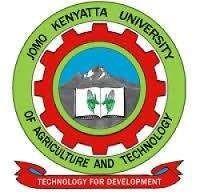
****

**JOMO KENYATTA UNIVERSITY OF AGRICULTURE AND TECHNOLOGY**

**SCHOOL: COLLEE OF ENGINEERING AND TECHNOLOGY**

**DEPARTMENT: ELECTRICAL AND ELECTRONICS ENGINEERING**

**UNIT: COMPUTER PROGRAMMING II**

**UNIT CODE: SMA 2276**

**TITLE: CAT 2**

**LECTURER: MR. BETT**

**SEMESTER: 3.2**

**YEAR: 2023**

**GROUP MEMBERS**

|  |  |
| --- | --- |
| **STUDENT NAME** | **STUDENT REG. NO** |
| **GRACE KARINGA** | **ENE211-0012/2020** |
| **CYNTHIA CHEPKURUI** | **ENE211-0014/2020** |
| **CELINE KARIUKI** | **ENE211-0032/2020** |
| **NICOLE BEATY** | **ENE211-0038/2020** |
| **LENOX NZIA** | **ENE211-0050/2020** |

**No.1**

**INPUT**

#include <iostream>

#include <string>

using namespace std;

// Book class to store book details

class Book {

public:

string author;

float price;

string title;

int book\_number;

int num\_copies;

};

// Library class to maintain inventory

class Library {

protected:

Book books[100];

int num\_books;

public:

// Constructor to initialize library with zero books

Library() {

num\_books = 0;

}

// Add a book to the inventory

void add\_book() {

Book new\_book;

cout << "Enter author: ";

cin >> new\_book.author;

cout << "Enter price: ";

cin >> new\_book.price;

cout << "Enter title: ";

cin >> new\_book.title;

cout << "Enter book number: ";

cin >> new\_book.book\_number;

cout << "Enter number of copies: ";

cin >> new\_book.num\_copies;

books[num\_books] = new\_book;

num\_books++;

}

// Edit a book in the inventory

void edit\_book() {

int book\_num;

cout << "Enter book number to edit: ";

cin >> book\_num;

for (int i = 0; i < num\_books; i++) {

if (books[i].book\_number == book\_num) {

cout << "Enter new author: ";

cin >> books[i].author;

cout << "Enter new price: ";

cin >> books[i].price;

cout << "Enter new title: ";

cin >> books[i].title;

cout << "Enter new number of copies: ";

cin >> books[i].num\_copies;

cout << "Book details updated successfully." << endl;

return;

}

}

cout << "Book not found." << endl;

}

void display\_books() {

if (num\_books == 0) {

cout << "No books in inventory." << endl;

} else {

cout << "Book List:" << endl;

for (int i = 0; i < num\_books; i++) {

cout << "Author: " << books[i].author << endl;

cout << "Price: " << books[i].price << endl;

cout << "Title: " << books[i].title << endl;

cout << "Book Number: " << books[i].book\_number << endl;

cout << "Number of Copies: " << books[i].num\_copies << endl;

cout << endl;

}

}

}

};

class ChiefLibrarian : public Library {

public:

// Display all books in the inventory

void display\_books() {

if (num\_books == 0) {

cout << "No books in inventory." << endl;

} else {

cout << "Book List:" << endl;

for (int i = 0; i < num\_books; i++) {

cout << "Author: " << books[i].author << endl;

cout << "Price: " << books[i].price << endl;

cout << "Title: " << books[i].title << endl;

cout << "Book Number: " << books[i].book\_number << endl;

cout << "Number of Copies: " << books[i].num\_copies << endl;

cout << endl;

}

}

}

};

// Main function

int main() {

Library library;

