

-----CHINA BAZAR-----

```
#include <iostream>
#include <exception>
```

```
using namespace std;
class ExceptionHandler : public exception
{
    public:
        const char * what() const throw()
        {
            return "Invalid Customer Name\nProvide a proper Customer Name\n";
        }
};
class CustomerDetails
{
    private:
        string customerId;
        string customerName;
        long int phoneNumber;
        string emailId;
        string toyType;
        double price;
    public:
        CustomerDetails()
        {

        }
        CustomerDetails(string customerId, string customerName, long int phoneNumber, string
        emailId, string toyType,double price)
        {
            this->customerId = customerId;
            this->customerName = customerName;
            this->phoneNumber = phoneNumber;
            this->emailId = emailId;
            this->toyType = toyType;
            this->price = price;
        }
        void validateCustomerName()
        {
            for(int i = 0; i < customerName.length(); i++)
            {
                if(!((customerName[i]>='a'      &&      customerName[i]<='z')      ||
(customerName[i]>='A' && customerName[i]<='Z'))))
                {
                    // cout<<customerName[i];
                    ExceptionHandler obj;
                    throw obj;
                    break;
                }
            }
        }
        double calculateDiscount()
        {
            double amount=0.0;
            if(toyType=="softToys")
                amount=price-(price*0.05);
            else if(toyType=="fidgetToys")
                amount=price-(price*0.10);
        }
    };
};
```

```

        else if(toyType=="sensoryToys")
            amount=price-(price*0.15);
        else if(toyType=="puzzles")
            amount=price-(price*0.20);
        return amount;
    }
    ~CustomerDetails()
    {
        cout<<"CustomerDetails Object is destroyed"<<endl;
    }
};
int main()
{
    string id,name,email,type;
    long int phone;
    double price,amount;
    cout<<"Enter Customer Id"<<endl;
    cin>>id;
    cout<<"Enter Customer Name"<<endl;
    cin>>name;
    cout<<"Enter Phone Number"<<endl;
    cin>>phone;
    cout<<"Enter Email Id"<<endl;
    cin>>email;
    cout<<"Enter type"<<endl;
    cin>>type;
    cout<<"Enter Price"<<endl;
    cin>>price;

    CustomerDetails *c=new CustomerDetails(id,name,phone,email,type,price);
    try
    {
        c->validateCustomerName();
        amount=c->calculateDiscount();
        cout<<"Amount to be paid by the Customer "<<amount<<endl;
    }
    catch(exception& e)
    {
        cout << e.what();
    }

    delete c;
    return 0;
}

```

-----HELPING HAND-----

```

#include <iostream>
#include <string.h>
//#include <conio.h>
#include <cstdio>
#include <cstdlib>
using namespace std;

```

```

class Endowment
{
    private:
        string endowmentId;
        string holderName;

```

```

        string endowmentType;
        string registrationDate;
    public:
        virtual double calculateEndowment()=0;

        string getEndowmentId(){
            return endowmentId;
        }

        Endowment(){}
        Endowment(string endowmentId, string holderName, string endowmentType, string
registrationDate)
        {
            this->endowmentId = endowmentId;
            this->holderName = holderName;
            this->endowmentType = endowmentType;
            this->registrationDate = registrationDate;
        }
};

class HealthEndowment : public Endowment
{
    private:
        string healthCareCenter;
        int holderAge;
    public:
        string getHealthCareCenter(){
            return healthCareCenter;
        }
        void setHealthCareCenter(string healthCareCenter){
            this->healthCareCenter = healthCareCenter;
        }
        int getHolderAge(){
            return holderAge;
        }
        void setHolderAge(int holderAge){
            this->holderAge = holderAge;
        }

        HealthEndowment(){}
        HealthEndowment(string endowmentId, string holderName, string endowmentType,
string
registrationDate, string healthCareCenter, int holderAge)
:Endowment(endowmentId, holderName, endowmentType, registrationDate)
        {
            this->healthCareCenter = healthCareCenter;
            this->holderAge = holderAge;
        }
        double calculateEndowment()
        {
            double amount=0;
            if(holderAge<=30)
                amount=120000;
            else if(holderAge>30 && holderAge<60)
                amount=200000;
            else if(holderAge>=60)
                amount=500000;

            return amount;
        }
};

