KALPANA N 2347229

VIKRAM VINODH 2347263

TUSHAR SAKHUJA 2347262

CODE AND OUTPUT

STOCK TRADING SIMulator

**CODE :-**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define MAX\_STOCKS 10

#define MAX\_USERS 10

#define MAX\_NAME\_LENGTH 50

struct Stock {

    char symbol[10];

    char name[MAX\_NAME\_LENGTH];

    double price;

};

struct User {

    char username[MAX\_NAME\_LENGTH];

    double balance;

    struct Stock stocks[MAX\_STOCKS];

    int stockQuantities[MAX\_STOCKS]; // Keep track of stock quantities owned

    int stockCount;

};

struct Stock stocks[MAX\_STOCKS];

struct User users[MAX\_USERS];

int userCount = 0;

int stockCount = 0;

void createStock(char symbol[], char name[], double price) {

    strcpy(stocks[stockCount].symbol, symbol);

    strcpy(stocks[stockCount].name, name);

    stocks[stockCount].price = price;

    stockCount++;

}

void createUser(char username[], double balance) {

    strcpy(users[userCount].username, username);

    users[userCount].balance = balance;

    users[userCount].stockCount = 0;

    userCount++;

}

int findUserIndex(char username[]) {

    int i;

    for (i = 0; i < userCount; i++) {

        if (strcmp(users[i].username, username) == 0) {

            return i;

        }

    }

    return -1;

}

int findStockIndex(char symbol[]) {

    int i;

    for (i = 0; i < stockCount; i++) {

        if (strcmp(stocks[i].symbol, symbol) == 0) {

            return i;

        }

    }

    return -1;

}

void buyStock(int userIndex, int stockIndex, int quantity) {

    int i;

     int userStockIndex;

    double totalCost = stocks[stockIndex].price \* quantity;

    if (users[userIndex].balance >= totalCost) {

        users[userIndex].balance -= totalCost;

    userStockIndex = -1;

        for (i = 0; i < users[userIndex].stockCount; i++) {

            if (strcmp(users[userIndex].stocks[i].symbol, stocks[stockIndex].symbol) == 0) {

                userStockIndex = i;

                break;

            }

        }

        if (userStockIndex != -1) {

            users[userIndex].stockQuantities[userStockIndex] += quantity;

        } else {

            userStockIndex = users[userIndex].stockCount;

            users[userIndex].stocks[userStockIndex] = stocks[stockIndex];

            users[userIndex].stockQuantities[userStockIndex] = quantity;

            users[userIndex].stockCount++;

        }

        printf("Successfully bought %d shares of %s\n", quantity, stocks[stockIndex].symbol);

    } else {

        printf("Insufficient balance.\n");

    }

}

void sellStock(int userIndex, int stockIndex, int quantity) {

    int userStockIndex = -1;

    int i;

    for (i = 0; i < users[userIndex].stockCount; i++) {

        if (strcmp(users[userIndex].stocks[i].symbol, stocks[stockIndex].symbol) == 0) {

            userStockIndex = i;

            break;

        }

    }

    if (userStockIndex != -1 && quantity <= users[userIndex].stockQuantities[userStockIndex]) {

        double totalEarnings = stocks[stockIndex].price \* quantity;

        users[userIndex].balance += totalEarnings;

        users[userIndex].stockQuantities[userStockIndex] -= quantity;

        if (users[userIndex].stockQuantities[userStockIndex] == 0) {

            // If all stocks are sold, remove the stock from the user's portfolio

            for (i = userStockIndex; i < users[userIndex].stockCount - 1; i++) {

                users[userIndex].stocks[i] = users[userIndex].stocks[i + 1];

                users[userIndex].stockQuantities[i] = users[userIndex].stockQuantities[i + 1];

            }

            users[userIndex].stockCount--;

        }

        printf("Successfully sold %d shares of %s\n", quantity, stocks[stockIndex].symbol);

    } else {

        printf("Invalid quantity or insufficient shares.\n");