ChiefLibrarian chiefLibrarian;

int choice, role;

cout << "What's your role: "<< endl;

cout << "1. Librarian" << endl;

cout << "2. ChiefLibrarian" << endl;

cin >> role;

while (true) {

cout << "1. Add Book" << endl;

cout << "2. Edit Book" << endl;

cout << "3. Display Books" << endl;

cout << "4. Quit" << endl;

cout << "Enter your choice: ";

cin >> choice;

if (role == 1) {

switch (choice) {

case 1: cout << "Restricted choice. You do not have access to this method." <<

endl;

break;

case 2: cout << "Restricted choice. You do not have access to this method." <<

endl;

break;

break;

case 3:

library.display\_books();

case 4:

return 0;

default:

cout << "Invalid choice. Please try again." << endl;

}

} else {

switch (choice) {

case 1:

chiefLibrarian.add\_book();

break;

case 2:

chiefLibrarian.edit\_book();

break;

case 3:

chiefLibrarian.display\_books();

break;

case 4:

return 0;

default:

cout << "Invalid choice. Please try again." << endl;

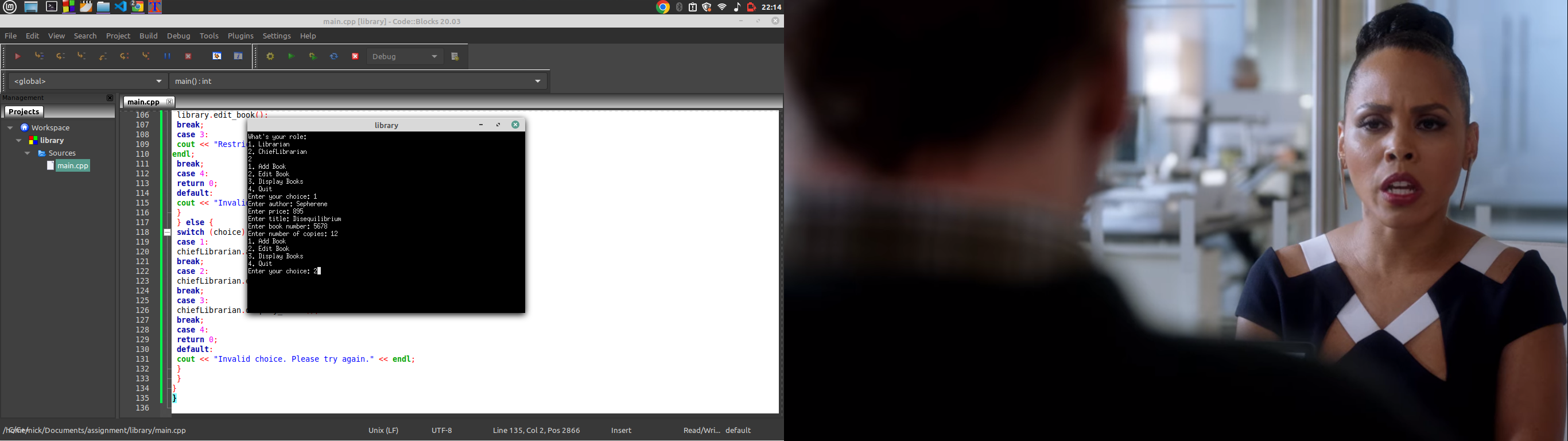
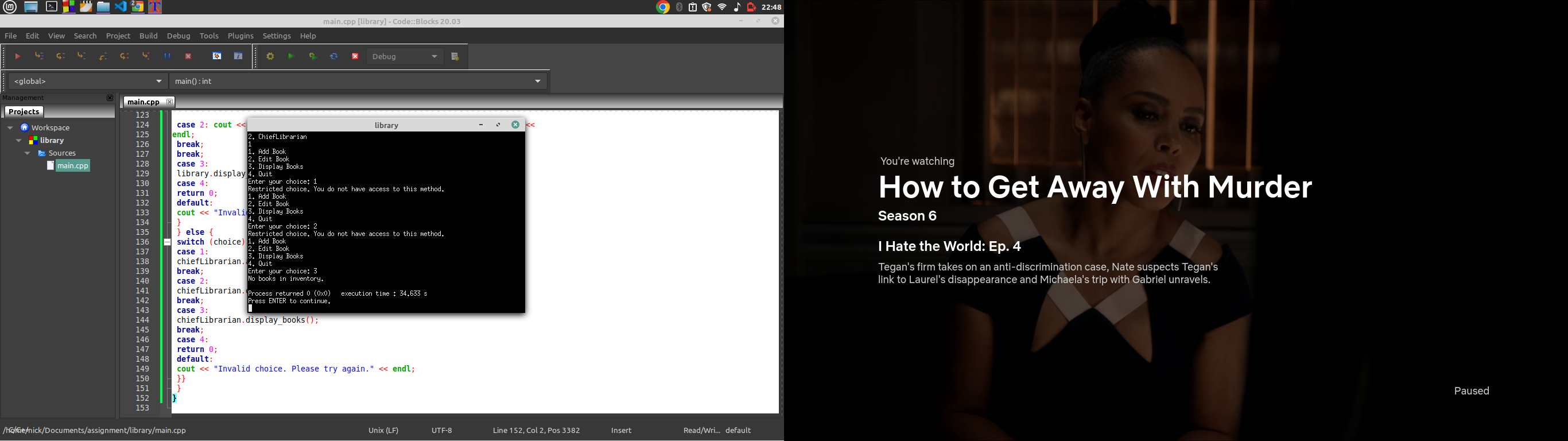
}}

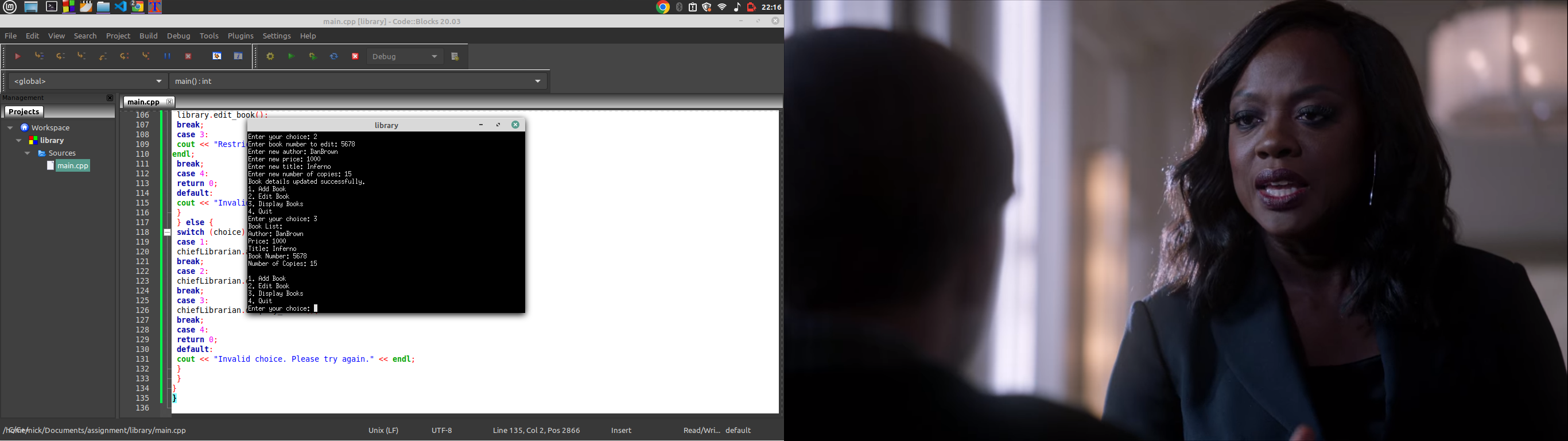
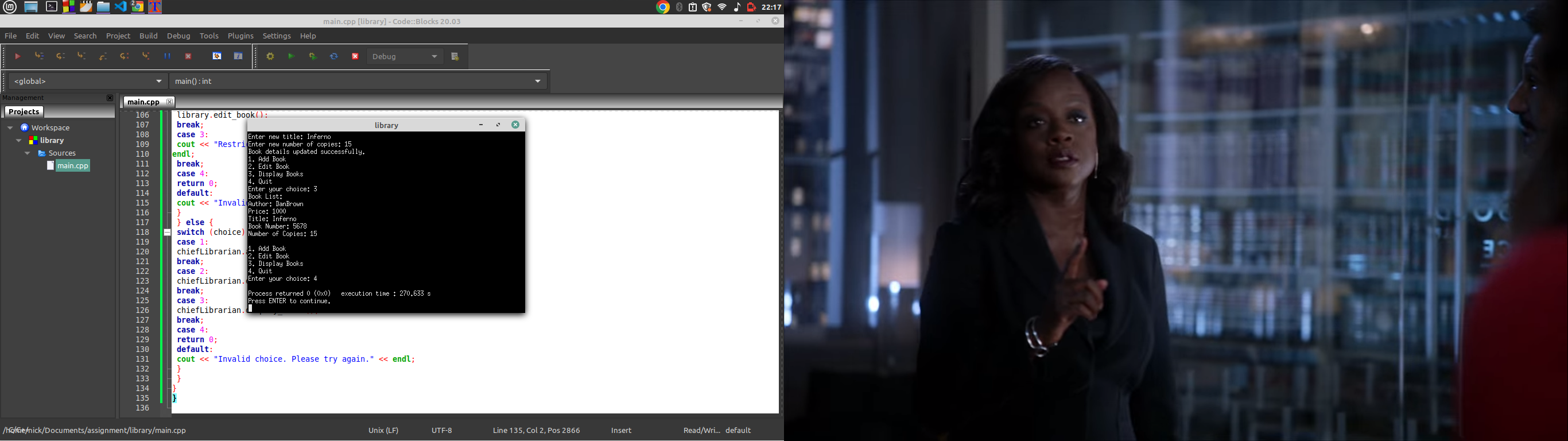
}

