|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | |  | | |
|  |  |
|  |  |
|  | |
| PREPARED BY:  (22k-4217): Fatima Farhan  PREPARED BY:  (22k-4335): Urwa Junaid  PREPARED BY:  (22k-4272): Najaah Noor | | |  | USER SPECIFIED MANAGEMENT SYSTEMOOP PROJECT SECTION:2-D |

|  |  |  |  |
| --- | --- | --- | --- |
| CONCEPTS USED:   * 1. Encapsulation   2. Abstraction: abstract classes: EMPLOYEE AN DEMPLOYEES’ CHILD CLASS DOCTOR   3. Pointers and pointer array using dma   4. Late binding: overloading functions: in inventory class: restock.   5. Member initialization   6. Static: receptionist salary   7. Destructor: of hotel and hospital   8. Dma = used in pointer arrays   9. Virtual destructor: doctors (maybe)   10. Abstract: implemented in doc and emp--   11. Generic: templates -   12. Filing in receptionist -- | |  | IntroductionUSER SPECIFIED MANAGEMENT SYSTEM Our project presents a program to a start-up CEO with the purpose of helping them to create a management system. Instead of giving them a fixed structured management system our program allows flexibility for the CEO and allows them to provide their own specifications for the management system. Our program provides a choice between a hospital and hotel.  BACKGROUND:  The project we chose is a management system. The reason for this being that a program as such shows proper implementation of OOP concepts. Moreover, in order increase the complexity level we decided to make it a user specified management system.  PROBLEM ANALYSIS:  Our program aims to solve the problem of helping small startup CEOs. Allowing them to create a management system according their needs. |
|  |  | |  |
|  |  | |
| g #include<iostream>  #include<string.h>  #include<vector>  #include<cstdlib>  #include<ctime>  #include<fstream>  using namespace std;  //-------------------------------------------------//  //--------------------FINANCE---------------------//  //-----------------------------------------------//  class finance  {  friend class shopkeeper;  friend class chef;  friend class cleaner;  friend class doctor;  static float budget;  static float profit;  static float expenditure;  float taxrate;  class inventory;  float bill;  vector<float>cleaningitems; //price  vector<float>chef\_items;  vector <float> shop\_items;  public:  //------setters for budget profit and taxrate-------//  void setbudget(float o)  {budget=o;}  void setexpenditure(float o)  {expenditure=o;}  void setprofit(float o)  {profit=o;}  void settax(float o)  {taxrate=o;}  //------getters for budget profit and taxrate-------//  float getbudget()  {return budget;}  float getexpenditure()  {return expenditure;}  float getprofit()  {return profit;}  float gettax()  {return taxrate;}  //--------operation on finance attributes---------//  void addinbudget(float t)  {budget=budget+t;}  void subinbudget(float t)  {  budget=budget-t;  expenditure=expenditure+t;  }  void setbill(float n)  {  bill=bill+n;  }  float getbill()  {  return bill;  }  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//  // factory bought //  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//  void cleanset()  {float t;  float upper=50.0;  float lower=20.0;  for(int i=0;i<100;i++)  {  t=(rand() % 50)+20;  cleaningitems.push\_back(t);  }  }  void chefset()  {float t;  float upper=35.0;  float lower=10.0;  for(int i=0;i<100;i++)  {  t=(rand() % 50)+20;  chef\_items.push\_back(t);  }  }  void shopset()  {float t;  float upper=40.0;  float lower=10.0;  for(int i=0;i<100;i++)  {  t=(rand() % 50)+20;  shop\_items.push\_back(t);  }  }  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//  // RESTOCKING //  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*//  void chefrestock(int i,int n)  {  float sub;  sub=chef\_items[i]\*n;  subinbudget(sub);  }  void cleanrestock(int i,int n)  {  float sub;  sub=cleaningitems[i]\*n;  subinbudget(sub);  }  void shoprestock(int i,int n)  {  float sub;  sub=shop\_items[i]\*n;  subinbudget(sub);  }  };  float finance::budget=0;  float finance::expenditure=0;  float finance::profit=0;  //-------------------------------------    //INVENTORY  //-------------------------------------  class inventory{  friend class shopkeeper;  friend class chef;  friend class cleaner;  friend class doctor;  friend class finance;  friend class manager;  vector<int>cleaningitems;  vector<int>chef\_items;//quant  vector <int> shop\_items;  finance f;  public:  inventory()  {  f.chefset();  f.cleanset();  f.shopset();  }  //functions  void buyandsell(){} //subtracting from inventory  void restock() //cleaner  {int n;  for(int i=0;i<6;i++)  {  if(cleaningitems[i]==0)  {  cout<<"enter by how much do you wish to restock"<<endl;  cin>>n;  f.cleanrestock(i,n);  cleaningitems[i]=n;  break;  }  }  cout<<"Restocking done"<<endl;  }  void restock(string nm, int index) //chef  {  int n;  //ask by how many to increment  cout<<"restocking"<<nm<<endl;  cout<<"enter by how much do you wish to restock"<<endl;  cin>>n;  f.chefrestock(index,n);  chef\_items[index]=n;  }  void restock(int index) //shopkeeper  {  int n;  //ask by how many to increment  cout<<"enter by how much do you wish to restock"<<endl;  cin>>n;  f.chefrestock(index,n);  shop\_items[index]=n;  }  };  //-------employee-----------//  //CLASS 3  class employee  {  int id;  string name;  int office\_hrs,num\_of\_emp;  double bonus;  static double total\_sal;  double mon\_salary;  public:  //constructer  employee(int i)  {}  employee()  {  }  //setter and getters  int getid()  {  return id;  }  void setname(string name){  this->name=name;  }  string getname(){  return name;  }  void setofficehrs(int office\_hrs){  this->office\_hrs=office\_hrs;  }  int getofficehrs(){  return office\_hrs;  }  void setbonus(double bonus){  this->bonus=bonus;  }  double getbonus(){  return bonus;  }  void setnumofemployees(int num\_of\_emp){  this->num\_of\_emp=num\_of\_emp;  }  int getnumofemployees(){  return num\_of\_emp;  }  //function  void inputfunct(){ //abstract  }  void workfunction(){  }  };  //-----------------------------------  //DOCTOR  //--------------------------------------  class doctor :protected employee  {  protected :  float sal;  friend class manager;    public:  //parametrized construtor  doctor(int i):employee(i)  {  }  doctor()  {  }  //member functions  virtual void workfunction()=0;  };  //-----------------ENT CHILD CLASS:DOCTOR----------------------  class ENT\_specialist:public doctor  {  public:  //member functions  ENT\_specialist(int i):doctor(i)  {}  ENT\_specialist()  {}  void setsal(float t)  {  sal=t;  }  float getsal()  {return sal;}  void workfunction()  {  string disease;  cout<<"enter which of the follwing do you have"<<endl  <<"earinfection"<<endl  <<"strepthroat"<<endl  <<"sinusitis"<<endl;  cin>>disease;  if(disease=="earinfection")  {  cout<<"prescription:"<<endl  <<" 1.Ceclara(500 mg 3times)"<<endl<<  "2. painkiller "<<endl<<  "3. cypotec (ear drops) "<<endl;  }  else