    }

}

void printPortfolio(int userIndex) {

    int i;

    printf("Portfolio for %s:\n", users[userIndex].username);

    printf("Balance: %.2f\n", users[userIndex].balance);

    printf("Stocks:\n");

    for (i = 0; i < users[userIndex].stockCount; i++) {

        printf("%s (%s): %.2f per share\n",

               users[userIndex].stocks[i].name,

               users[userIndex].stocks[i].symbol,

               users[userIndex].stocks[i].price);

        printf("Owned quantity: %d\n", users[userIndex].stockQuantities[i]);

    }

}

int main() {

    createStock("APPLE", "Apple Inc.", 150.00);

    createStock("GOOGLE", "Alphabet Inc.", 2500.00);

    createStock("WIPRO", "Wipro Inc.", 175.00);

    createStock("INFOSYS", "Infosys Inc.", 1500.00);

    createUser("Tushar", 10000.00);

    createUser("Kalpana", 15000.00);

    createUser("Vikram", 12000.00);

    createUser("Jerry", 17000.00);

    while (1) {

        int i;

        int userIndex;

        int stockIndex;

        int choice;

        printf("\nAvailable stocks:\n");

        for (i = 0; i < stockCount; i++) {

            printf("%s (%s): %.2f per share\n", stocks[i].name, stocks[i].symbol, stocks[i].price);

        }

        printf("\n1. Buy\n2. Sell\n3. View Portfolio\n4. Exit\n");

        scanf("%d", &choice);

        if (choice == 1) {

            char username[MAX\_NAME\_LENGTH];

            char symbol[10];

            int quantity;

            printf("Enter your username: ");

            scanf("%s", username);

            userIndex = findUserIndex(username);

            if (userIndex == -1) {

                printf("User not found.\n");

                continue;

            }

            printf("Enter stock symbol: ");

            scanf("%s", symbol);

            stockIndex = findStockIndex(symbol);

            if (stockIndex == -1) {

                printf("Stock not found.\n");

                continue;

            }

            printf("Enter quantity: ");

            scanf("%d", &quantity);

            buyStock(userIndex, stockIndex, quantity);

        } else if (choice == 2) {

            char username[MAX\_NAME\_LENGTH];

            char symbol[10];

            int quantity;

            printf("Enter your username: ");

            scanf("%s", username);

            userIndex = findUserIndex(username);

            if (userIndex == -1) {

                printf("User not found.\n");

                continue;

            }

            printf("Enter stock symbol: ");

            scanf("%s", symbol);

            stockIndex = findStockIndex(symbol);

            if (stockIndex == -1) {

                printf("Stock not found.\n");

                continue;

            }

            printf("Enter quantity: ");

            scanf("%d", &quantity);

            sellStock(userIndex, stockIndex, quantity);

        } else if (choice == 3) {

            char username[MAX\_NAME\_LENGTH];

            printf("Enter your username: ");

            scanf("%s", username);

            userIndex = findUserIndex(username);

            if (userIndex == -1) {

                printf("User not found.\n");

                continue;

            }

            printPortfolio(userIndex);

        } else if (choice == 4) {

            break;

        } else {

            printf("Invalid choice.\n");