```

```

class EducationalEndowment : public Endowment
{
    private:
        string educationalInstitution;
        string educationalDivision;

    public:
        string getEducationalInstitution(){
            return educationalInstitution;
        }
        void setEducationalInstitution(string educationalInstitution){
            this->educationalInstitution = educationalInstitution;
        }
        string getEducationalDivision(){
            return educationalDivision;
        }
        void setEducationalDivision(string educationalDivision){
            this->educationalDivision = educationalDivision;
        }

        EducationalEndowment(){}
        EducationalEndowment(string endowmentId, string holderName, string endowmentType,
string          registrationDate,string          educationalInstitution,          string
educationalDivision):Endowment(endowmentId,          holderName,          endowmentType,
registrationDate)
        {
            this->educationalInstitution = educationalInstitution;
            this->educationalDivision = educationalDivision;
        }
        double calculateEndowment()
        {
            double amount=0;
            if(educationalDivision=="School")
                amount=30000;
            else if(educationalDivision=="UnderGraduate")
                amount=60000;
            else if(educationalDivision=="PostGraduate")
                amount=90000;
            return amount;
        }
};

int main()
{
    Endowment *e,*h;
    string id,name,type,date,ins,divi,center;
    int age;
    double amount;
    cout<<"Enter Endowment Id"<<endl;
    cin>>id;
    cout<<"Enter Holder Name"<<endl;
    cin>>name;
    cout<<"Enter Endowment Type"<<endl;
    cin>>type;
    cout<<"Enter Registration Date"<<endl;
    cin>>date;
    if(type=="educational")
    {
        cout<<"Enter Educational Institution"<<endl;

```

```

        cin>>ins;
        cout<<"Enter Educational Division"<<endl;
        cin>>divi;
        e=new EducationalEndowment(id,name,type,date,ins,divi);
        amount=e->calculateEndowment();
        cout<<"Endowment Amount "<<amount<<endl;
    }

    else if(type=="health")
    {
        cout<<"Enter Health Care Center"<<endl;
        cin>>center;
        cout<<"Enter Holder Age"<<endl;
        cin>>age;
        h=new HealthEndowment(id,name,type,date,center,age);
        amount=h->calculateEndowment();
        cout<<"Endowment Amount "<<amount<<endl;
    }

    return 0;
}

```

-----BUS TICKET RESERVATION MARK 62-----

```

#include <iostream>
#include <string.h>
#include <string>
using namespace std;

struct Bus
{
    int busNo;
    string busType;
    string busName;
    string source;
    string destination;
    float departureTime;
    float arrivalTime;
    int noOfSeats;
    double ticketCost;

    void displayBusDetails() // displayBusDetails() method is provided as a part of code
    skeleton
    {

        cout<<busNo<<"\t"<<busName<<"\t"<<busType<<"\t"<<source<<"\t"<<destination<<"\t"<<
        departureTime <<"\t"<<arrivalTime<<"\t"<<noOfSeats<<"\t"<<ticketCost<<endl;
    }

    //Fill the code here
    void checkAvailability(string source, string destination);
    /*{
        for(int i=0;i<10;i++){
            if(source==busObj[i].source && destination==busObj[i].destination) {
                cout<<busObj[i].displayBusDetails();
            }
        }
    }*/

    bool bookATicket(string from,string to,string btype,int no);
}