}

**OUTPUT**

i)When the librarian is identified as the user





**No.2**

**INPUT**

#include <iostream>

#include <string>

using namespace std;

class Vehicle{

private:

string make;

string model;

string engine\_number;

float sale\_price;

public:

void set\_vehicle(string m, string mod, string en, float sp){

make = m;

model = mod;

engine\_number = en;

sale\_price = sp;

}

float get\_profit(){

float profit = sale\_price \* 0.15;

return profit;

}

};

int main(){

Vehicle v1;

string make, model, engine\_number;

float sale\_price;

cout<<"Enter the make of the vehicle:";

cin>>make;

cout<<"Enter the model of the vehicle:";

cin>>model;

cout<<"Enter the engine\_number of the vehicle:";

cin>>engine\_number;

cout<<"Enter the sale price of the vehicle:";

cin>>sale\_price;

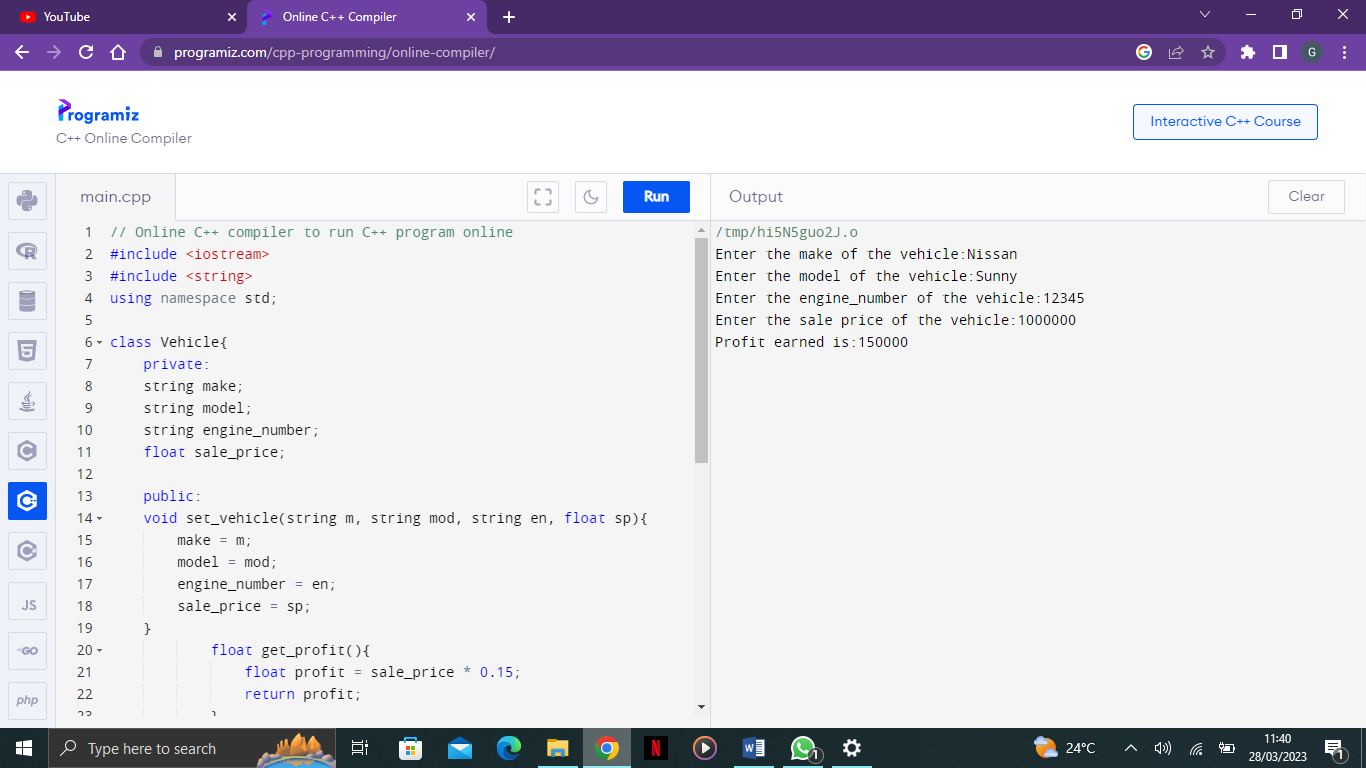
v1.set\_vehicle(make, model, engine\_number,sale\_price);

cout<<"Profit earned is:"<<v1.get\_profit()<<endl;

return 0;

}

**Outputs**



**No.3**

**INPUT**

#include <iostream>

#include <string>

#include <vector>

using namespace std;

class Voter{

private:

string voter\_card\_id;

string national\_id\_number;

string first\_name;

string middle\_name;