if(disease=="strepthroat")  {  cout<<"prescription:"<<endl  <<"1.Azomax "<<endl  <<"2.Amoxil (500mg 3times) "<<endl;  }  else if(disease=="sinusitis")  {  cout<<"prescription:"<<endl  <<"1.Avelox (400mg 1 time night)"<<endl  <<" 2.Talfast "<<endl;  }  }  };  //-----------------SURGEON CHILD CLASS:DOCTOR----------------------  class surgeon:public doctor  {  public:  //member functions  surgeon(int i):doctor(i)  {}  surgeon()  {  }  void setsal(float t)  {  sal=t;  }  float getsal()  {return sal;}  void workfunction()  {  string disease;  cout<<"enter which of the follwing do you have"<<endl  <<"hernia"<<endl  <<"cholilithiasis"<<endl  <<"Kidneystone"<<endl;  cin>>disease;  if(disease=="hernia")  {  cout<<"type of surgery required: Hernioraphy "<<endl;  }  else if(disease=="cholilithiasis")  {  cout<<"type of surgery required: Cholisystectomy "<<endl;  }  else if(disease=="Kidneystone")  {  cout<<"type of surgery required: Ureteroscopy (lithotripsy)"<<endl;  }  }  };  //-----------------GENERAL PHYSICIAN CHILD CLASS:DOCTOR----------------------  class gen\_phy:public doctor  {  public:  //member functions    gen\_phy(int i):doctor(i)  {}  gen\_phy()  {    }  void setsal(float t)  {  sal=t;  }  float getsal()  {return sal;}  void workfunction()  {  string disease;  cout<<"enter which of the follwing do you have"<<endl  <<"dysentry"<<endl  <<"pneumonia"<<endl  <<"gastritis"<<endl;  cin>>disease;  if(disease=="dysentry")  {  cout<<"prescription:"<<endl  <<"1.Ciproxin"<<endl<<  "2.Flagyl"<<endl  <<"3.ORS"  <<endl;  }  else if(disease=="pneumonia")  {  cout<<"prescription:"<<endl  <<"1.Augmentin(1 grm daily)"<<endl  <<" 2.Paracetamol(2 tablets 3 times a day)"<<endl  <<" 3.Acefyll "<<endl;  }  else if(disease=="gastritis")  {  cout<<"prescription:"<<endl  <<"1.Risek (20mg 2 times)"<<endl  <<" 2.Nospa fort (2 times)"<<endl  <<" 3.Itopride (50mg) "<<endl;  }  }  };  //--------------------------------------  //SHOPKEEPER CLASS  //--------------------------------------  class shopkeeper:protected employee{  friend class manager;  int no\_of\_items; //size of array  int \*noof\_items\_price; //make setter n getters  float bill=0.0;  float \*price;  string \*menu;  inventory inv;  int quantity;  float salary;  finance f;  float pay,left;  public:  shopkeeper(int i):employee(i)  {    }  shopkeeper()  {}  void setsal(float salary )  {  this->salary=salary;  }  float getsal()  {  return salary;  }  void input()  {  cout<<"--------------------TAKING INPUTS FOR SHOP------------------"<<endl;  cout<<"enter the salary of the shopkeeper"<<endl;  cin>>salary;  f.subinbudget(salary);  cout<<"enter number of items in shop: "<<endl;  cin>>no\_of\_items; //allocate memory dynamically  price = new float [no\_of\_items];  menu = new string [no\_of\_items];    cout<<"enter the items of menu and their quantity. "<<endl;  for(int i=0; i<no\_of\_items; i++)  {  cin>>menu[i]>>quantity;  inv.shop\_items.push\_back(quantity);  }  cout<<"enter the prices of items "<<endl;  for(int i=0; i<no\_of\_items; i++)  {  cin>>price[i];  }  } //items and its quantity  //Dma no of items  //menu array  //menu price array    void display()  {  cout<<"the menu items are: "<<endl;  for(int i=0; i<no\_of\_items; i++){  cout <<i+1<<" "<< menu[i] << " price is "<< price[i] << endl;  }  }  void workfunction(){  //return bill  int choice,n,q,l;  cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*WELCOME TO SHOP\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;  display();  cout<<"enter how many different menu items do you want to buy? "<<endl;  cin>>n;  for(int i=0; i<n; i++)  {  cout<<"enter the position of the item you want to select and its quantity "<<endl;  cin>>choice;  cin>>q;  if(inv.shop\_items[choice-1]==0 || inv.shop\_items[choice-1]<q ){  inv.restock(choice-1);  }    bill = bill + (price[choice-1]\*q);  l=inv.shop\_items[choice-1];  l=l-q;  inv.shop\_items[choice-1]=l;  }  cout<<"the bill is: "<<bill<<endl<<"enter the pay"<<endl;  cin>>pay;  left=pay-bill;  cout<<"your change is: "<<left<<endl;  f.addinbudget(bill);  }  };  //---------------------------------------------------------  //CHEF CLASS  //---------------------------------------------------------  class chef:protected employee  {    //------------attributes-------------  string \*menu;  float \*price;  int nof\_items;  finance f;  float bill=0.0;  float salary;  inventory in;  int quantity;  float pay,left;  //nofitems variable + setter getter  //------------constructor------------  public:  //----------member functions----------  chef(int i):employee(i)  {}  chef()  {  }  void setsal(float salary)  {  this->salary=salary;  }  float getsal()  {  return salary;  }  void setno(int no)  {nof\_items=no;}  void set\_bill(float bill)  {  this->bill=bill;  }  float get\_bill()  {  return bill;  }  void input()  {  cout<<"-------------------TAKING INPUTS FOR CHEF----------------"<< endl;  cout<<"enter the salary of the chef";  cin>>salary;  f.subinbudget(salary);  cout<<"enter no.of items in your menu"<<endl;  cin>>nof\_items;  price= new float [nof\_items];  menu= new string [nof\_items];      cout<<"please enter the name of the menu item you want to add and amount of that dish"<<endl;  for(int i=0;i<nof\_items;i++)  {  cin>>menu[i]>>quantity;  in.chef\_items.push\_back(quantity);  }  cout<<"now enter the prices of each item respectively"<<endl;  for(int i=0;i<nof\_items;i++)  {  cout<<menu[i]<<": "<<endl;  cin>>price[i];  cout<<endl;  }  }  void display()  {  cout<<nof\_items;  cout<<"the following are our menu choices"<<endl;  for(int i=0;i<nof\_items;i++)  {  cout<<i+1<<" "<<menu[i]<<" "<<price[i]<<endl;  }  }  void workfunction()  {  int choice,n,m,q;  cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*WELCOME TO FOOD COURT\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;  cout<<"enter how many different menu items do you want to order? "<<endl;  cin>>n;  display();  for(int i=0;i<n;i++)  {  cout<<"enter the serial number of the item you want "<<endl;  cin>>choice;  choice=choice-1;  cout<<"enter the quantity of the selected item"<<endl;  cin>>m;  int l;  if((in.chef\_items[choice]==0)||(in.chef\_items[choice]<m))  {  in.restock(menu[choice],choice);  }      bill=bill+(price[choice]\*m);  l=in.chef\_items[choice];  l=l-m;  in.chef\_items[choice]=l;  }  f.addinbudget(bill);      }  };  //-----------------------------------------------//  //------------------CLEANER---------------------//  //---------------------------------------------//  class cleaner:protected employee  {friend class manager;  friend class receptionist;  bool check;// to check if room has been cleaned  int no;  int tno;  string itemsname[10]={"TOWELS","BEDSHEETS","CLEANING SPRAYS","SOAPS","SHAMPOO","TOOTPASTE"};  inventory in;  finance f;  float sal;//---------------salary of customers--------------//  public:  cleaner(int id):employee(id)  {}  cleaner()  {    }  void setsal(float t)  {sal=t;}  float getsal()  {return sal;}  void setbool(bool t)  {check=t;}  bool getbool()  {return check;}  void input()  {  cout<<"-------------------TAKING INPUTS FOR CLEANER----------------"<< endl;  cout<<"enter the salary of the cleaner"<<endl;  cin>>sal;  f.subinbudget(sal);  cout<<"Enter quantity for following items"<<endl;  for(int i=0;i<6;i++)  {  cout<<"ENTER QUANTITY OF "<<itemsname[i]<<endl;  cin>>no;  in.cleaningitems.push\_back(no);  }  }  void workfunction()// after customer leaves the function gets automatically called  {  int l;  for(int i=0;i<6;i++)  {  if (in.cleaningitems[i]==0)  {  in.restock();  }    l=in.cleaningitems[i];  l--;  in.cleaningitems[i]=l;    }  }  void customercall()// if customer wants extra bed or towels any service  {    int p,quantity,q;  cout<<"Choose which item you desire to use "<<endl;  for(int i=0;i<6;i++)  {  cout<<i+1<<"."