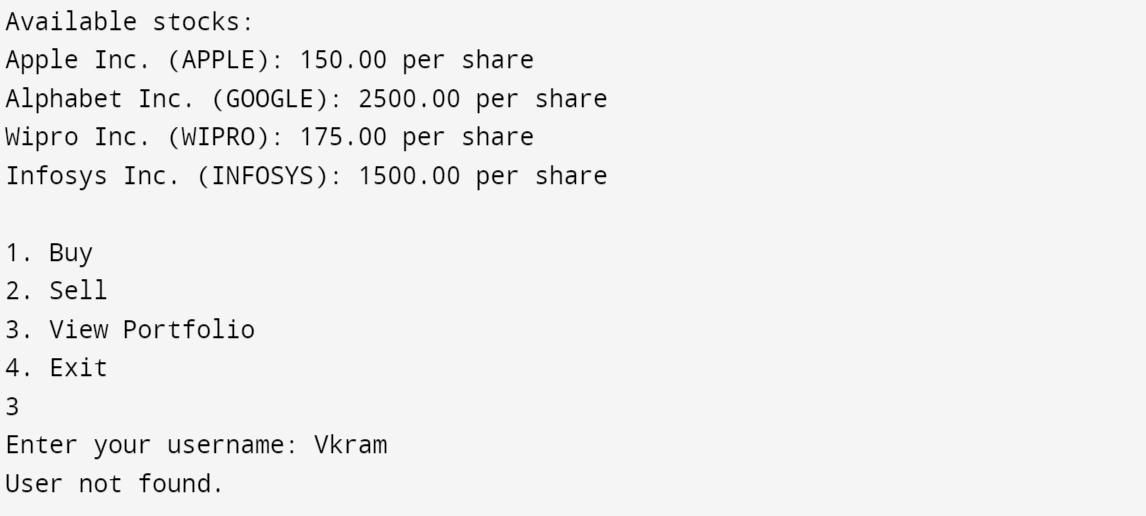
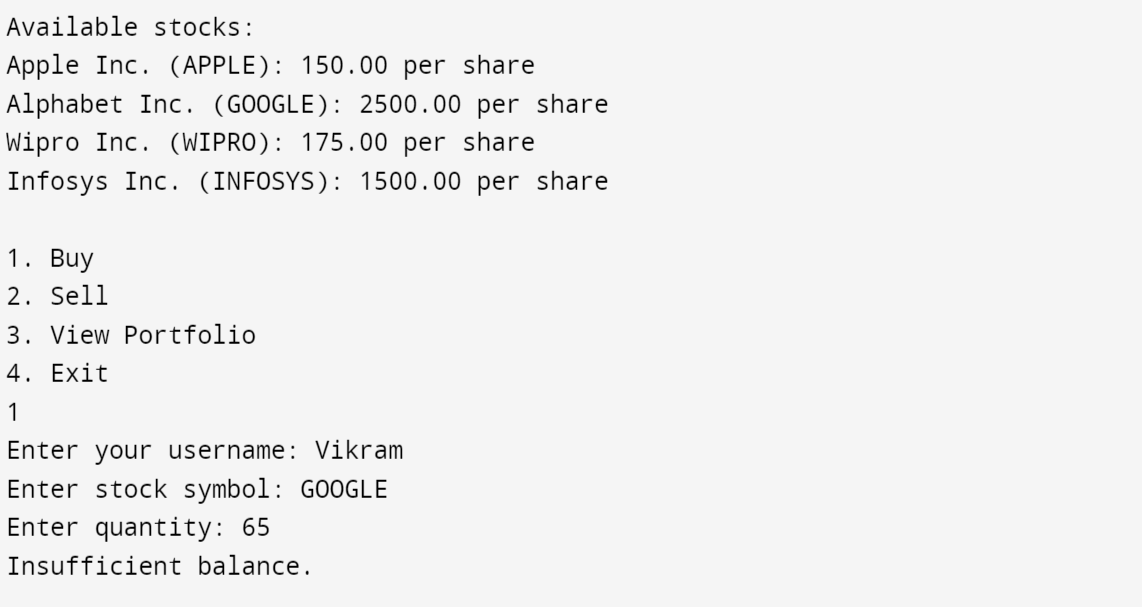
