYear;

// Create a HeartRates object

HeartRates person(fName, lName, birthMonth, birthDay, birthYear);

// Display person's information

cout << "\nPerson's Information:" << endl;

cout << "Name: " << person.getFirstName() << " " << person.getLastName() << endl;

cout << "Date of Birth: " << person.getBirthMonth() << "/" << person.getBirthDay() << "/" << person.getBirthYear() << endl;

// Calculate and display age, maximum heart rate, and target heart rate

cout << "\nAge: " << person.getAge() << " years" << endl;

cout << "Maximum Heart Rate: " << person.getMaxHeartRate() << " BPM" << endl;

cout << "Target Heart Rate Range: " << person.getTargetHeartRate() << endl;

return 0;

}