<<itemsname[i]<<endl;  }  //{"TOWELS","BEDSHEETS","CLEANING SPRAYS","SOAPS","SHAMPOO","TOOTPASTE"};  cin>>p;  cout<<"Enter quantity"<<endl;  cin>>quantity;  switch (p)  {  case 1:  {  if (in.cleaningitems[0]==0)  {  in.restock();  }  q=in.cleaningitems[0];  q=q-quantity;  in.cleaningitems[0]=q;  break;  }  case 2:  {  q=in.cleaningitems[1];  if (in.cleaningitems[1]==0)  {  in.restock();  }  q=q-quantity;  in.cleaningitems[1]=q;  break;  }  case 3:  {  q=in.cleaningitems[2];  if (in.cleaningitems[2]==0)  {  in.restock();  }  q=q-quantity;  in.cleaningitems[2]=q;  break;  }  case 4:  {  q=in.cleaningitems[3];  if (in.cleaningitems[3]==0)  {  in.restock();  }  q=q-quantity;  in.cleaningitems[3]=q;  break;  }  case 5:  {  q=in.cleaningitems[4];  if (in.cleaningitems[4]==0)  {  in.restock();  }  q=q-quantity;  in.cleaningitems[4]=q;  break;  }  case 6:  {  q=in.cleaningitems[5];  if (in.cleaningitems[5]==0)  {  in.restock();  }  q=q-quantity;  in.cleaningitems[5]=q;  break;  }  default:  break;  }  }  };  //---------------------------------------------------------  //ROOM CLASS  //---------------------------------------------------------  class room  {  //--------------attributes---------------  string typ\_rm;  bool clean;  bool book;  float price;  public:  //-------------constructors---------------  room()  {  }  //--------------member functions---------------  //setter  void set\_price(float price)  {  this->price=price;  }  void set\_booking(bool book)  {  this->book=book;  }  void set\_cleaningstatus(bool clean)  {  this->clean;  }  void set\_typofroom(string typ\_rm)  {  this->typ\_rm=typ\_rm;  }  //getter  bool get\_booking()  {  return book;  }  bool get\_cleanstatus()  {  return clean;  }  string get\_typofroom()  {  return typ\_rm;  }  float get\_price()  {  return price;  }  };  //---------------------------------------------------------  //CUSTOMER CLASS  //---------------------------------------------------------  class customer  {  //--------attributes----------  string nm;  string adrs;  int age=0;  char gen;  int phn\_no;  int CNIC;  string email;  string marital\_sts;  float weight;  string blood\_typ;  float height;  public:  //---------constructor-----------  customer()  {  }  //-----------member functions----------  //SETTERS  int get\_id()  {return CNIC;}  void set\_nm(string nm)  {  this->nm=nm;  }  void set\_age(int age)  {  this->age=age;  }  void set\_adrs(string adrs)  {  this->adrs=adrs;  }  void set\_gen(char gen)  {  this->gen=gen;  }  void set\_phn\_no(long int phn\_no)  {  this->phn\_no=phn\_no;  }  void set\_email(string email)  {  this->email=email;  }  void set\_blood\_typ(string blood\_typ)  {  this->blood\_typ=blood\_typ;  }  void set\_weight(float weight)  {  this->weight=weight;  }  void set\_height(float height)  {  this->height=height;  }  void set\_marital\_sts(string marital\_sts)  {  this->marital\_sts=marital\_sts;  }  //-------------------------  //GETTERS  string get\_nm( )  {  return nm;  }  int get\_age()  {  return age;  }  string get\_adrs()  {  return adrs;  }  char get\_gen()  {  return gen;  }  long int get\_phn\_no()  {  return phn\_no;  }  string get\_email()  {  return email;  }  string get\_blood\_typ()  {  return blood\_typ;  }  float get\_weight()  {  return weight;  }  float get\_height()  {  return height;  }  string get\_marital\_sts()  {  return marital\_sts;  }  //----------------------------  //INPUT FUNCTION  void input\_hotel()  {  cout<<"enter name"<<endl;  cin>>nm;  cout<<"enter CNIC";  cin>>CNIC;  cout<<"enter gender"<<endl;  cin>>gen;  cout<<"enter address"<<endl;  cin>>adrs;  cout<<"enter email"<<endl;  cin>>email;    cout<<"enter PHONE NO."<<endl;  cin>>phn\_no;  cout<<"enter marital status ('Y' for married 'N' for not married)"<<endl;  cin>>marital\_sts;  }  void input\_hospital()  {  cout<<"enter name"<<endl;  cin>>nm;  cout<<"enter CNIC";  cin>>CNIC;  cout<<"enter gender"<<endl;  cin>>gen;  cout<<"enter address"<<endl;  cin>>adrs;  cout<<"enter email"<<endl;  cin>>email;    cout<<"enter PHONE NO."<<endl;  cin>>phn\_no;  cout<<"enter marital status ('Y' for married 'N' for not married)"<<endl;  cin>>marital\_sts;  cout<<"enter your blood type"<<endl;  cin>>blood\_typ;  cout<<"enter your height"<<endl;  cin>>height;  cout<<"enter your weight"<<endl;  cin>>weight;  }  //----------------------------  //DISPLAY FUNCTION  void display\_hotel()  {  cout<<"NAME: "<<nm<<endl;  cout<<"CNIC: "<<CNIC<<endl;  cout<<"GENDER: "<<gen<<endl;  cout<<"AGE: "<<age<<endl;  cout<<"ADDRESS: "<<adrs<<endl;  cout<<"PHONE NO.: "<<phn\_no<<endl;  cout<<"EMAIL: "<<email<<endl;  cout<<"MARITAL STATUS: "<<marital\_sts<<endl;  }  void display\_hospital()  {  cout<<"NAME: "<<nm<<endl;  cout<<"CNIC: "<<CNIC<<endl;  cout<<"GENDER: "<<gen<<endl;  cout<<"AGE: "<<age<<endl;  cout<<"ADDRESS: "<<adrs<<endl;  cout<<"PHONE NO.: "<<phn\_no<<endl;  cout<<"EMAIL: "<<email<<endl;  cout<<"MARITAL STATUS: "<<marital\_sts<<endl;  cout<<"HEIGHT: "<<height<<endl;  cout<<"WEIGHT: "<<weight<<endl;  cout<<"BLOOD TYPE: "<<blood\_typ<<endl;  }  };  //---------------------------------------------------------  //CUSTOMER CLASS  //---------------------------------------------------------  class parking{  int no\_of\_amb, no\_of\_cars, spots\_cars, spots\_amb;  bool check;  public:  //constructer  parking(){  check=false;  }  //setters and getters  bool getcheck(){  return true;  }  //function  void inputdata(){  cout<<"-------------------TAKING INPUTS FOR PARKING----------------"<< endl;  cout<<"enter the total number of spots for ambulances ";  cin>>spots\_amb;  cout<<"enter the total number of spots for cars ";  cin>>spots\_cars;  } //from CEO  //function  void inputdata2(){  cout<<"-------------------TAKING INPUTS FOR PARKING----------------"<< endl;  cout<<"enter the total number of spots for cars ";  cin>>spots\_cars;  }  void working (){  char cha;  cout<<"--------PARKING INFORMATION----------"<<endl;  if(getcheck()==1){  cout<<"Is the vehicle ambulance or a car? (a for ambulance and c for car) " << endl;  cin>>cha;  if(cha=='a'){  cout<<"how many ambulances do you want to park? " << endl;  