#include<iostream>

#include<vector>

using namespace std;

class Product

{

string name;

float price;

public:

Product(string, float);

void setName(string);

void setPrice(float);

string getName() const;

float getPrice() const;

virtual float calculateCost() = 0;

};

class Electronics : public Product

{

int quan;

public:

Electronics(string,float,int);

void setQuan(int);

int getQuan() const;

float calculateCost();

};

class Clothing : public Product

{

int quan;

public:

Clothing(string,float,int);

void setQuan(int);

int getQuan() const;

float calculateCost();

};

class ShoppingCart

{

vector<Product\*> prod;

public:

void addProd(Product \* p);

float totalCost();

int quantProd(string) const;

void placeOrder();

int totalItems() const;

};

Product::Product(string name="",float price=0)

{

this->name=name;

if(price>=0)

this->price=price;

}

void Product::setName(string name)

{

this->name=name;

}

void Product::setPrice(float price)

{

if(price>=0)

this->price=price;

}

string Product::getName() const

{

return name;

}

float Product::getPrice() const

{

return price;

}

Electronics::Electronics(string name="",float price=0,int items=0): Product(name,price)

{

if(items>=0)

this->quan=items;

}

void Electronics::setQuan(int items)

{

if(items>=0)

this->quan=items;

}

int Electronics::getQuan() const

{

return quan;

}

float Electronics::calculateCost()

{

return quan\*getPrice();

}

Clothing::Clothing(string name="",float price=0,int items=0): Product(name,price)

{

if(items>=0)

this->quan=items;

}

void Clothing::setQuan(int items)

{

if(items>=0)

this->quan=items;

}

int Clothing::getQuan() const

{

return quan;

}

float Clothing::calculateCost()

{

return quan\*getPrice();

}

void ShoppingCart::addProd(Product \*p)

{

prod.push\_back(p);

}

float ShoppingCart::totalCost()

{

float cost=0;

for(auto p:prod)

{

cost+= p->calculateCost();

}

return cost;

}

int ShoppingCart::quantProd(string name) const

{

for(auto p:prod)

{

//if(p->getName()==name)

//return p->getQuan();

}

return -1;

}

void ShoppingCart::placeOrder()

{

int n=prod.size();

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

prod.pop\_back();

}

int ShoppingCart::totalItems() const

{

return prod.size();

}

int main()

{

ShoppingCart \* sc=new ShoppingCart();

int ch;

while(ch!=6)

{

cout<<endl<<"Shopping Cart:-"<<endl<<"\t1. Add Products"<<endl<<"\t2. Qunatity of Products"<<endl<<"\t3. Total cost"<<endl<<"\t4. Total Items"<<endl<<"\t5. Place Order"<<endl<<"\t6. exit"<<endl<<"choose products:- ";

cin>>ch;

int ch\_pro=0;

string name;

float price;

int items;

int quan;

switch(ch)

{

case 1:

cout<<endl<<"Products:-"<<endl<<"\t1. Electronics"<<endl<<"\t2. Clothing"<<endl<<"\t3. home."<<endl<<"choose products:- ";

cin>>ch\_pro;

if(ch\_pro==1)

{

cout<<"Name:- ";

cin>>name;

cout<<"Price:- ";

cin>>price;

cout<<"Items:- ";

cin>>items;

Product \*e1 = new Electronics(name,price,items);

sc->addProd(e1);

cout<<"Product added to Cart"<<endl;

}

else if(ch\_pro==2)

{

cout<<"Name:- ";

cin>>name;

cout<<"Price:- ";

cin>>price;

cout<<"Items:- ";

cin>>items;

Product \*c1 = new Clothing(name,price,items);

sc->addProd(c1);

cout<<"Product added to Cart"<<endl;

}

break;

case 2:

cout<<"Name:- ";

cin>>name;

quan=sc->quantProd(name);

if(quan!=-1)

cout<<"Quantity of "<<name<<":- "<<quan<<endl;

else

cout<<name<<" is not present"<<endl;

break;

case 3:

cout<<"Total Cost:- "<<sc->totalCost()<<endl;

break;

case 4:

cout<<"Total Items in Cart:- "<<sc->totalItems()<<endl;

break;

case 5:

sc->placeOrder();

cout<<"Your Order is Placed"<<endl;

break;

case 6:

cout<<endl<<"Exiting..."<<endl;

break;

default:

cout<<"Wrong Input"<<endl;

}

}

return 0;

}

//output:







