

```

#include<iostream>

using namespace std;

class book_store{
    private:
        int id;          // primary key - unique id for each transactions
        string customer_name; //storing name of customer
        string book_name;   //storing name of book that customer purchased
        string author_name; //storing purchased book's author name
        int price;         //storing price of the book purchased
        string date;       //storing date at which the book is purchased

    public:
        void read(int rid);    //reading the data to be stored and accepting id as 'rid'
from int main()

        void display();       //displaying the data that is stored
        void update();        //updating the data that is stored
        void del();           //deleting any particular transaction
        void insert(int iid); //inserting one another transaction with accepting id as 'iid'
from int main()

};

//READ

void book_store::read(int rid){
    id = rid;
    cout << "\n\nEnter customer name: ";
    cin >> customer_name;
    cout << "Enter book name: ";
    cin >> book_name;
    cout << "Enter author name: ";
    cin >> author_name;
    cout << "Enter price of the book: ";

```

```

        cin >> price;

        cout << "Enter the date at which the book is purchased: ";

        cin >> date;

        cout << "Transaction added successfully at id: " << id;

    }

```

//DISPLAY

```

void book_store::display(){

    cout << "\n\nTransaction ID: " << id << endl;

    cout << "Customer name: " << customer_name << endl;

    cout << "Book name: " << book_name << endl;

    cout << "Author name: " << author_name << endl;

    cout << "Price: " << price << endl;

    cout << "Date: " << date << endl;

}

```

//UPDATE

```

void book_store::update(){

    int choice;

    cout << "Enter: \n\n1 to update: Customer name\n" << "2 to update: Book name\n" <<
    "3 to update: Author name\n" << "4 to update: Price\n" << "5 to update: Date\n";

    cin >> choice;

    cout << "\n\nTransaction ID: " << id << endl;

    cout << "Customer name: " << customer_name << endl;

    cout << "Book name: " << book_name << endl;

    cout << "Author name: " << author_name << endl;

    cout << "Price: " << price << endl;

    cout << "Date: " << date << endl;

    switch(choice){

    case 1:

        cout << "Enter customer name: ";

```

```

        cin >> customer_name;

        break;

case 2:

        cout << "Enter book name: ";

        cin >> book_name;

        break;

case 3:

        cout << "Enter author name: ";

        cin >> author_name;

        break;

case 4:

        cout << "Enter price: ";

        cin >> price;

        break;

case 5:

        cout << "Enter date: ";

        cin >> date;

        break;

default:

        cout << "Wrong Choice!";

        return;

}

cout << "\n\nTransaction ID: " << id << endl;

cout << "Customer name: " << customer_name << endl;

cout << "Book name: " << book_name << endl;

cout<< "Author name: " << author_name << endl;

cout << "Price: " << price << endl;

cout << "Date: " << date << endl;

cout << "Transaction Updated Successfully!" << "\n\n\n";

}

```

```
//DELETE
```

```
void book_store::del(){  
    id--;  
}
```

```
//INSERT
```

```
void book_store::insert(int iid){  
    id = iid;  
    cout << "\n\nEnter customer name: ";  
    cin >> customer_name;  
    cout << "Enter book name: ";  
    cin >> book_name;  
    cout << "Enter author name: ";  
    cin >> author_name;  
    cout << "Enter price of the book: ";  
    cin >> price;  
    cout << "Enter the date at which the book is purchased: ";  
    cin >> date;  
    cout << "Transaction Inserted Successfully at id: " << id << "\n\n\n";  
}
```

```
int main(){
```

```
    cout << "Welcome to Book Store!\n\n";  
    int n; //Total number of transactions.  
    cout << "Enter total number of transactions: ";  
    cin >> n; //accept no of transactions
```

```
    book_store t[100]; // created an array object 't[100]' for storing transactions.
```

```
//read all transactions:
for(int i=0; i < n; i++){
    t[i].read(i); //Here the value of i is passed to read() function to store it as ID
    automatically.
}
```

```
cout << "\nAll transactions are saved successfully!\n";
```

```
//asking for choices to do functions on data:
```

```
ask:
```

```
cout << "\n\n-----\n";
```

```
int ch;
```

```
cout << "Enter the number: \n";
```

```
cout << "1 - to Display transactions\n";
```

```
cout << "2 - to Update any transaction\n";
```

```
cout << "3 - to Delete any transaction\n";
```

```
cout << "4 - to Insert one another transaction\n";
```

```
cin >> ch;
```

```
switch (ch){
```

```
    case 1:
```

```
        for(int i=0; i<n; i++){
```

```
            t[i].display();
```

```
        }
```

```
        break;
```

```
    case 2:
```

```
        int id;
```

```
        cout << "Enter transaction ID to update: ";
```

```
        cin >> id;
```

```
        if(id >= n || id < 0){
```

```

        cout << "Wrong ID!\n";
        goto ask;
        break;
    }
    t[id].update();
    break;

```

case 3:

```

    int id2;
    cout << "Enter transaction ID to delete: ";
    cin >> id2;
    if(id2 >= n || id2 < 0){
        cout << "Wrong ID!\n";
        goto ask;
        break;
    }
    for(int i=id2; i<n-1; i++){
        t[i]=t[i+1];
        t[i].del();
    }
    n--; //as the no. of transaction decreased by 1
    cout << "Deleted successfully!\n";
    break;

```

case 4:

```

    t[n].insert(n); //here n is passed so that the id of the new transaction can
    be added automatically
    n++; //as the no. of transaction increased by 1
    break;

```

default:

```
        cout << "Wrong choice!\n";  
        goto ask;  
    }  
  
    goto ask;  
    return 0;  
}
```