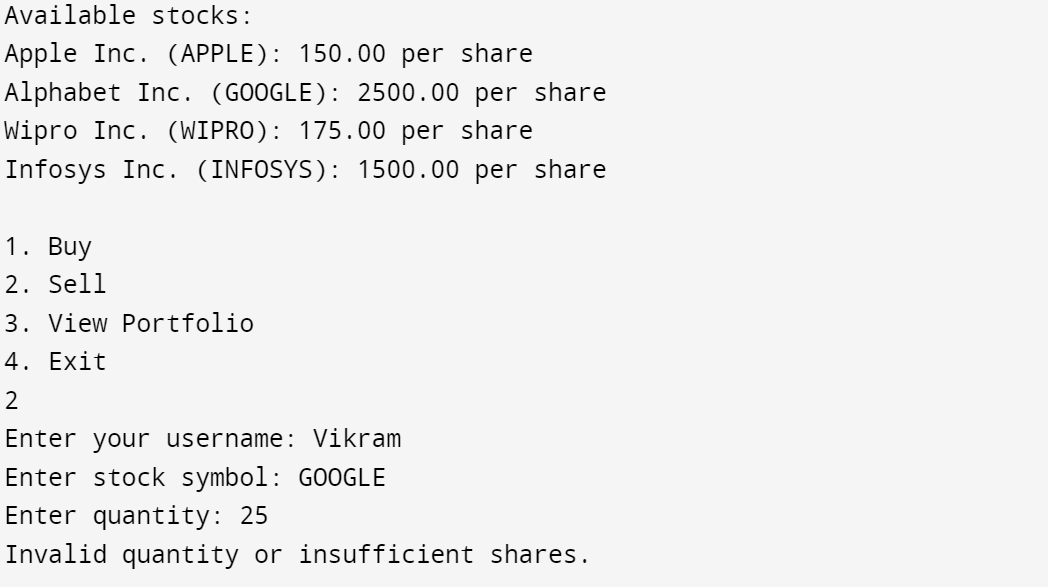
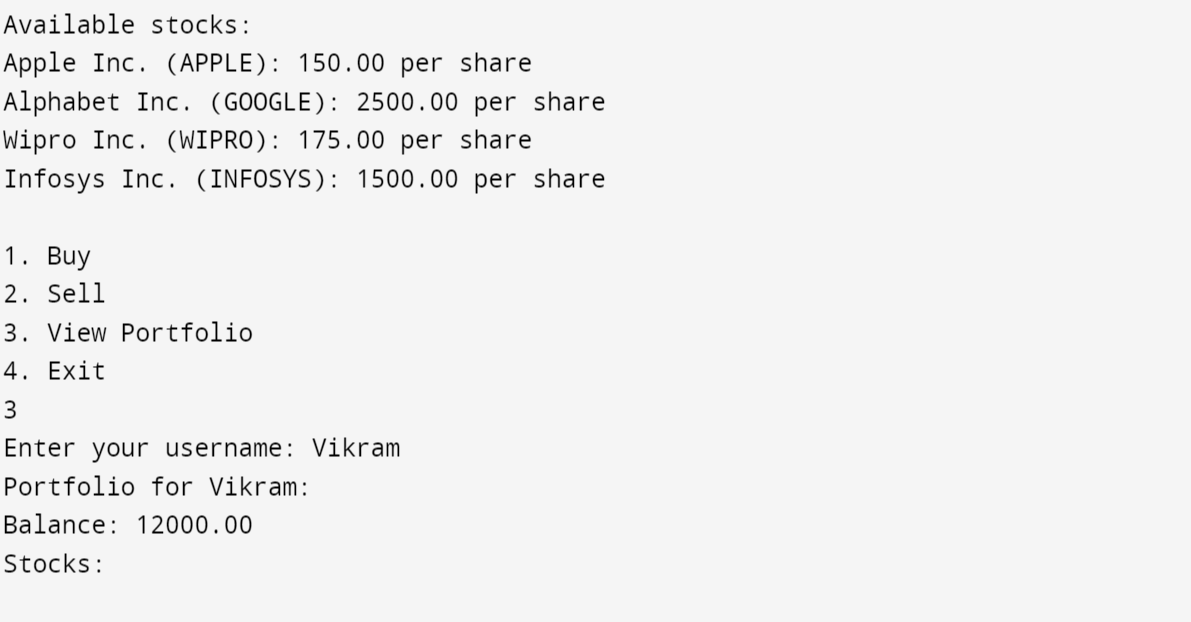