cin>>no\_of\_amb;  if(no\_of\_amb < spots\_amb ){  cout<<"you can park your ambulance"<<endl;  spots\_amb=spots\_amb-no\_of\_amb;  }  else {  cout<<"you cant park your ambulance"<<endl;  }  }  else if(cha=='c'){  cout<<"how many cars do you want to park? " << endl;  cin>>no\_of\_cars;  if(no\_of\_cars < spots\_cars ){  cout<<"you can park your car"<<endl;  spots\_cars=spots\_cars-no\_of\_cars;  }  else {  cout<<"you cant park your car"<<endl;  }  }  }  else {  cout<<"there are no spots available currently "<<endl;  }  }  void working2 (){  char cha;  cout<<"--------PARKING INFORMATION----------"<<endl;  if(getcheck()==1)  {  cout<<"how many cars do you want to park? " << endl;  cin>>no\_of\_cars;  if(no\_of\_cars < spots\_cars ){  cout<<"you can park your car"<<endl;  spots\_cars=spots\_cars-no\_of\_cars;  }  else {  cout<<"you cant park your car"<<endl;  }  }  else {  cout<<"there are no spots available currently "<<endl;  }    }  };  //----------------------------------------------------//  //-----------------RECEPTIONIST-----------------//  //--------------------------------------------------//  class receptionist:protected employee  {  //WRAPPER: ROOM  //WRAPPER CLEANER  //WRAPPER: CHEF  room r;  chef ce;  cleaner c1;  static float salary;  finance f;  public:  receptionist()  {}  receptionist(int i):employee(i)  {}  void input()  {  cout<<"Enter salary of receptionsinst"<<endl;  cin>>salary;  f.subinbudget(salary);  }  void workfunction();//filing  void workfunction2();//filing  void filing(customer c)  {  ofstream f("Hospitalinfo.txt",ios::app);  f<<c.get\_nm()<<" ";  f<<c.get\_id()<<" ";  f<<c.get\_adrs()<<" ";  f<<c.get\_email()<<" ";  f<<c.get\_phn\_no()<<" ";  f<<c.get\_age()<<" ";  f<<c.get\_blood\_typ()<<" ";  f<<c.get\_height()<<" ";  f<<c.get\_weight()<<" ";  f<<c.get\_gen()<<" ";  f<<c.get\_marital\_sts()<<endl;  f.close();    }  void filing2(customer c)//hotel  {  ofstream f("Hotelinfo.txt",ios::app);  f<<c.get\_nm()<<" ";  f<<c.get\_id()<<" ";  f<<c.get\_adrs()<<" ";  f<<c.get\_email()<<" ";  f<<c.get\_phn\_no()<<" ";  f<<c.get\_age()<<" ";  f<<c.get\_marital\_sts()<<endl;  f.close();    }  void callingcleaner() // calling cleaner on customer demands  {  c1.customercall();  }  void clean()// as soon as customer checks out aotmatically call this function  {  c1.workfunction();  }  void orderfood()// customer orders food  {  ce.workfunction();  }  void bookaroom()  {  r.set\_booking(1);  }  void setsal(float salary)  {  this->salary=salary;  }  float getsal()  {  return salary;  }  };  float receptionist::salary=0;  void receptionist::workfunction()  {  string nm;  string adrs;  int age;  char gen;  int phn\_no;  int CNIC;  string email;  string marital\_sts;  float weight;  string blood\_typ;  float height;  ifstream f2;  f2.open("Hospitalinfo.txt");    int d;  cout<<"enter id;";  cin>>d;    while(f2>>nm>>CNIC>>adrs>>email>>phn\_no>>age>>blood\_typ>>height>>weight>>gen>>marital\_sts)  {  if(CNIC==d)  {  cout<<"NAME: "<<nm<<endl;  cout<<"CNIC: "<<CNIC<<endl;  cout<<"GENDER: "<<gen<<endl;  cout<<"AGE: "<<age<<endl;  cout<<"ADDRESS: "<<adrs<<endl;  cout<<"PHONE NO.: "<<phn\_no<<endl;  cout<<"EMAIL: "<<email<<endl;  cout<<"MARITAL STATUS: "<<marital\_sts<<endl;  cout<<"HEIGHT: "<<height<<endl;  cout<<"WEIGHT: "<<weight<<endl;  cout<<"BLOOD TYPE: "<<blood\_typ<<endl;  break;  }  else  {  cout<<"not avaialable";  break;  }    }  f2.close();  }  void receptionist::workfunction2()  {  string nm;  string adrs;  int age=0;  char gen;  int phn\_no;  int CNIC;  string email;  string marital\_sts;  ifstream f2;  f2.open("Hotelinfo.txt");    int d;  cout<<"enter id;";  cin>>d;    while(f2>>nm>>CNIC>>adrs>>email>>phn\_no>>age>>gen>>marital\_sts)  {  if(CNIC==d)  {  cout<<"NAME: "<<nm<<endl;  cout<<"CNIC: "<<CNIC<<endl;  cout<<"GENDER: "<<gen<<endl;  cout<<"AGE: "<<age<<endl;  cout<<"ADDRESS: "<<adrs<<endl;  cout<<"PHONE NO.: "<<phn\_no<<endl;  cout<<"EMAIL: "<<email<<endl;  cout<<"MARITAL STATUS: "<<marital\_sts<<endl;  }  else{  cout<<"not avaialable";  }    }  f2.close();  }  //---------------------------------------------------------  //HOTEL CLASS  //---------------------------------------------------------  class hotel  {  //aaray of objects  customer \*cus;  room \*rm;  shopkeeper shop;  chef chf;  cleaner clean;  receptionist \*rec;  int no\_c,no\_r,no\_rec;  finance f;  int choice;  parking p;  float salary;  float bill=0.0,t\_bill=0.0;  float set,pay;  public:  //wrapper functions  //-----------------------  //-------------customer wrapper functions-----------------  void input\_nof\_employees() //for ceo to enter no.of receptionists  {  float top;  cout<<"enter the no.of receptionests"<<endl;  cin>>no\_rec;  cout<<"enter the salary of your receptionists"<<endl;  cin>>salary;  rec[0].setsal(salary);  top=no\_rec\*salary;  f.subinbudget(top);  cout<<"enter the no.of rooms"<<endl;  cin>>no\_r;    rm= new room [no\_r];  rec= new receptionist [no\_rec];  }  void enter\_nof\_customers()  {  cout<<"enter the no.of customers"<<endl;  cin>>no\_c;  cus=new customer [no\_c];  }  void display\_customer()  {  for(int i=0;i<no\_c;i++)  {  cus[i].display\_hotel();  }    }  //-------------------------------------------------------------------------------------------//  // CEO INPUTS //  //------------------------------------------------------------------------------------------//  void ceo\_input()  {  //CALL WRAPPER THRO MANAGER  cout<<"enter your choices from the following"<<endl<<  "1. Shop + parking"<<endl  <<"2.Chef + parking"<<endl  <<"3. Shop + chef"<<endl  <<"4.NO additional requirements needed"<<endl  <<"5.parking"<<endl;  cin>>choice;  if(choice==1)  {    cout<<"enter shop inputs"<<endl;  shop.input();  cout<<"Taking inputs for parking "<<endl;  p.inputdata2();  cout<<"Taking inputs for cleaner"<<endl;  clean.input();  }  else if(choice==2)  {  cout<<"enter chef inputs"<<endl;  chf.input();  cout<<"enter parking inputs"<<endl;  p.inputdata2();  cout<<"Taking inputs for cleaner"<<endl;  clean.input();  }  else if(choice==3)  {  cout<<"enter shop inputs"<<endl;  shop.input();  cout<<"enter chef inputs"<<endl;  chf.input();  cout<<"Taking inputs for cleaner"<<endl;  clean.input();  }  else if(choice==4)  {  cout<<"Taking inputs for cleaner"<<endl;  clean.input();  }  else if(choice==5)  {  cout<<"enter parking inputs"<<endl;  p.inputdata2();  cout<<"Taking inputs for cleaner"<<endl;  clean.input();  }  input\_nof\_employees();  string typ;  float price;  for(int j=0;j<no\_r;j++)  {  cout<<"enter the type of rooms you have for your each room respectively"<<endl;  cin>>typ;  rm[j].set\_typofroom(typ);  cout<<"enter the price of this room"<<endl;  cin>>price;  rm[j].set\_price(price);  }  }  //-------------------------------------------------------------------------------------------//  // FILING //  //------------------------------------------------------------------------------------------//  void linearsearch()  {  rec[0].workfunction2();  }  //-------------------------------------------------------------------------------------------//  // MODIFICATION //  //------------------------------------------------------------------------------------------//  void modification() //for ceo modification through manager  {  int m,loop=1;  while(loop)  {  cout<<"Choose what you want to modify from below"<<endl;  cout<<"1.CHEF "<<endl;  cout<<"2.CLEANER"<<endl;  cout<<"3.SHOP"<<endl;  cout<<"4.RECEPTIONIST"<<endl;  cout<<"5.DONE WITH MODIFICATION"<<endl;  cin>>m;  switch (m)  {  case 1:  {int nl,l;  cout<<"Enter 1. to change salary\nEnter 2. to make changes in chef items"<<endl;  cin>>l;  if(l==1)  {  float s;  cout<<"Enter new salary "<<endl;  cin>>s;  chf.setsal(s);  }  else if(l==2)  {  cout<<"Enter number of Dishes that should be in menue"<<endl;  chf.input();  }  break;  }  case 2:  {int nl,l;  cout<<"Enter 1. to change salary\nEnter 2. to make changes in cleaner items"<<endl;  cin>>l;  if(l==1)  {  float s;  cout<<"Enter new salary "<<endl;  cin>>s;  clean.setsal(s);  }  else if(l==2)  {  clean.input();  }    break;  }  case 3:  {  int nl,l;  cout<<"Enter 1. to change salary\nEnter 2. to make changes in cleaner items"<<endl;  cin>>l;  if(l==1)  {  float s;  cout<<"Enter new salary "<<endl;  cin>>s;  shop.setsal(s);  }  else if(l==2)  {  shop.input();  }  break;  }  case 4:  {  float s,l;  cout<<"Enter new salary for receptionist"<<endl;  cin>>s;  rec[0].setsal(s);  break;  }  case 5:  {  loop=0;  break;  }  default:  break;  }  }  }  //-------------checkout function---------------------//  void checkout()  {  float set;  cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Thank you for your stay\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;  t\_bill=bill+chf.get\_bill();  cout<<"The Total bill for your stay: "<<bill<<endl;  cout<<"The bill for your food orders: "<<chf.get\_bill()<<endl;  chf.set\_bill(0.0);  cout<<" total bill: "<<t\_bill<<endl<<"enter your payment"<<endl;  cin>>pay;  set=pay-t\_bill;  cout<<" amount returned: "<<set<<endl;  f.setbudget(t\_bill);    }  //-------------------------------------------------------------------------------------------//  // WORK FUNCTION //  //------------------------------------------------------------------------------------------//  void work()  {    int j=1,ans1,rm\_choice,nof\_days,rent;    //QUE LOGIC START  if(choice==1) //parking+shop  {  for(int i=1;i<=no\_c;i++) //loop for customers  {  cout<<"-------CUSTOMER#"<<i<<"----------"<<endl;  if(i%(no\_rec)==0)  {  j=1;//LOGIC  p.working2() ;  cus[i-1].input\_hotel();  rec[j-1].filing2(cus[i-1]);  cout<<"enter the type of room you wish to book your options are as following"<<endl;  for(int i=0;i<no\_r;i++)  {  cout<<i+1<<" "<<rm[i].get\_typofroom()<<endl;  }  cin>>rm\_choice;  rm\_choice=rm\_choice-1;  rm[rm\_choice].set\_booking(true);  cout<<"enter your no.of days for stay"<<endl;  cin>>nof\_days;  bill=nof\_days\*rm[rm\_choice].get\_price();    do{  cout<<"enter the number for the service you wish to access"<<endl  <<"1.Room Service (cleaning) "<<endl  <<"2.Visit Shop"<<endl  <<"0.to leave"<<endl;  cin>>ans1;  switch(ans1)  {  case 1:  {  clean.customercall();  }  break;  case 2:  {  shop.workfunction();  }  break;  }    }while(ans1!=0);    checkout();  clean.workfunction();  j++;    }else  {  p.working2() ;  cus[i-1].input\_hotel();  rec[j-1].filing2(cus[i-1]);  cout<<"enter the type of room you wish to book your options are as following"<<endl;  for(int i=0;i<no\_r;i++)  {  cout<<i+1<<" "<<rm[i].get\_typofroom()<<endl;  }  cin>>rm\_choice;  rm\_choice=rm\_choice-1;  rm[rm\_choice].set\_booking(true);  cout<<"enter your no.of days for stay";  cin>>nof\_days;  bill=nof\_days\*rm[rm\_choice].get\_price();  //---------send bill to finance--------------  cus[i-1].input\_hotel();  cout<<"enter the number for the service you wish to access"<<endl  <<"1.Room Service (cleaning) "<<endl  <<"2.Food service"<<endl  <<"3.Visit Shop"<<endl;  cin>>ans1;  do{  cout<<"enter the number for the service you wish to access"<<endl  <<"1.Room Service (cleaning) "<<endl  <<"2.Visit Shop"<<endl  <<"0.to leave"<<endl;  cin>>ans1;  switch(ans1)  {  case 1:  {  clean.customercall();  }  break;  case 2:  {  shop.workfunction();  }  break;  }    }while(ans1!=0);      checkout();  clean.workfunction();  j++;    }  }  }  else if(choice==2)// CHEF+PARKING  {  for(int i=1;i<=no\_c;i++) //loop for customers  {  if(i%(no\_rec)==0) //TO SEND CUSTOMER AGAIN TO FIRST RECPTIONIST  {  j=1;//LOGIC  p.working2();  cus[i-1].input\_hotel();  rec[j-1].filing2(cus[i-1]);  cout<<"enter the type of room you wish to book your options are as following"<<endl;  for(int i=0;i<no\_r;i++)  {  cout<<i+1<<" "<<rm[i].get\_typofroom()<<endl;  }  cin>>rm\_choice;  rm\_choice=rm\_choice-1;  rm[rm\_choice].set\_booking(true);  cout<<"enter your no.of days for stay"<<endl;  cin>>nof\_days;  bill=nof\_days\*rm[rm\_choice].get\_price();  //------------send bill to finance--------------  //ENTER WHILE LOOP  do{  cout<<"enter the number for the service you wish to access"<<endl  <<"1.Room Service (cleaning) "<<endl  <<"2.Food service"<<endl  <<"0.To leave"<<endl;    cin>>ans1;  switch(ans1)  {  case 1:  {  clean.customercall();  }  break;  case 2:  {  chf.workfunction();  }  break;    }  }while(ans1!=0);    checkout();  clean.workfunction();    j++;    }else  {  p.working2();  cus[i-1].input\_hotel();  rec[j-1].filing2(cus[i-1]);  cout<<"enter the type of room you wish to book your options are as following"<<endl;  for(int i=0;i<no\_r;i++)  {  cout<<i+1<<" "<<rm[i].get\_typofroom()<<endl;  }  cin>>rm\_choice;  rm\_choice=rm\_choice-1;  rm[rm\_choice].set\_booking(true);  cout<<"enter your no.of days for stay";  cin>>nof\_days;  bill=nof\_days\*rm[rm\_choice].get\_price();  //---------send bill to finance--------------  do{  cout<<"enter the number for the service you wish to access"<<endl  <<"1.Room Service (cleaning) "<<endl  <<"2.Food service"<<endl  <<"0.To leave"<<endl;    cin>>ans1;  switch(ans1)  {  case 1:  {  clean.customercall();  }  break;  case 2:  {  chf.workfunction();  }  break;    }  }while(ans1!=0);    checkout();  clean.workfunction();    j++;    }  }  }  else if(choice==3) //SHOP+CHEF  {  for(int i=1;i<=no\_c;i++) //loop for customers  {  if(i%(no\_rec)==0) //TO SEND CUSTOMER AGAIN TO FIRST RECPTIONIST  {  j=1;//LOGIC  cus[i-1].input\_hotel();  rec[j-1].filing2(cus[i-1]);  cout<<"enter the type of room you wish to book your options are as following"<<endl;  for(int i=0;i<no\_r;i++)  {  cout<<i+1<<" "<<rm[i].get\_typofroom()<<endl;  }  cin>>rm\_choice;  rm\_choice=rm\_choice-1;  rm[rm\_choice].set\_booking(true);  cout<<"enter your no.of days for stay";  cin>>nof\_days;  bill=nof\_days\*rm[rm\_choice].get\_price();  //------------send bill to finance--------------  //ENTER WHILE LOOP  do{  cout<<"enter the number for the service you wish to access"<<endl  <<"1.Room Service (cleaning) "<<endl  <<"2.Food service"<<endl  <<"3.Visit Shop"<<endl  <<"0.To leave"<<endl;  cin>>ans1;    switch(ans1)  {  case 1:  {  clean.customercall();  }  break;  case 2:  {  chf.workfunction();  }  break;  case 3:  {  shop.workfunction();  }  break;  }  }while(ans1!