string surname;

string polling\_station;

string date\_of\_birth;

string gender;

public:

// Constructor to initialize all member variables

Voter() : voter\_card\_id(""), national\_id\_number(""), first\_name(""), middle\_name(""), surname(""), polling\_station(""), date\_of\_birth(""), gender("") {}

// Destructor to clean up memory if needed

~Voter() {}

// Function to set all member variables to their default values

void reset\_voter() {

voter\_card\_id = "";

national\_id\_number = "";

first\_name = "";

middle\_name = "";

surname = "";

polling\_station = "";

date\_of\_birth = "";

gender = "";

}

// Function to add a new voter

void add\_voter(){

reset\_voter(); // reset all member variables to default values before adding a new voter

cout<<"Enter your voter card ID: ";

cin>>voter\_card\_id;

cout<<"Enter your national ID: ";

cin>>national\_id\_number;

cout<<"Enter your first name: ";

cin>>first\_name;

cout<<"Enter your middle name: ";

cin>>middle\_name;

cout<<"Enter your surname: ";

cin>>surname;

cout<<"Enter polling station: ";

cin>>polling\_station;

cout<<"Enter date of birth (in dd-mm-yyyy format): ";

cin>>date\_of\_birth;

cout<<"Enter your gender: ";

cin>>gender;