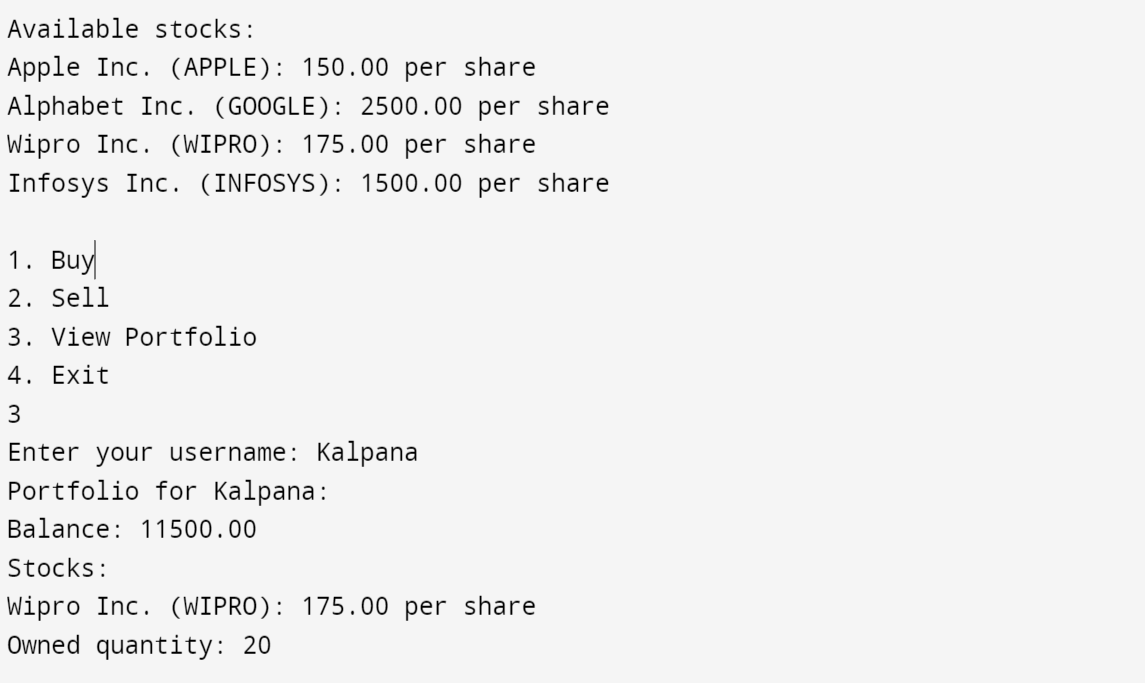
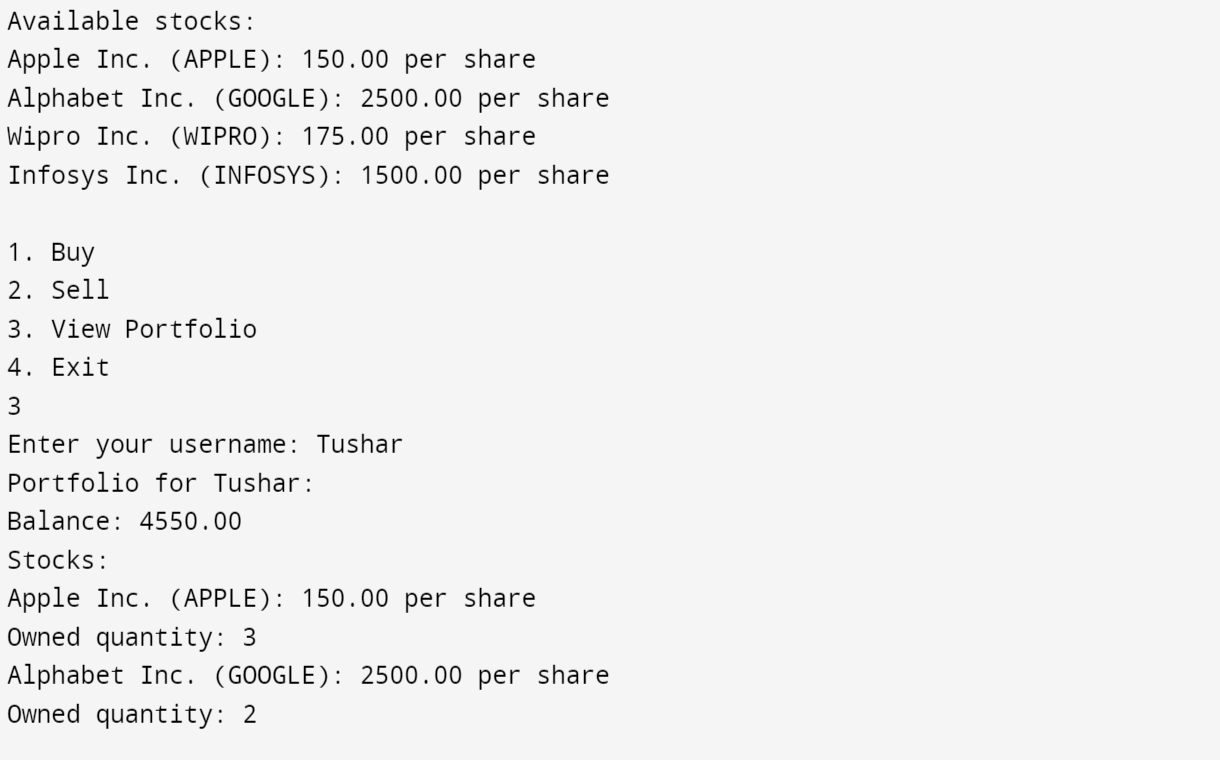