=0);      checkout();  clean.workfunction();  j++;    }  else  {  cus[i-1].input\_hotel();  rec[j-1].filing2(cus[i-1]);  cout<<"enter the type of room you wish to book your options are as following"<<endl;  for(int i=0;i<no\_r;i++)  {  cout<<i+1<<" "<<rm[i].get\_typofroom()<<endl;  }  cin>>rm\_choice;  rm\_choice=rm\_choice-1;  rm[rm\_choice].set\_booking(true);  cout<<"enter your no.of days for stay";  cin>>nof\_days;  bill=nof\_days\*rm[rm\_choice].get\_price();  //---------send bill to finance--------------  do{  cout<<"enter the number for the service you wish to access"<<endl  <<"1.Room Service (cleaning) "<<endl  <<"2.Food service"<<endl  <<"3.Visit Shop"<<endl  <<"0.To leave"<<endl;  cin>>ans1;    switch(ans1)  {  case 1:  {  clean.customercall();  }  break;  case 2:  {  chf.workfunction();  }  break;  case 3:  {  shop.workfunction();  }  break;  }  }while(ans1!=0);  checkout();  clean.workfunction();  j++;    }  }  }  else if(choice==4) //nothing  {  for(int i=1;i<=no\_c;i++) //loop for customers  {  if(i%(no\_rec)==0) //TO SEND CUSTOMER AGAIN TO FIRST RECPTIONIST  {  j=1;//LOGIC  cus[i-1].input\_hotel();  rec[j-1].filing2(cus[i-1]);  cout<<"enter the type of room you wish to book your options are as following"<<endl;  for(int i=0;i<no\_r;i++)  {  cout<<i+1<<" "<<rm[i].get\_typofroom()<<endl;  }  cin>>rm\_choice;  rm\_choice=rm\_choice-1;  rm[rm\_choice].set\_booking(true);  cout<<"enter your no.of days for stay";  cin>>nof\_days;  bill=nof\_days\*rm[rm\_choice].get\_price();  //------------send bill to finance--------------  //ENTER WHILE LOOP  do{    cout<<"enter 1 for calling room service else enter 0"<<endl;  cin>>ans1;  if(ans1==1)  {  rec[j-1].callingcleaner();  }  }while(ans1!=0);    checkout();  clean.workfunction();  j++;    }else  {  cus[i-1].input\_hotel();  rec[j-1].workfunction();  cout<<"enter the type of room you wish to book your options are as following"<<endl;  for(int i=0;i<no\_r;i++)  {  cout<<i+1<<" "<<rm[i].get\_typofroom()<<endl;  }  cin>>rm\_choice;  rm\_choice=rm\_choice-1;  rm[rm\_choice].set\_booking(true);  cout<<"enter your no.of days for stay";  cin>>nof\_days;  bill=nof\_days\*rm[rm\_choice].get\_price();  //---------send bill to finance--------------    do{    cout<<"enter 1 for calling room service else enter 0"<<endl;  cin>>ans1;  if(ans1==1)  {  rec[j-1].callingcleaner();  }  }while(ans1!=0);    checkout();  clean.workfunction();  j++;    }  }  }  else if(choice==5)//PARKING  {  for(int i=1;i<=no\_c;i++) //loop for customers  {  if(i%(no\_rec)==0) //TO SEND CUSTOMER AGAIN TO FIRST RECPTIONIST  {  j=1;//LOGIC  p.working2();  cus[i-1].input\_hotel();  rec[j-1].filing2(cus[i-1]);  cout<<"enter the type of room you wish to book your options are as following"<<endl;  for(int i=0;i<no\_r;i++)  {  cout<<i+1<<" "<<rm[i].get\_typofroom()<<endl;  }  cin>>rm\_choice;  rm\_choice=rm\_choice-1;  rm[rm\_choice].set\_booking(true);  cout<<"enter your no.of days for stay";  cin>>nof\_days;  bill=nof\_days\*rm[rm\_choice].get\_price();  //ENTER WHILE LOOP  do{  cout<<"enter the number for the service you wish to access"<<endl  <<"1.Room Service (cleaning) "<<endl  <<"0.To leave"<<endl;  cin>>ans1;  switch(ans1)  {  case 1:  {  clean.customercall();  }  break;  //CHNAGES  }  }while(ans1!=0);          checkout();  clean.workfunction();  j++;    }else  {  p.working2();  cus[i-1].input\_hotel();  rec[j-1].filing2(cus[i-1]);  cout<<"enter the type of room you wish to book your options are as following"<<endl;  for(int i=0;i<no\_r;i++)  {  cout<<i+1<<" "<<rm[i].get\_typofroom()<<endl;  }  cin>>rm\_choice;  rm[rm\_choice].set\_booking(true);  cout<<"enter your no.of days for stay";  cin>>nof\_days;  bill=nof\_days\*rm[rm\_choice].get\_price();  //---------send bill to finance--------------    //ENTER WHILE LOOP  do{  cout<<"enter the number for the service you wish to access"<<endl  <<"1.Room Service (cleaning) "<<endl  <<"0.To leave"<<endl;  cin>>ans1;  switch(ans1)  {  case 1:  {  clean.customercall();  }  break;  //CHNAGES  }  }while(ans1!=0);  checkout();  clean.workfunction();  j++;    }        }  }  }  //DESTRUCTORS  ~hotel()  {  delete rec;  delete cus;  delete rm;  }  };  //---------------------------------------------------------  //HOSPITAL CLASS  //---------------------------------------------------------  class hospital  {  //aaray of objects  customer \*custo;  room \*r;  receptionist \*rec;  int no\_c,no\_r,no\_rec,choicess;  char ch,c;  finance fi;  surgeon sur;  ENT\_specialist ent;  gen\_phy gen;  chef che;  cleaner cl;  shopkeeper sh;  parking p;  float salary;  float bill=0, t\_bill=0;  float ent\_sal,gen\_sal,sur\_sal;  float a=0.0,x,y;  float set;  //r[6]={0,1,2,3,4,5,6,}  public:  //wrapper functions  //-----------------------  //-------------customer wrapper functions-----------------  void enter\_customer()  {  cout<<"enter the no.of customers";  cin>>no\_c;  custo=new customer [no\_c];    }  void input\_no\_of\_employee() //ceo  {  cout<<"enter the no.of receptionests"<<endl;  cin>>no\_rec;  cout<<"enter the salary of recepetionsit: "<<endl;  cin>>salary;  cout<<"enter the salary of ENT: "<<endl;  cin>>ent\_sal;  ent.setsal(ent\_sal);  cout<<"enter the salary of surgeon: "<<endl;  cin>>sur\_sal;  sur.setsal(sur\_sal);  cout<<"enter the salary of general physician: "<<endl;  cin>>gen\_sal;  gen.setsal(gen\_sal);  cout<<"enter the no.of rooms";  cin>>no\_r;    r= new room [no\_r];  rec= new receptionist [no\_rec];  }  void display\_customer()  {  for(int i=0;i<no\_c;i++)  {  custo[i].display\_hospital();  }  }  void checkout()  {  float pay;  cout<<"\*thankyou for your stay\*"<<endl;  t\_bill=bill+che.get\_bill();  cout<<"the total bill of room is: "<<bill<<endl;  cout<<"The bill for your food orders "<<che.get\_bill()<<endl;  che.set\_bill(0.0);  cout<<"enter your payment"<<endl;  cin>>pay;  cout<<" total bill: "<<t\_bill<<endl;  set=pay-t\_bill;  cout<<" amount returned: "<<set<<endl;  fi.setbudget(t\_bill);  }  //-------------employee wrapper functions----------------  //--------------------  //shopkeepers wraaper functions  void modification()  {  //calll wrapper function for modify  int m,loop=1;  while(loop)  { cout<<"Choose what you want to modify from below"<<endl;  cout<<"1.CHEF "<<endl;  cout<<"2.CLEANER"<<endl;  cout<<"3.SHOP"<<endl;  cout<<"4.RECEPTIONIST"<<endl;  cout<<"5.DOCTORS"<<endl;  cout<<"6.END"<<endl;  cin>>m;  switch (m)  {  case 1:  {int nl,l;  cout<<"Enter 1. to change salary\nEnter 2. to make changes in chef items"<<endl;  cin>>l;  if(l==1)  {  float s;  cout<<"Enter new salary "<<endl;  cin>>s;  che.setsal(s);  }  else if(l==2)  {  che.input();  }  break;  }  case 2:  {int nl,l;  cout<<"Enter 1. to change salary\nEnter 2. to