Game Seller Shop Project

Semester Project

C++ Console Application

Interface

```
struct Inventory    //Struct for storing games
{
    string NAME;
    double PRICE;
};

Inventory game[50]={
    {"Black Myth: Wukong", 50.1},
        {"Monster Hunter Wilds", 45.9},
        {"HELlDIVERS 2", 80.0},
        {"GTA V", 96.7},
        {"Rust", 74.3}
};


struct Cart          //Struct for storing user selected games on chechkout
{
    string name;
    double price;
};

Cart cart[50];

void updateDiscount()···


int showAdminFunctions()···


int showClientFunctions()···


void addGames()···


void viewGames()···


void deleteGames()···


void addToCart()···
```

```
====================================
Choose an option:
1.Admin    2.Customer    3.Exit
====================================
1
Enter password:admin123
Admin have successfully logged in.


====================================
              Admin
====================================
Select from the following:
1. Update games and prices
2. Delete games and prices
3. Views Games
4. Back to Main Menu
5. Update discount
You chose: 3



============================================
Serial No.      NAME                  PRICE
============================================
0               Black Myth: Wukong    50.1
--------------------------------------------
1               Monster Hunter Wilds  45.9
--------------------------------------------
2               HELlDIVERS 2          80
--------------------------------------------
3               GTA V                 96.7
--------------------------------------------
```

Project Overview

- Console-based Game Seller Shop
- Developed using C++
- Supports Admin and Customer roles
- Manages games, cart, and checkout system

Key Features

- Admin login & Customer login
- Game inventory management
- Add / Delete / View games
- Shopping cart & checkout system
- Automatic discount on purchases above $100

Data Structures Used

- Struct Inventory → stores game name & price
- Struct Cart → stores selected games
- Arrays used for inventory & cart management
- Global counters to track games and cart items

Important Functions

- addGames() → Add new games to inventory
- deleteGames() → Remove games from inventory
- viewGames() → Display available games
- addToCart() → Add selected games to cart
- checkout() → Generate receipt & apply discount

Conclusion & Learning Outcomes

- Demonstrates use of structures and arrays
- Implements menu-driven programming
- Uses conditional logic & loops effectively
- Good example of real-world shop simulation in C++