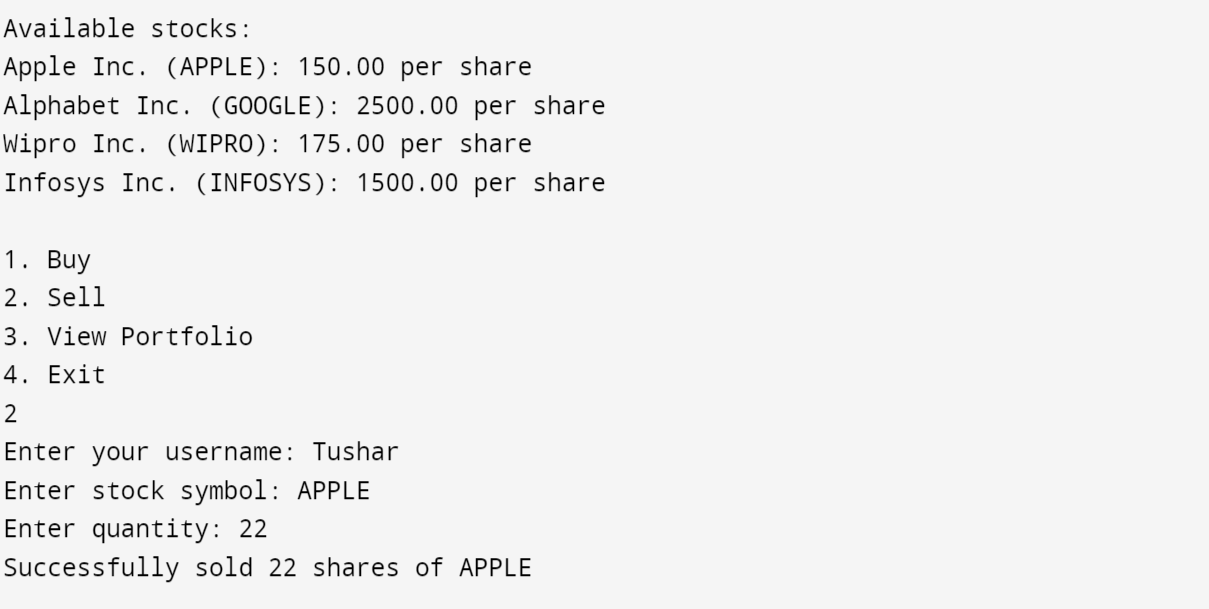
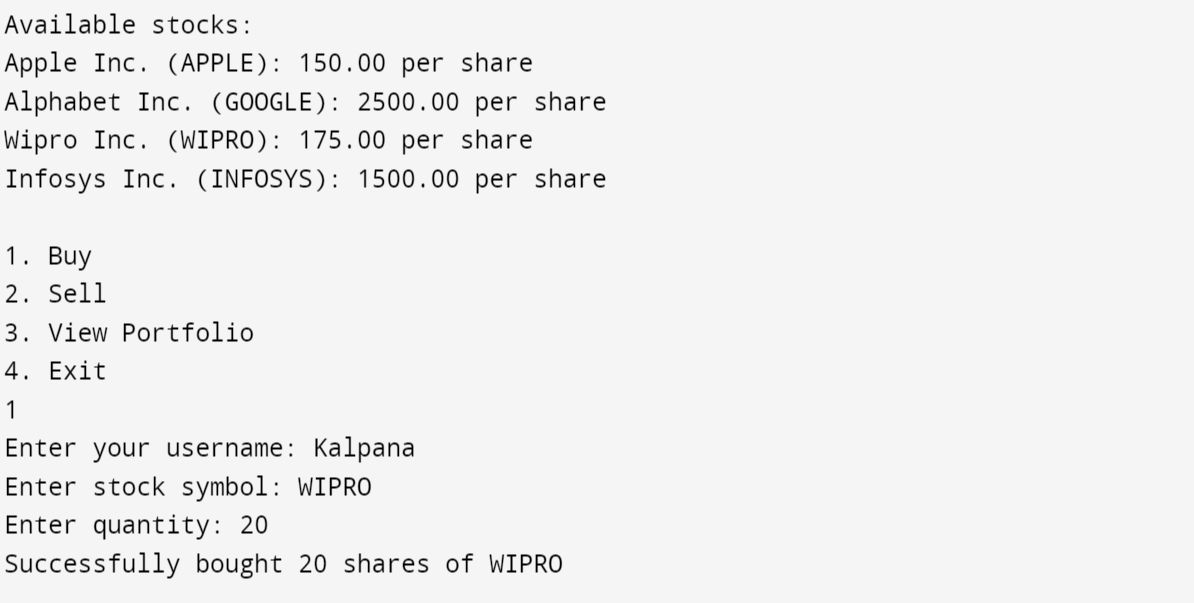
