**ASSIGNMENT**

1. **Definition:-**

* Write a program to create Water bill , Electricity bill of different village panchayat using Abstract Factory Design Pattern.

**2) Classes:-**

* AbadWaterBill
* GnagarWaterBill
* SuratWaterBill
* AbadElectricityBill
* GnagarElectricityBill
* SuratElectricityBill
* AbadPropertyBill
* GnagarPropertyBill
* SuratPropertyBill

**3) Class Diagram:-**

AbadPropertyBill

-getPropertyBil ()

-TearmCond ()

AbadElectricityBill

-getElectricityBill ()

-TearmCond ()

AbadWaterBill

-getWaterBill ()

-TearmCond ()

GnagarPropertyBill

-getPropertyBil ()

-TearmCond ()

GnagarElectricityBill

-getElectricityBill ()

-TearmCond ()

GnagarWaterBill

-getWaterBill ()

-TearmCond ()

SuratPropertyBill

-getPropertyBil ()

-TearmCond ()

SuratElectricityBill

-getElectricityBill ()

-TearmCond ()

SuratWaterBill

-getWaterBill ()

-TearmCond ()

1. **Data Members:-**

* Int choice1 , choice2 , n;
* Double ltr , unit , rant , charge;

**5) Methods:-**

* WaterBill

public double getWaterBill();

public void setWaterCharge(double c);

public void setWaterLtr(double ltr);

public void setNoofMonth(int n);

public String TearmCond();

* ElectricityBill

public double getElectricityBill();

public void setElectricityCharge(double c);

public void setElectricityUnit(double unt);

public void setNoofMonth(int n);

public String TearmCond();

* PropertyBill

public double getPropertyBill();

public void setPropertyRant(double rant);

public void setNoofMonth(int n);

public String TearmCond();

6**)Structure:-**

Abad

Gnagar

Gnagar

Abad

Surat

Abad

Gnagar

Abad

getPropertyBill()

PropertyBill(interface)

ElectricityBill(interface)

AbstractBillClient

AbstractBillFactory

PropertyBillFactory

ElectrycityBillFactory

WaterBillFactory

Surat

Surat

WaterBill(Interface)

Gnagar

Surat

Abad

getElectricityBill()

getPropertyBill()

getPropertyBill()

getElectricityBill()

getElectricityBill()

-getWaterBill()

-

getWaterBill()

-

getWaterBill()

**7)Intent:-**

* Provide an interface for creating families of related or dependent objects without specifying their concrete classes
* The *abstract factory pattern* provides a way to encapsulate a group of individual *factories* that have a common theme without specifying their concrete classes.

**8)Participants:-**

* **AbstractFactory**:-declare an interface for operations that create abstract product objects.(AbstractBillFactory)
* **ConcreteFactory**:-implements the operations to create concrete product objects.(WaterBillFactory , ElectricityBillFactory , PropertyBillFactory)
* **AbstractProduct**:-declares an interface for a type of product object.(WaterBill , ElectricityBill , PropertyBill)
* **ConcreteProduct**:-defines a products object to be created by the corresponding concrete factory.

Implements the AbstractProduct interface.(AbadWaterBill…AbadElectricityBill..

AbadPropertyBill…)

* **Client:-** uses only interfaces declared by AbstractFactory and AbstractProduct classes.(AbstractBillClient)

**9) Known As:-**

* Kit

**10)Collaborations:-**

* To create different product , the client should use a different concrete factory.

**11)Applicability:-**

* A system should be independent of hows its product are created , composed , and represented.
* A system should be configured with one of multiple families of products.
* A family of related product objects is designed to be used together , and you need to enforce this constraint.

**12)Advantages:-**

* It isolation concrete classes from the client.
* You use the Abstract Factory to control the classes of objects the client creates.
* Product names are isolated in the implementations of the Concrete Factory , clients use the instances through their interfaces.
* Exchanging product families is easy.
* It is the concrete factory's job to make sure that the right products are used together.

**13)Disadvantages:-**

* Adding a new product requires extending the interface which implies that all of its derived concrete classes also must change . The following changes needs to be taken care of:
* New abstract product class is added
* New product implementation is added
* Abstract factory interface is extended
* Derived concrete factories must implements the extensions
* Clients has to be extends to use the new product

**SAMPLE CODE**

* **WATERBILL**

**public interface WaterBill**

**{**

**public double getWaterBill();**

**public void setWaterCharge(double c);**

**public void setWaterLtr(double ltr);**

**public void setNoofMonth(int n);**

**public String TearmCond();**

**}**

* **ELECTRICITYBILL**

**public interface ElectricityBill {**

**public double getElectricityBill();**

**public void setElectricityCharge(double c);**

**public void setElectricityUnit(double unt);**

**public void setNoofMonth(int n);**

**public String TearmCond();**

**}**

* **PROPERTYBILL**

**public interface PropertyBill {**

**public double getPropertyBill();**

**public void setPropertyRant(double rant);**

**public void setNoofMonth(int n);**

**public String TearmCond();**

**}**

* **ABADWATERBILL**

**public class AbadWaterBill implements WaterBill**

**{**

**double charge,lrt;**

**int month;**

**public AbadWaterBill()**

**{**

**charge=0;**

**ltr=0;**

**month=0;**

**}**

**public double getWaterBill()**

**{**

**return (Charge\*lrt)\*month;**

**}**

**public void setWaterCharge(double c)**

**{**

**this.charge=c;**

**}**

**public void setWaterLtr(double ltr)**

**{**

**this.lrt=lrt;**

**}**

**public void setNoofMonth(int n)**

**{**

**this.month=n;**

**}**

**public String TearmCond()**

**{**

**return "This is Water Bill of Ahmedabad ";**

**}**

**}**

* **GNAGARWATERBILL**

**public class GnagarWaterBill implements WaterBill**

**{**

**double charge,lrt;**

**int month;**

**public GnagarWaterBill()**

**{**

**charge=0;**

**ltr=0;**

**month=0;**

**}**

**public double getWaterBill()**

**{**

**return (Charge\*lrt)\*month;**

**}**

**public void setWaterCharge(double c)**

**{**

**this.charge=c;**

**}**

**public void setWaterLtr(double ltr)**

**{**

**this.lrt=lrt;**

**}**

**public void setNoofMonth(int n)**

**{**

**this.month=n;**

**}**

**public String TearmCond()**

**{**

**return "This is Water Bill of Gandhinagar";**

**}**

**}**

* **SURATWATERBILL**

**public class SuratWaterBill implements WaterBill**

**{**

**double charge,lrt;**

**int month;**

**public SuratWaterBill()**

**{**

**charge=0;**

**ltr=0;**

**month=0;**

**}**

**public double getWaterBill()**

**{**

**return (Charge\*lrt)\*month;**

**}**

**public void setWaterCharge(double c)**

**{**

**this.charge=c;**

**}**

**public void setWaterLtr(double ltr)**

**{**

**this.lrt=lrt;**

**}**

**public void setNoofMonth(int n)**

**{**

**this.month=n;**

**}**

**public String TearmCond()**

**{**

**return "This is Water Bill of Surat ";**

**}**

**}**

* **ABADELECTRICITYBILL**

**public class AbadElectricityBill implements ElectricityBill**

**{**

**double charge,lrt;**

**int month;**

**public AbadElectricityBill()**

**{**

**charge=0;**

**ltr=0;**

**month=0;**

**}**

**public double getWaterBill()**

**{**

**return (Charge\*lrt)\*month;**

**}**

**public void setWaterCharge(double c)**

**{**

**this.charge=c;**

**}**

**public void setWaterLtr(double ltr)**

**{**

**this.lrt=lrt;**

**}**

**public void setNoofMonth(int n)**

**{**

**this.month=n;**

**}**

**public String TearmCond()**

**{**

**return "This is ElectricityBill of Ahmedabad ";**

**}**

**}**

* **GNAGARELECTRICITYBILL**

**public class GnagarElectricityBill implements ElectricityBill**

**{**

**double charge,lrt;**

**int month;**

**public GnagarElectricityBill()**

**{**

**charge=0;**

**ltr=0;**

**month=0;**

**}**

**public double getWaterBill()**

**{**

**return (Charge\*lrt)\*month;**

**}**

**public void setWaterCharge(double c)**

**{**

**this.charge=c;**

**}**

**public void setWaterLtr(double ltr)**

**{**

**this.lrt=lrt;**

**}**

**public void setNoofMonth(int n)**

**{**

**this.month=n;**

**}**

**public String TearmCond()**

**{**

**return "This is Electricity Bill of Gandhinagar ";**

**}**

**}**

* **SURATELECTRICITYBILL**

**public class SuratElectricityBill implements ElectricityBill**

**{**

**double charge,lrt;**

**int month;**

**public SuratElectricityBill()**

**{**

**charge=0;**

**ltr=0;**

**month=0;**

**}**

**public double getWaterBill()**

**{**

**return (Charge\*lrt)\*month;**

**}**

**public void setWaterCharge(double c)**

**{**

**this.charge=c;**

**}**

**public void setWaterLtr(double ltr)**

**{**

**this.lrt=lrt;**

**}**

**public void setNoofMonth(int n)**

**{**

**this.month=n;**

**}**

**public String TearmCond()**

**{**

**return "This is Electricity Bill of Surat ";**

**}**

**}**

* **ABADPROPERTYBILL**