```

```

double calculateTotalFare(int noOfPassengers){
    return noOfPassengers*ticketCost;
}

}

busObj[10]={ {1122,"ACSeater","SRS","Coimbatore","Chennai",9.15f,5.35f,30,450.00},
             {1023,"ACSeater","PNR","Coimbatore","Bangalore",8.15f,5.35f,20,550.00},
             {1123,"NonACSeater","SRS","Coimbatore","Pune",10.15f,7.35f,30,300.00},

             {1234,"NonACSeater","PSS","Coimbatore","Hyderabad",7.15f,4.35f,18,500.00},
             {1128,"ACSleeper","ORS","Chennai","Coimbatore",9.15f,5.35f,30,450.00},
             {1063,"ACSleeper","KPY","Bangalore","Coimbatore",8.15f,5.35f,20,550.00},
             {7123,"NonACSleeper","MMC","Pune","Coimbatore",10.15f,7.55f,30,300.00},

             {1784,"NonACSleeper","PSS","Hyderabad","Coimbatore",7.15f,4.35f,18,500.00},
             {7122,"NonACSeater","PSS","Pune","Chennai",7.15f,4.35f,18,500.00},
             {1100,"ACSleeper","ORS","Chennai","Pune",9.15f,5.45f,30,450.00}};;

//Initialize the array using the below values
Bus bus[] = { {1122,"ACSeater","SRS","Coimbatore","Chennai",9.15f,5.35f,30,450.00},
              {1023,"ACSeater","PNR","Coimbatore","Bangalore",8.15f,5.35f,20,550.00},
              {1123,"NonACSeater","SRS","Coimbatore","Pune",10.15f,7.35f,30,300.00},

              {1234,"NonACSeater","PSS","Coimbatore","Hyderabad",7.15f,4.35f,18,500.00},
              {1128,"ACSleeper","ORS","Chennai","Coimbatore",9.15f,5.35f,30,450.00},
              {1063,"ACSleeper","KPY","Bangalore","Coimbatore",8.15f,5.35f,20,550.00},
              {7123,"NonACSleeper","MMC","Pune","Coimbatore",10.15f,7.55f,30,300.00},

              {1784,"NonACSleeper","PSS","Hyderabad","Coimbatore",7.15f,4.35f,18,500.00},
              {7122,"NonACSeater","PSS","Pune","Chennai",7.15f,4.35f,18,500.00},
              {1100,"ACSleeper","ORS","Chennai","Pune",9.15f,5.45f,30,450.00}};

//busObj[]=bus[];

Bus selectedBusObj=bus[10];
//Assign the selected bus values to this object

void Bus :: checkAvailability(string source, string destination){
    for(int i=0;i<10;i++){
        if(source==bus[i].source && destination==bus[i].destination) {

            cout<<bus[i].busNo<<"\t"<<bus[i].busName<<"\t"<<bus[i].busType<<"\t"<<bus[i].source<<"\t"<<
            bus[i].destination<<"\t"<<bus[i].departureTime
            <<"\t"<<bus[i].arrivalTime<<"\t"<<bus[i].noOfSeats<<"\t"<<bus[i].ticketCost<<endl;
        }
    }

}

bool Bus :: bookATicket(string from,string to,string btype,int no){
    for(int i=0;i<10;i++){
        if(from==busObj[i].source && to==busObj[i].destination &&
        btype==busObj[i].busType && no <= busObj[i].noOfSeats) {
            return true;
            // calculateTotalFare(no);
        }
    }
}