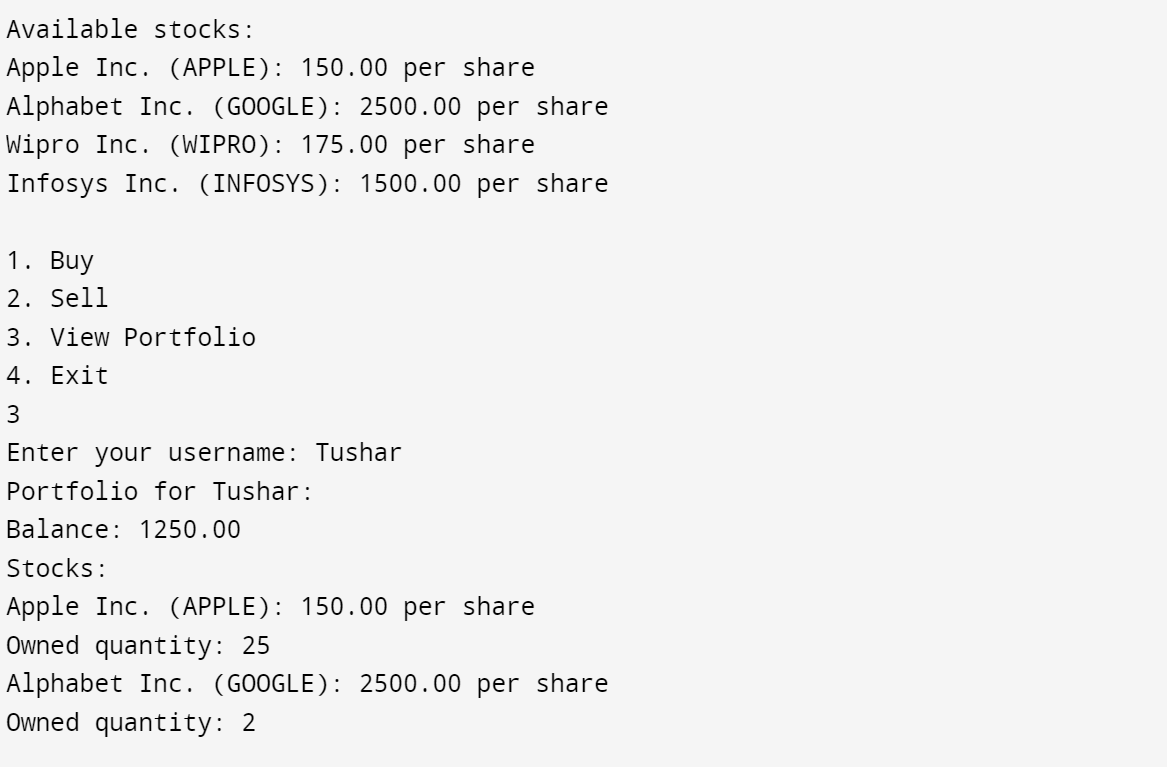
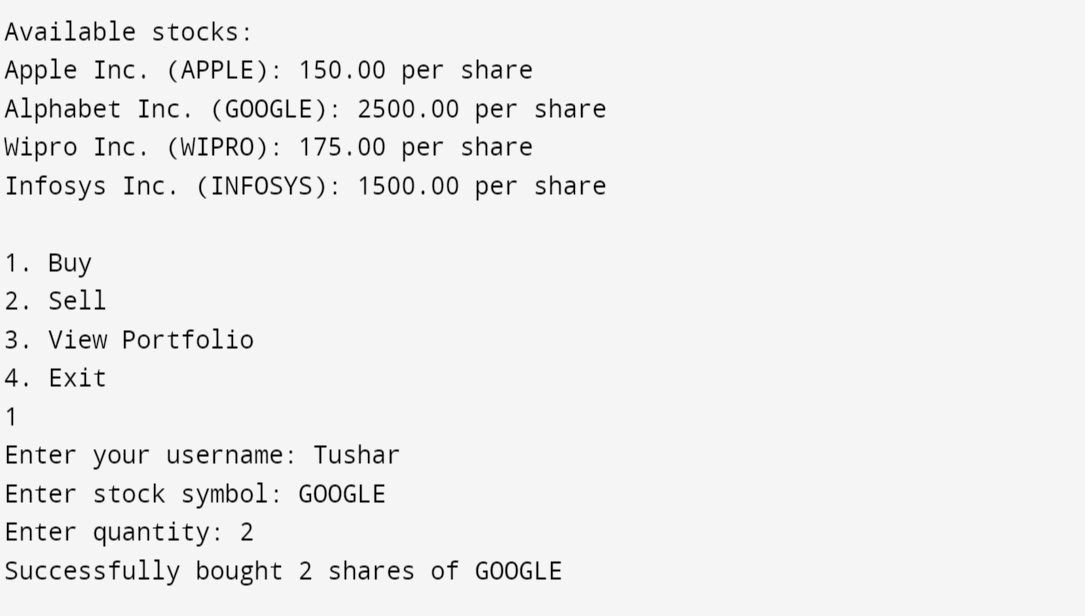
