**DSA Project : Resturant Management System**

**Code:**

#include<iostream>

using namespace std;

class Cuisine{

private:

string name;

double price;

Cuisine\* next;

Cuisine\* previous;

public:

Cuisine(string name , double price)

{

this->name = name;

this->price = price;

this->next = NULL;

this->previous = NULL;

}

void setName(string name)

{

this->name = name;

}

string getName()

{

return name;

}

void setPrice(double price)

{

this->price = price;

}

double getPrice()

{

return price;

}

void setNext( Cuisine\* next)

{

this->next = next;

}

Cuisine\* getNext()

{

return next;

}

void setPrevious( Cuisine\* previous)

{

this->previous = previous;

}

Cuisine\* getPrevious()

{

return previous;

}

};

class Cart{

private:

Cuisine\* item;

int quantity;

Cart\* next;

public:

Cart(Cuisine\* item,int quantity)

{

this->item = item;

this->quantity = quantity;

this->next = NULL;

}

void setItem(Cuisine\* item)

{

this->item = item;

}

Cuisine\* getItem()

{

return item;

}

void setQuantity(int quantity)

{

this->quantity = quantity;

}

int getQuantity()

{

return quantity;

}

void setNext(Cart\* next)

{

this->next = next;

}

Cart\* getNext()

{

return next;

}

};

class Resturant\_Management\_System{

private:

Cuisine\* head;

Cart\* cartHead;

double total;

public:

Resturant\_Management\_System()

{

this->head = NULL;

this->cartHead = NULL;

this->total = 0.0;

}

// add at head

void add\_item\_at\_head()

{

string name;

double price;

cout<<"Enter the name of the item : ";

cin>>name;

cout<<"Enter price of the item : ";

cin>>price;

Cuisine \* new\_Cuisine = new Cuisine(name,price);

new\_Cuisine->setNext(head);

if(head != NULL)

{

head->setPrevious(new\_Cuisine);

}

head = new\_Cuisine;

}

// add at tail

void add\_item\_at\_tail()

{

string name;

double price;

cout<<"Enter the name of the item : ";

cin>>name;

cout<<"Enter price of the item : ";

cin>>price;

Cuisine \* new\_Cuisine = new Cuisine(name,price);

if(head == NULL)

{

head = new\_Cuisine;

return;

}

Cuisine\* temp = head;

while(temp->getNext() != NULL)

{

temp = temp->getNext();

}

temp->setNext(new\_Cuisine);

new\_Cuisine->setPrevious(temp);

}

// delete function

void delete\_item()

{

string name;

cout<<"Enter the name of the item you want to remove : ";

cin>>name;

if(head == NULL)

{

cout<<"No items available \n";

return;

}

Cuisine\* temp = head;

while(temp != NULL && temp->getName() != name)

{

temp = temp->getNext();

}

// if item not found

if(temp == NULL)

{

cout<<" Item not found \n";

return;

}

// if item is at head

if(temp == head)

{

head = temp->getNext();

if(head != NULL)

{

head->setPrevious(NULL);

}

delete temp;

return;

}

// if item is at middle or at tail

if(temp->getPrevious() != NULL)

{

temp->getPrevious()->setNext(temp->getNext());

}

else if(temp->getNext() != NULL)

{

temp->getNext()->setPrevious(temp->getPrevious());

}

delete temp;

}

// updates price function

void update\_price()

{

if(head == NULL)

{

cout<<"No items available \n";

return;

}

string name;

double new\_price;

cout << "Enter the name of the item to update the price : ";

cin >> name;

Cuisine\* temp = head;

while (temp != NULL && temp->getName() != name)

{

temp = temp->getNext();

}

if(temp == NULL)

{

cout<<" Item not found \n";

return;

}

else

{

cout << "Enter the new price: ";

cin >> new\_price;

temp->setPrice(new\_price);

cout << "Price updated successfully!\n";

}

}

// view items or we can say displaying function

void view\_items()

{

if(head == NULL)

{

cout<<"No items available \n";

return;

}

Cuisine\* temp = head;

cout << "\n--- Menu Items ---\n";

while (temp != NULL)

{

cout << "Name: " << temp->getName() << ", Price: Rs"

<< temp->getPrice() << endl;

temp = temp->getNext();

}

}

void add\_to\_cart()

{

if (head == NULL)

{

cout << "No items available to add to the cart.\n";

return;

}

string name;

int quantity;

cout << "Enter the name of the item to add to cart: ";

cin >> name;

cout << "Enter the quantity: ";

cin >> quantity;

Cuisine\* temp = head;

while (temp != NULL && temp->getName() != name)

{

temp = temp->getNext();

}

if (temp == NULL) {

cout << "Item not found.\n";

return;

}

Cart\* newCartItem = new Cart(temp, quantity);

newCartItem->setNext(cartHead);

cartHead = newCartItem;

total += temp->getPrice() \* quantity;

}

void view\_cart()

{

if (cartHead == NULL)

{

cout << "Your cart is empty.\n";

return;

}

Cart\* temp = cartHead;

cout << "\n--- Cart Items ---\n";

while (temp != NULL)

{

cout << "Name: " << temp->getItem()->getName()

<< ", Quantity: " << temp->getQuantity()

<< ", Price: Rs" << temp->getItem()->getPrice() \* temp->getQuantity() << endl;

temp = temp->getNext();

}

cout << "Total Price: Rs" << total << endl;

}

void admin\_menu()

{

int choice;

do{

cout<<" Admin Menu \n";

cout<<"1) Add Item \n";

cout<<"2) Update Price \n";

cout<<"3) Delete Item \n";

cout<<"4) View Items \n";

cout<<"5) Exit to Main menu \n";

cout<<"Enter a choice : ";

cin>>choice;

switch(choice)

{

case 1:

add\_item\_at\_tail();

break;

case 2:

update\_price();

break;

case 3:

delete\_item();

break;

case 4:

view\_items();

break;

case 5:

cout << "Exiting the Admin menu .\n";

break;

default:

cout << "Invalid choice. Please try again.\n";

}

}while(choice != 5);

}

void customer\_menu()

{

int choice;

do{

cout<<" Customer Menu \n";

cout<<"1) View Items \n";

cout<<"2) Add to Cart \n";

cout<<"3) View Cart \n";

cout<<"4) Exit to Main menu \n";

cout<<"Enter a choice : ";

cin>>choice;

switch(choice)

{

case 1:

view\_items();

break;

case 2:

add\_to\_cart();

break;

case 3:

view\_cart();

break;

case 4:

cout << "Exiting the Customer menu .\n";

break;

default:

cout << "Invalid choice. Please try again.\n";

}

}while(choice != 4);

}

// LogIn for admin

void LogIn()

{

string storedUsername = "123";

string storedPassword = "1234";

string username;

string password;

cout << "Enter username: ";

cin >> username;

if (username == storedUsername)

{

cout << "Enter password: ";

cin >> password;

if (password == storedPassword)

{

cout << "Login successful!" << endl;

admin\_menu();

}

else

{

cout << "Incorrect password." << endl;

}

}

else

{

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

}

}

};

int main()

{

Resturant\_Management\_System r;

int choice;

do{

cout<<" Resturant Management System \n";

cout<<"1) Admin \n";

cout<<"2) Customer \n";

cout<<"3) Exit \n";

cout<<"Enter a choice : ";

cin>>choice;

switch(choice)

{

case 1:

r.LogIn();

break;

case 2:

r.customer\_menu();

break;

case 3:

cout << "Exiting the system. Goodbye!\n";

break;

default:

cout << "Invalid choice. Please try again.\n";

}

}while(choice != 3);

}