**public class AbadPropertyBill implements PropertyBill**

**{**

**double rant;**

**int month;**

**public AbadPropertyBill()**

**{**

**rant=0;**

**month=0;**

**}**

**public double getWaterBill()**

**{**

**return (rant/8\*100)\*month;**

**}**

**public void setWaterCharge(double rant)**

**{**

**this.rant=rant;**

**}**

**public void setNoofMonth(int n)**

**{**

**this.month=n;**

**}**

**public String TearmCond()**

**{**

**return "This is Property Bill of Ahmedabad ";**

**}**

**}**

* **GNAGARPROPERTYBILL**

**public class GnagarPropertyBill implements PropertyBill**

**{**

**double rant;**

**int month;**

**public GnagarPropertyBill()**

**{**

**rant=0;**

**month=0;**

**}**

**public double getWaterBill()**

**{**

**return (rant/8\*100)\*month;**

**}**

**public void setWaterCharge(double rant)**

**{**

**this.rant=rant;**

**}**

**public void setNoofMonth(int n)**

**{**

**this.month=n;**

**}**

**public String TearmCond()**

**{**

**return "This is Property Bill of Gandhinagar ";**

**}**

**}**

* **SURATPROPERTYBILL**

**public class SuratPropertyBill implements PropertyBill**

**{**

**double rant;**

**int month;**

**public SuratPropertyBill()**

**{**

**rant=0;**

**month=0;**

**}**

**public double getWaterBill()**

**{**

**return (rant/8\*100)\*month;**

**}**

**public void setWaterCharge(double rant)**

**{**

**this.rant=rant;**

**}**

**public void setNoofMonth(int n)**

**{**

**this.month=n;**

**}**

**public String TearmCond()**

**{**

**return "This is Property Bill of Surat ";**

**}**

**}**

* **WATERBILLFACTORY**

**public class WaterBillFactory extends AbstractBillFactory**

**{**

**public WaterBill getWaterBill(int i)**

**{**

**if( i == 1 )**

**{**

**return new AbadWaterBill();**

**}**

**else if( i== 2 )**

**{**

**return new GnagarWaterBill();**

**}**

**else if(i==3)**

**{**

**return new SuratWaterBill();**

**}**

**else**

**{**

**return null;**

**}**

**}**

**}**

* **ELECTRICITYBILLFACTORY**

**public class ElectricityBillFactory extends AbstractBillFactory**

**{**

**public ElectricityBill getElectricityBill(int i) {**

**if ( i== 1 )**

**{**

**return new AbadElectricityBill();**

**}**

**else if ( i== 2 )**

**{**

**return new GnagarElectricityBill();**

**}**

**else if(i==3)**

**{**

**return new SuratElectricityBill();**

**}**

**else**

**{**

**return null;**

**}**

**}**

**}**

* **PROPERTYBILLFACTORY**

**public class PropertyBillFactory extends AbstractBillFactory**

**{**

**public PropertyBill getPropertyBill(int i) {**

**if ( i== 1 )**

**{**

**return new AbadPropertyBill();**

**}**

**else if ( i== 2 )**

**{**

**return new GnagarPropertyBill();**

**}**

**else if(i==3)**

**{**

**return new SuratPropertyBill();**

**}**

**else**

**{**

**return null;**

**}**

**}**

**}**

* **ABSTRACTBILLFACTORY**

**public abstract class AbstractBillFactory**

**{**

**public abstract WaterBill getWaterBill(int i );**

**public abstract ElectricityBill getElectricityBill(int i );**

**public abstract PropertyBill getPropertyBill(int i);**

**public static AbstractBillFactory getBill(int i)**

**{**

**if ( i == 1 )**

**{**

**return new WaterBillFactory();**

**}**

**else if(i == 2 )**

**{**

**return new ElectricityBillFactory();**

**}**

**else if(i == 3)**

**{**

**return new PropertyBillFactory();**

**}**

**{**

**return null;**

**}**

**}**

**}**

* **ABSTRACTBILLCLIENT**

**import java.util.Scanner;**

**public class AbstractBillClient**

**{**

**public static void main(String[] args)**

**{**

**AbstractBillFactory abf;**

**int choice1,choice2;**

**Scanner in = new Scanner(System.in);**

**System.out.println("1. WaterBill");**

**System.out.println("2. Electricity Bill");**

**System.out.println("3. Property Bill");**

**System.err.print("Enter your choice :- ");**

**choice1 = in.nextInt();**

**System.out.println("1.Abad ");**

**System.out.println("2.Gnagar ");**

**System.out.println("2.Surat");**

**System.out.print("Select your City :- ");**

**choice2 = in.nextInt();**

**abf = AbstractBillFactory.getBill(choice1);**

**if (choice1 == 1 )**

**{**

**WaterBill w = abf.getWaterBill(choice2);**

**w.TearmCond();**

**}**

**else if(choice1 == 2)**

**{**

**ElectricityBill e = abf.getElectricityBill(choice2);**

**e.TearmCond();**

**}**

**else if (choice1 == 3)**

**{**

**PropertyBill p = abf.getPropertyBill(choice2);**

**p.TearmCond();**

**}**

**}**

**}**