```

```

        return false;
    }

int main()
{
    //Fill the code here
    busObj[2].displayBusDetails();
    int a=1;
    while(a){
        int x;
        string s1="";
        string s2="";
        string from,to,date,name,type;
        int num;
        cout<<"1. Check availability"<<endl;
        cout<<"2. Book Ticket"<<endl;
        cout<<"3. Exit"<<endl;
        cout<<"Enter your choice:"<<endl;
        cin>>x;
        switch(x){
            case 1:

                cout<<"Enter the Source:"<<endl;
                cin>>s1;
                cout<<"Enter the Destination:"<<endl;
                cin>>s2;
                selectedBusObj.checkAvailability(s1,s2);
                // cout<<bus[1].busNo;
                break;

            case 2:
                cout<<"From:"<<endl;
                cin>>from;
                cout<<"To:"<<endl;
                cin>>to;
                cout<<"Date of Journey:"<<endl;
                cin>>date;
                cout<<"Passenger Name:"<<endl;
                cin>>name;
                cout<<"Bus      Type      (ACSleeper/      ACSeater/      NonACSleeper/
NonACSeater):"<<endl;
                cin>>type;
                cout<<"Number of passengers:"<<endl;
                cin>>num;
                if(selectedBusObj.bookATicket(from,to,type,num)){
                    cout<<"Dear "<<name<<endl;
                    cout<<"Congratulations! Thank you for using online road reservation
facility. Your booking details are indicated below."<<endl;
                    cout<<"*****"<<endl;
                    cout<<"Passenger Name      : "<<name<<endl;
                    cout<<"Date of Journey      : "<<date<<endl;
                    cout<<"From                  : "<<from<<endl;
                    cout<<"To                  : "<<to<<endl;
                    cout<<"Number of passengers : "<<num<<endl;
                    cout<<"Total      Fare                  :
"<<selectedBusObj.calculateTotalFare(num)<<endl;

                }
                else
                {
                    cout<<"Booking Cancelled!!!! Please check the availability for your

```

```

        preference."<<endl;
    }
    break;
    case 3:
        a=0;
        break;
    }
}
}
}

```

-----BUS TICKET RESERVATION-ANOTHER VERSION-----

```

#include <iostream>
#include <string.h>
#include <string>
using namespace std;

struct Bus
{
    int busNo;
    string busType;
    string busName;
    string source;
    string destination;
    float departureTime;
    float arrivalTime;
    int noOfSeats;
    double ticketCost;

    void displayBusDetails() // displayBusDetails() method is provided as a part of code
    skeleton
    {

        cout<<busNo<<"\t"<<busName<<"\t"<<busType<<"\t"<<source<<"\t"<<destination<<"\t"<<
        departureTime <<"\t"<<arrivalTime<<"\t"<<noOfSeats<<"\t"<<ticketCost<<endl;
    }

    //Fill the code here
    void checkAvailability(string source, string destination);
    /*{
        for(int i=0;i<10;i++){
            if(source==busObj[i].source && destination==busObj[i].destination) {
                cout<<busObj[i].displayBusDetails();
            }
        }
    }*/

    bool bookATicket(string from,string to,string btype,int no);

    double calculateTotalFare(int noOfPassengers){
        return noOfPassengers*ticketCost;
    }

} busObj[10]={ {1122,"ACSeater","SRS","Coimbatore","Chennai",9.15f,5.35f,30,450.00},
               {1023,"ACSeater","PNR","Coimbatore","Bangalore",8.15f,5.35f,20,550.00},