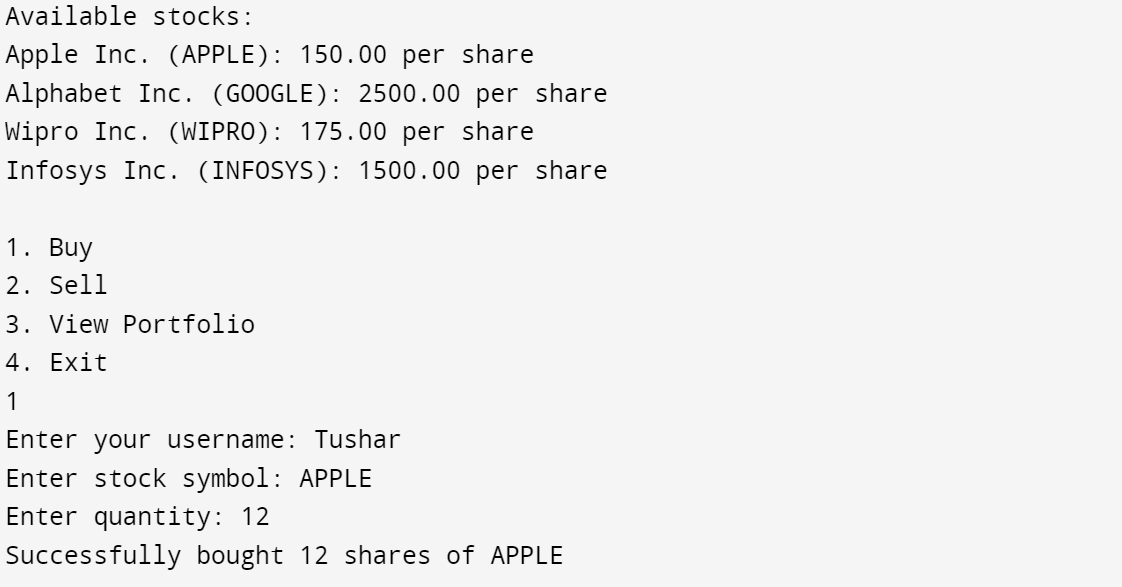
        }

    }

    return 0;

}

**Outputs:**

****