make changes in cleaner items"<<endl;  cin>>l;  if(l==1)  {  float s;  cout<<"Enter new salary "<<endl;  cin>>s;  cl.setsal(s);  }  else if(l==2)  {  cl.input();  }    break;  }  case 3:  {  int nl,l;  cout<<"Enter 1. to change salary\nEnter 2. to make changes in cleaner items"<<endl;  cin>>l;  if(l==1)  {  float s;  cout<<"Enter new salary "<<endl;  cin>>s;  sh.setsal(s);  }  else if(l==2)  {  sh.input();  }  break;  }  case 4:  {  float s;  cout<<"Enter new salary for receptionist"<<endl;  cin>>s;  rec[0].setsal(s);  break;  }  case 5:  {  int a;  cout<<"Enter 1.ENT SPECIALIST\n2.SURGEON\n3.GENERAL PHYSICIAN"<<endl;  if(a==1)  {  float s;  cout<<"Enter new salary"<<endl;  cin>>s;  ent.setsal(s);  }  else if(a==2)  {  float s;  cout<<"Enter new salary"<<endl;  cin>>s;  sur.setsal(s);  }  else if(a==3)  {  float s;  cout<<"Enter new salary"<<endl;  cin>>s;  gen.setsal(s);  }  break;  }  case 6:  {  loop=0;  break;  }  default:  break;  }  }  }  //-------------------------------------------------------------------------------------------//  // FILING //  //------------------------------------------------------------------------------------------//  void linearsearch()  {  rec[0].workfunction();  }  void ceo\_input()  {  //CALL WRAPPER THRO MANAGER  cout<<"enter your vchoices from yhe following"<<endl<<  "1. Shop + parking"<<endl  <<"2.Chef + parking"<<endl  <<"3. Shop + chef"<<endl  <<"4.NO additional requiremnets needed"<<endl  <<"5.parking"<<endl;  cin>>choicess;  if(choicess==1)  {  cout<<"enter shop inputs"<<endl;  sh.input();  cout<<"Taking inputs for parking "<<endl;  p.inputdata();  cout<<"enter inputs of the cleaner"<<endl;  cl.input();    }  else if(choicess==2)  {  cout<<"enter chef inputs"<<endl;  che.input();  cout<<"enter parking inputs"<<endl;  p.inputdata();  cout<<"enter inputs of the cleaner"<<endl;  cl.input();  }  else if(choicess==3)  {  cout<<"enter shop inputs"<<endl;  sh.input();  cout<<"enter chef inputs"<<endl;  che.input();  cout<<"enter inputs of the cleaner"<<endl;  cl.input();  }  else if(choicess==4)  {  cout<<"enter inputs of the cleaner"<<endl;  cl.input();  }  else if(choicess==5)  {  cout<<"enter parking inputs"<<endl;  p.inputdata();  cout<<"enter inputs of the cleaner"<<endl;  cl.input();  }  input\_no\_of\_employee();  string type;  float price;  for(int i=0; i<no\_r; i++)  {  cout<<"enter the type of rooms you have for your each room respectively. "<<endl;  cin>>type;  r[i].set\_typofroom(type);  cout<<"enter the price of this room too. "<<endl;  cin>>price;  r[i].set\_price(price);  }  }  void calldoc()  {  char ch;  int b,no\_of\_days;    if(choicess==1)  {  for( int i=0; i<no\_c; i++ )  {  cout<<"-------CUSTOMER#"<<i+1<<"----------"<<endl;  p.working();  custo[i].input\_hospital(); //send info to receptionist's funct  rec[0].filing(custo[i]);  cout<<"What type of docter do you want? (e for ENT, s for surgeon and g for general physician) "<<endl;  cin>>c;  if(c=='e'){  ent.workfunction();  a=ent.getsal()+100;  cout<<"your bill for ENT is: "<<a<<endl;  cout<<"Enter your payment"<<endl;  cin>>x;  y=x-a;  cout<<"The amount returned is: "<<y<<endl;  fi.addinbudget(a);    cout<<"do you want to go to the shop? (y for yes)"<<endl;  cin>>ch;    if(ch=='y'){  sh.workfunction();  }  }  else if(c=='s'){  int typ;  sur.workfunction();  //----ROOM BOOK-------  cout<<"what type of room?"<<endl;  for(int i=0; i<no\_r; i++){  cout<<i+1 << " " << r[i].get\_typofroom();  } // room booking will be opt depending on type of disease  cin>>typ; //== execuitibve  typ=typ-1;  r[typ].set\_booking(true);  cout<<"how many days? "<<endl;  cin>>no\_of\_days;  bill=no\_of\_days\*r[typ].get\_price();  //PRINT THE BILL WHILE CHECKING OUT  //----------send bill to finance-----//  fi.setbudget(bill);  int d=1;  while(d){  cout<<"Do you want to avail these services? "<<endl;  cout<<"1. pharmacy, 2.cleaner, 3.leave)"<<endl;  cin>>b;  if(b==1){  sh.workfunction();  }  else if(b==2){  cl.customercall();  //call work funct thru recept  }  else if(b==3){  d=0;  break;  }  }  a=sur.getsal()+100;  cout<<"your bill for surgeon is: "<<a<<endl;  cout<<"Enter your payment"<<endl;  cin>>x;  y=x-a;  cout<<"The amount returned is: "<<y<<endl;  fi.addinbudget(a);  checkout();  cl.workfunction();  }  else if(c=='g'){  gen.workfunction();  a=gen.getsal()+100;  cout<<"your bill for genereal physician is: "<<a<<endl;  cout<<"Enter your payment"<<endl;  cin>>x;  y=x-a;  cout<<"The amount returned is: "<<y<<endl;  fi.addinbudget(a);  cout<<"do you want to go to the shop? (y for yes)"<<endl;  cin>>ch;    if(ch=='y'){  sh.workfunction();  }  }  }  }  else if(choicess==2)  {  for( int i=0; i<no\_c; i++ ){  cout<<"-------CUSTOMER#"<<i+1<<"----------"<<endl;  p.working();  custo[i].input\_hospital();  rec[0].filing(custo[i]);  cout<<"What type of docter do you want? (e for ENT, s for surgeon and g for general physician) "<<endl;  cin>>c;  if(c=='e'){  ent.workfunction();  a=ent.getsal()+100;  cout<<"your bill for ENT is: "<<a<<endl;  cout<<"Enter your payment"<<endl;  cin>>x;  y=x-a;  cout<<"The amount returned is: "<<y<<endl;  fi.addinbudget(a);  }  else if(c=='s'){  int typ;  //----ROOM BOOK-------  sur.workfunction();  cout<<"what type of room?"<<endl;  for(int i=0; i<no\_r; i++){  cout<<i+1 << " " << r[i].get\_typofroom()<<endl;  } // room booking will be opt depending on type of disease  cin>>typ; //== execuitibve  typ=typ-1;  r[typ].set\_booking(true);  cout<<"how many days? "<<endl;  cin>>no\_of\_days;  bill=no\_of\_days\*r[typ].get\_price();    //----------send bill to finance-----//  fi.setbudget(bill);  int d=1;  while(d){  cout<<"Do you want to avail these services? "<<endl;  cout<<"1. chef, 2. cleaner, 3.leave)"<<endl;  cin>>b;  if(b==1){  che.workfunction();  }  else if(b==2){  cl.customercall();  //call work funct thru recept  }  else if(b==3){  d=0;  break;  }  }  a=sur.getsal()+100;  cout<<"your bill for surgeon is: "<<a<<endl;  cout<<"Enter your payment"<<endl;  cin>>x;  y=x-a;  cout<<"The amount returned is: "<<y<<endl;  fi.addinbudget(a);  checkout();  cl.workfunction();  }  else if(c=='g'){    gen.workfunction();  a=gen.getsal()+100;  cout<<"your bill for genereal physician is: "<<a<<endl;  cout<<"Enter your payment"<<endl;  cin>>x;  y=x-a;  cout<<"The amount returned is: "<<y<<endl;  fi.addinbudget(a);  cout<<"do you want to go to the shop? (y for yes)"<<endl;  cin>>ch;    if(ch=='y'){  sh.workfunction();  }    }  }  }  else if(choicess==3){  for( int i=0; i<no\_c; i++ ){  custo[i].input\_hospital();  rec[0].filing(custo[i]);  cout<<"What type of docter do you want? (e for ENT, s for surgeon and g for general physician) "<<endl;  cin>>c;  if(c=='e'){  ent.workfunction();  a=ent.getsal()+100;  cout<<"your bill for ENT is: "<<a<<endl;  cout<<"Enter your payment"<<endl;  cin>>x;  y=x-a;  cout<<"The amount returned is: "<<y<<endl;  fi.addinbudget(a);    cout<<"do you want to go to the shop? (y for yes)"<<endl;  cin>>ch;    if(ch=='y'){  sh.workfunction();  }  }  else if(c=='s'){  int typ;  //----ROOM BOOK-------  sur.workfunction();  cout<<"what type of room?"