}

// Function to display voter details

void display\_voter(){

cout<<"Voter card ID: "<<voter\_card\_id<<endl;

cout<<"National ID number: "<<national\_id\_number<<endl;

cout<<"Name: "<<first\_name<<" "<<middle\_name<<" "<<surname<<endl;

cout<<"Date of birth: "<<date\_of\_birth<<endl;

cout<<"Gender: "<<gender<<endl;

}

};

int main(){

vector<Voter> voters; // create a vector to store all voters

string command; // create a variable to store user commands

// loop to input voters until user chooses to stop

while (true) {

Voter v;

v.add\_voter();

voters.push\_back(v); // add the new voter to the vector

v.display\_voter();

// ask user if they want to continue or stop

cout << "Enter 'exit' to stop adding voters, or any other key to continue: ";

cin >> command;

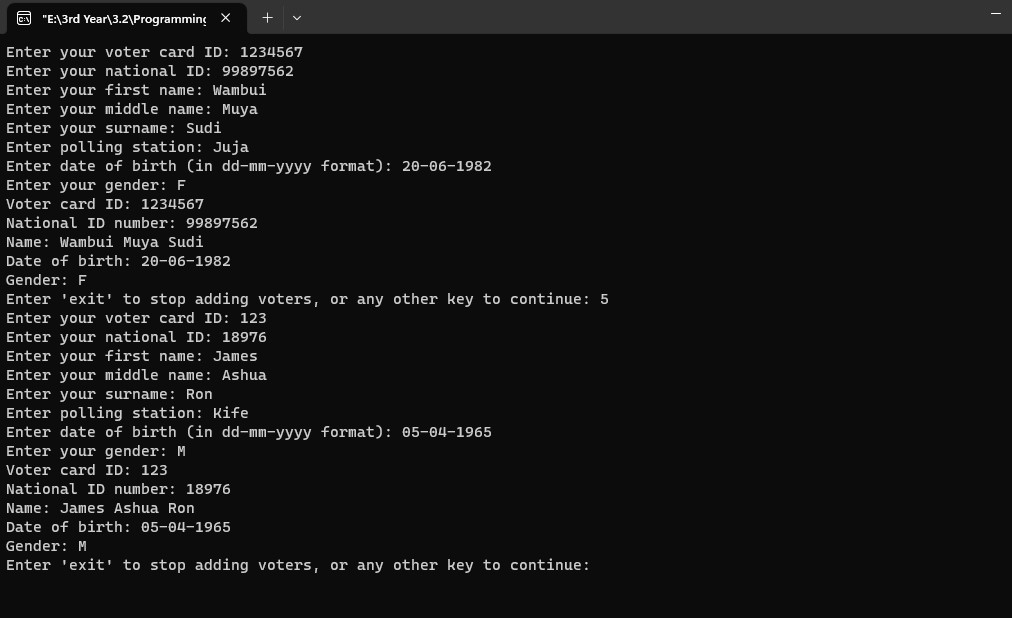
if (command == "exit") {

break; // exit the loop if user types "exit"

}

}

}

**OUTPUT**

**No.4**

**INPUT**

#include<iostream>

#include <string>

using namespace std;

class Subscriber {

private:

string name;

string phone\_number;

double airtime\_amount;

int bonus\_points;

public:

Subscriber(string n, string p, double a){

name=n;

phone\_number=p;

airtime\_amount=a;

bonus\_points=compute\_bonuspoints();

}

int

compute\_bonuspoints(){

if(airtime\_amount>=2000.00)

{

return 500.00;

}

else if(airtime\_amount>=1000.00 && airtime\_amount<2000.00)

{

return 300;

}

else if(airtime\_amount>=500.00 && airtime\_amount< 1000.00)

{

return 100;

}

else if(airtime\_amount>=100.00 && airtime\_amount< 500.00)

{

return 50;

}

else {

return 0;

}

}

void display\_info(){

cout<<name<<":(PHONE NO:"<<phone\_number<<"):AWARDED "<<bonus\_points<<" BONGA POINTS. STAY WITH SAFARICOM. THE BETTER OPTION!"<<endl;

}

};

int main(){

string name,

phone\_number;

double airtime\_amount;

cout<<"Enter Subscriber name:";

getline(cin, name);

cout<<"Enter Phone number:";

getline(cin, phone\_number);

cout<<"Enter Airtime amount:";

cin>>airtime\_amount;

Subscriber

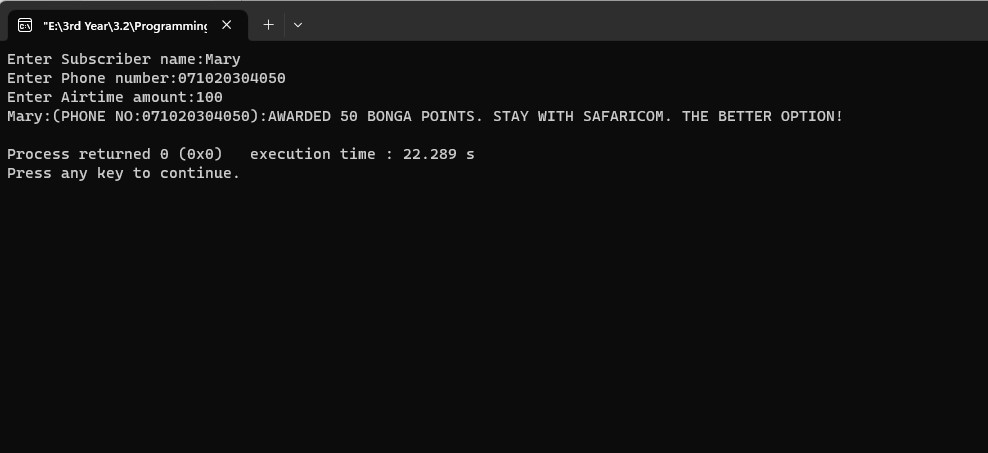
subscriber(name, phone\_number,airtime\_amount);

subscriber.display\_info();

return 0;

}

**OUTPUT**

****

**No.5**

**INPUT**

#include <iostream>

#include <string>

using namespace std;

class employee {

private:

int employee\_id;

string first\_name;

string second\_name;

string surname;

string gender;

string dob;

float monthly\_salary;

public:

employee(int id, string fname, string sname, string surn, string g, string birthdate, float salary)

: employee\_id(id), first\_name(fname), second\_name(sname), surname(surn), gender(g), dob(birthdate), monthly\_salary(salary) {}

void show\_employee() {

cout << "Employee ID: " << employee\_id << endl;

cout << "Name: " << first\_name << " " << second\_name << " " << surname << endl;

cout << "Gender: " << gender << endl;

cout << "Date of Birth: " << dob << endl;

cout << "Monthly Basic Salary: $" << monthly\_salary << endl;

}

friend float compute\_pension(const employee& emp);

};

float compute\_pension(const employee& emp) {

float pension\_contribution = 0.05 \* emp.monthly\_salary;

cout << "Pension Contribution: $" << pension\_contribution << endl;

return pension\_contribution;

}

int main() {

// Prompts the user to enter employee details

int employee\_id;

string first\_name, second\_name, surname, dob\_str;

string gender;

float monthly\_salary;

cout << "Enter Employee ID: ";

cin >> employee\_id;

cout << "Enter First Name: ";

cin >> first\_name;

cout << "Enter Second Name: ";

cin >> second\_name;

cout << "Enter Surname: ";

cin >> surname;

cout << "Enter Gender (Male/Female): ";

cin >> gender;

cout << "Enter Date of Birth (dd-mm-yyyy): ";

cin >> dob\_str;

cout << "Enter Monthly Basic Salary: $";

cin >> monthly\_salary;

// Create an employee object

cout << endl<<" THE EMPLOYEE DETAILS ARE: "<< endl<<endl;

employee emp\_obj(employee\_id, first\_name, second\_name, surname, gender, dob\_str, monthly\_salary);

// Display employee information

emp\_obj.show\_employee();

// Compute pension contribution

compute\_pension(emp\_obj);

return 0;

}

**OUTPUT**