```



```

        {1123,"NonACSeater","SRS","Coimbatore","Pune",10.15f,7.35f,30,300.00},

{1234,"NonACSeater","PSS","Coimbatore","Hyderabad",7.15f,4.35f,18,500.00},
        {1128,"ACSleeper","ORS","Chennai","Coimbatore",9.15f,5.35f,30,450.00},
        {1063,"ACSleeper","KPY","Bangalore","Coimbatore",8.15f,5.35f,20,550.00},
        {7123,"NonACSleeper","MMC","Pune","Coimbatore",10.15f,7.55f,30,300.00},

{1784,"NonACSleeper","PSS","Hyderabad","Coimbatore",7.15f,4.35f,18,500.00},
        {7122,"NonACSeater","PSS","Pune","Chennai",7.15f,4.35f,18,500.00},
        {1100,"ACSleeper","ORS","Chennai","Pune",9.15f,5.45f,30,450.00}};

//Initialize the array using the below values
Bus bus[] = { {1122,"ACSeater","SRS","Coimbatore","Chennai",9.15f,5.35f,30,450.00},
               {1023,"ACSeater","PNR","Coimbatore","Bangalore",8.15f,5.35f,20,550.00},
               {1123,"NonACSeater","SRS","Coimbatore","Pune",10.15f,7.35f,30,300.00},

               {1234,"NonACSeater","PSS","Coimbatore","Hyderabad",7.15f,4.35f,18,500.00},
               {1128,"ACSleeper","ORS","Chennai","Coimbatore",9.15f,5.35f,30,450.00},
               {1063,"ACSleeper","KPY","Bangalore","Coimbatore",8.15f,5.35f,20,550.00},
               {7123,"NonACSleeper","MMC","Pune","Coimbatore",10.15f,7.55f,30,300.00},

               {1784,"NonACSleeper","PSS","Hyderabad","Coimbatore",7.15f,4.35f,18,500.00},
               {7122,"NonACSeater","PSS","Pune","Chennai",7.15f,4.35f,18,500.00},
               {1100,"ACSleeper","ORS","Chennai","Pune",9.15f,5.45f,30,450.00}};

//busObj[] = bus[];

Bus selectedBusObj = bus[10];
//Assign the selected bus values to this object

void Bus :: checkAvailability(string source, string destination){
    for(int i=0;i<10;i++){
        if(source==bus[i].source && destination==bus[i].destination) {

cout<<bus[i].busNo<<"\t"<<bus[i].busName<<"\t"<<bus[i].busType<<"\t"<<bus[i].source<<"\t"<<
bus[i].destination<<"\t"<<bus[i].departureTime
<<"\t"<<bus[i].arrivalTime<<"\t"<<bus[i].noOfSeats<<"\t"<<bus[i].ticketCost<<endl;
        }
    }

}

bool Bus :: bookATicket(string from,string to,string btype,int no){
    for(int i=0;i<10;i++){
        if(from==busObj[i].source && to==busObj[i].destination &&
btype==busObj[i].busType && no <= busObj[i].noOfSeats) {
            return true;
            // calculateTotalFare(no);
        }
    }
    return false;
}

int main()
{
    //Fill the code here
    busObj[2].displayBusDetails();
    int a=1;
    while(a){
        int x;
        string s1="";

```

```

        string s2="";
        string from,to,date,name,type;
        int num;
        cout<<"1. Check availability"<<endl;
        cout<<"2. Book Ticket"<<endl;
        cout<<"3. Exit"<<endl;
        cout<<"Enter your choice:"<<endl;
        cin>>x;
        switch(x){
            case 1:

                cout<<"Enter the Source:"<<endl;
                cin>>s1;
                cout<<"Enter the Destination:"<<endl;
                cin>>s2;
                selectedBusObj.checkAvailability(s1,s2);
                // cout<<bus[1].busNo;
                break;
            case 2:
                cout<<"From:"<<endl;
                cin>>from;
                cout<<"To:"<<endl;
                cin>>to;
                cout<<"Date of Journey:"<<endl;
                cin>>date;
                cout<<"Passenger Name:"<<endl;
                cin>>name;
                cout<<"Bus      Type      (ACsleeper/      ACSeater/      NonACsleeper/
NonACSeater):"<<endl;
                cin>>type;
                cout<<"Number of passengers:"<<endl;
                cin>>num;
                if(selectedBusObj.bookATicket(from,to,type,num)){
                    cout<<"Dear "<<name<<endl;
                    cout<<"Congratulations! Thank you for using online road reservation
facility. Your booking details are indicated below."<<endl;
                    cout<<"*****"<<endl;
                    cout<<"Passenger Name      : "<<name<<endl;
                    cout<<"Date of Journey      : "<<date<<endl;
                    cout<<"From                  : "<<from<<endl;
                    cout<<"To                      : "<<to<<endl;
                    cout<<"Number of passengers : "<<num<<endl;
                    cout<<"Total      Fare                  :
"<<selectedBusObj.calculateTotalFare(num)<<endl;

                }
            else{
                cout<<"Booking Cancelled!!!! Please check availability for your preference."<<endl;
            }
            break;
            case 3:
                a=0;
                break;
        }
    }
}

```