<<endl;  for(int i=0; i<no\_r; i++){  cout<<i+1<< " " << r[i].get\_typofroom()<<endl;  } // room booking will be opt depending on type of disease  cin>>typ; //== executive  typ=typ-1;  r[typ].set\_booking(true);  cout<<"how many days? "<<endl;  cin>>no\_of\_days;  bill=no\_of\_days\*r[typ].get\_price();    //----------send bill to finance-----//  fi.setbudget(bill);  int d=1;  while(d){  cout<<"Do you want to avail these services? "<<endl;  cout<<"1. chef, 2. pharmacy, 3.cleaner, 4.leave)"<<endl;  cin>>b;  if(b==1){  che.workfunction();  }  else if(b==2){  sh.workfunction();  //call work funct thru recept  }  else if(b==3){  cl.customercall();  }  else if(b==4){  d=0;  break;  }  }  a=sur.getsal()+100;  cout<<"your bill for surgeon is: "<<a<<endl;  cout<<"Enter your payment"<<endl;  cin>>x;  y=x-a;  cout<<"The amount returned is: "<<y<<endl;  fi.addinbudget(a);  checkout();  cl.workfunction();  }  else if(c=='g'){  gen.workfunction();  a=gen.getsal()+100;  cout<<"your bill for genereal physician is: "<<a<<endl;  cout<<"Enter your payment"<<endl;  cin>>x;  y=x-a;  cout<<"The amount returned is: "<<y<<endl;  fi.addinbudget(a);  cout<<"do you want to go to the shop? (y for yes)"<<endl;  cin>>ch;    if(ch=='y'){  sh.workfunction();  }    }  }  }  else if(choicess==4){  for( int i=0; i<no\_c; i++ )  {  cout<<"-------CUSTOMER#"<<i+1<<"----------"<<endl;  custo[i].input\_hospital();  cout<<"What type of docter do you want? (e for ENT, s for surgeon and g for general physician) "<<endl;  cin>>c;  if(c=='e'){  ent.workfunction();  a=ent.getsal()+100;  cout<<"your bill for ENT is: "<<a<<endl;  cout<<"Enter your payment"<<endl;  cin>>x;  y=x-a;  cout<<"The amount returned is: "<<y<<endl;  fi.addinbudget(a);    }  else if(c=='s'){  int typ;  //----ROOM BOOK-------  sur.workfunction();  cout<<"what type of room?"<<endl;  for(int i=0; i<no\_r; i++){  cout<<i+1<< " " << r[i].get\_typofroom();  } // room booking will be opt depending on type of disease  cin>>typ; //== execuitibve  typ=typ-1;  r[typ].set\_booking(true);  cout<<"how many days? "<<endl;  cin>>no\_of\_days;  bill=no\_of\_days\*r[typ].get\_price();    //----------send bill to finance-----//  fi.setbudget(bill);  int d=1;  while(d){  cout<<"Do you want to avail these services? "<<endl;  cout<<"1.cleaner, 2.leave)"<<endl;  cin>>b;  if(b==1){  cl.customercall();  }  else if(b==2){  d=0;  break;  }  }  a=sur.getsal()+100;  cout<<"your bill for surgeon is: "<<a<<endl;  cout<<"Enter your payment"<<endl;  cin>>x;  y=x-a;  cout<<"The amount returned is: "<<y<<endl;  fi.addinbudget(a);  checkout();  cl.workfunction();  }  else if(c=='g'){  gen.workfunction();  a=gen.getsal()+100;  cout<<"your bill for genereal physician is: "<<a<<endl;  cout<<"Enter your payment"<<endl;  cin>>x;  y=x-a;  cout<<"The amount returned is: "<<y<<endl;  fi.addinbudget(a);    }  }  }  else if(choicess==5){  for( int i=0; i<no\_c; i++ )  { cout<<"-------CUSTOMER#"<<i+1<<"----------"<<endl;  p.working();  custo[i].input\_hospital();  rec[0].filing(custo[i]);  cout<<"What type of docter do you want? (e for ENT, s for surgeon and g for general physician) "<<endl;  cin>>c;  if(c=='e'){  ent.workfunction();  a=ent.getsal()+100;  cout<<"your bill for ENT is: "<<a<<endl;  cout<<"Enter your payment"<<endl;  cin>>x;  y=x-a;  cout<<"The amount returned is: "<<y<<endl;  fi.addinbudget(a);  }  else if(c=='s'){  int typ;  //----ROOM BOOK-------  sur.workfunction();  cout<<"what type of room?"<<endl;  for(int i=0; i<no\_r; i++){  cout<<i+1 << " " << r[i].get\_typofroom();  } // room booking will be opt depending on type of disease  cin>>typ; //== execuitibve  typ=typ-1;  r[typ].set\_booking(true);  cout<<"how many days? "<<endl;  cin>>no\_of\_days;  bill=no\_of\_days\*r[typ].get\_price();    //----------send bill to finance-----//  fi.setbudget(bill);  int d=1;  while(d){  cout<<"Do you want to avail these services? "<<endl;  cout<<"1.cleaner, 2.leave)"<<endl;  cin>>b;  if(b==1){  cl.customercall();  }  else if(b==2){  d=0;  break;  }  }  a=sur.getsal()+100;  cout<<"your bill for surgeon is: "<<a<<endl;  cout<<"Enter your payment"<<endl;  cin>>x;  y=x-a;  cout<<"The amount returned is: "<<y<<endl;  fi.addinbudget(a);  checkout();  cl.workfunction();  }  else if(c=='g'){  gen.workfunction();  a=gen.getsal()+100;  cout<<"your bill for genereal physician is: "<<a<<endl;  cout<<"Enter your payment"<<endl;  cin>>x;  y=x-a;  cout<<"The amount returned is: "<<y<<endl;  fi.addinbudget(a);  }  }  }  }  };  //-------------------------------------------------------------------------------------------//  // MAIN FUNCTION //  //------------------------------------------------------------------------------------------//  int main()  {  hospital hos;  int d,k;  char ch;  cout<<"------------------------------WELCOME-----------------------------"<<endl;  cout<<"which management system do u desire to create? (1. hospital, 2. hotel) "<<endl;  cin>>d;  switch(d)  {  case 1:  {  hospital hos;  cout<<"CEO please input requirements for the hospital management system"<<endl;  hos.ceo\_input();  cout<<"Enter how many customers can your hospital cater"<<endl;  hos.enter\_customer();    cout<<"Is there anything u want to modify? (y for yes and n for no) "<<endl;  cin>>ch;  if(ch=='y'){  hos.modification();  }  else {  cout<<"Your management system is in progress. "<<endl;  }    system("CLS");  cout<<flush;  cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*YOUR MANAGEMENT SYSTEM HAS BEEN CREATED\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;  cout<<"You have reached hospital "<<endl;  hos.calldoc();  cout<<"Enter 1.To Search for a specific customer 2.To leave"<<endl;  cin>>k;  if(k==1)  {  hos.linearsearch();  }  else  {  cout<<"program has ended"<<endl;  break;  }  break;  }  case 2:  {  hotel ho;  cout<<"Now CEO will input requirements for their hotel management system"<<endl;  ho.ceo\_input();  cout<<"Enter how many customers can your hotel cater"<<endl;  ho.enter\_nof\_customers();  cout<<"Is there anything u want to modify? (y for yes and n for no) "<<endl;  cin>>ch;  if(ch=='y'){  ho.modification();  }  else {  cout<<"Your management system is in progress. "<<endl;  }  system("CLS");  cout<<flush;  cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*YOUR MANAGEMENT SYSTEM HAS BEEN CREATED\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;    cout<<"You have reached Hotel "<<endl;  ho.work();    // cout<<"Enter 1.To Search for a specific customer 2.To leave"<<endl;  // cin>>k;  // if(k==1)  // {  // ho.linearsearch();  // }  // else  // {  // cout<<"program has ended"<<endl;  // break;  // }    break;  }  }  return 0;  } | |  |
|  |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  | |  |  |
|  |  | |  |
|  |  | |
|  | | | |

|  |  |  |
| --- | --- | --- |
|  | | |
|  |  |  |
|  |  |
|  |  |
|  | | |
|  | | |