ASSIGNMENT

Date
Page

Name - Gajiya Amrita Kesubhai
Enrollment - 120081101

1) Definition -

Write a program to create Water Bill, Electricity Bill of different village pamchayat using Abstract Factory Design Pattern.

2] Classes :-

- AbadWaterBill
- AmrajanWaterBill
- SuratWaterBill
- AbadElectricityBill
- AmrajanElectricityBill
- SuratElectricityBill
- AbadPropertyBill
- AmrajanPropertyBill
- SuratPropertyBill

3] Class Diagram -

AbadWaterBill	AmrajanElectricityBill	SuratPropertyBill
-getWaterBill() -TaxAmount()	-getElectricityBill() -TaxAmount()	-getPropertyBill() -TaxAmount()

4) Data Members -

```
Int choice1, choice2, n;
Double ltr, unit, rcnt, cncnge;
```

5] Methods -

* WaterBill

```
public double getWaterBill();
public void setWaterCharge(double c);
public void setWaterLtr(double ltr);
public void setNoofMonth(int n);
public String TecrmCond();
```

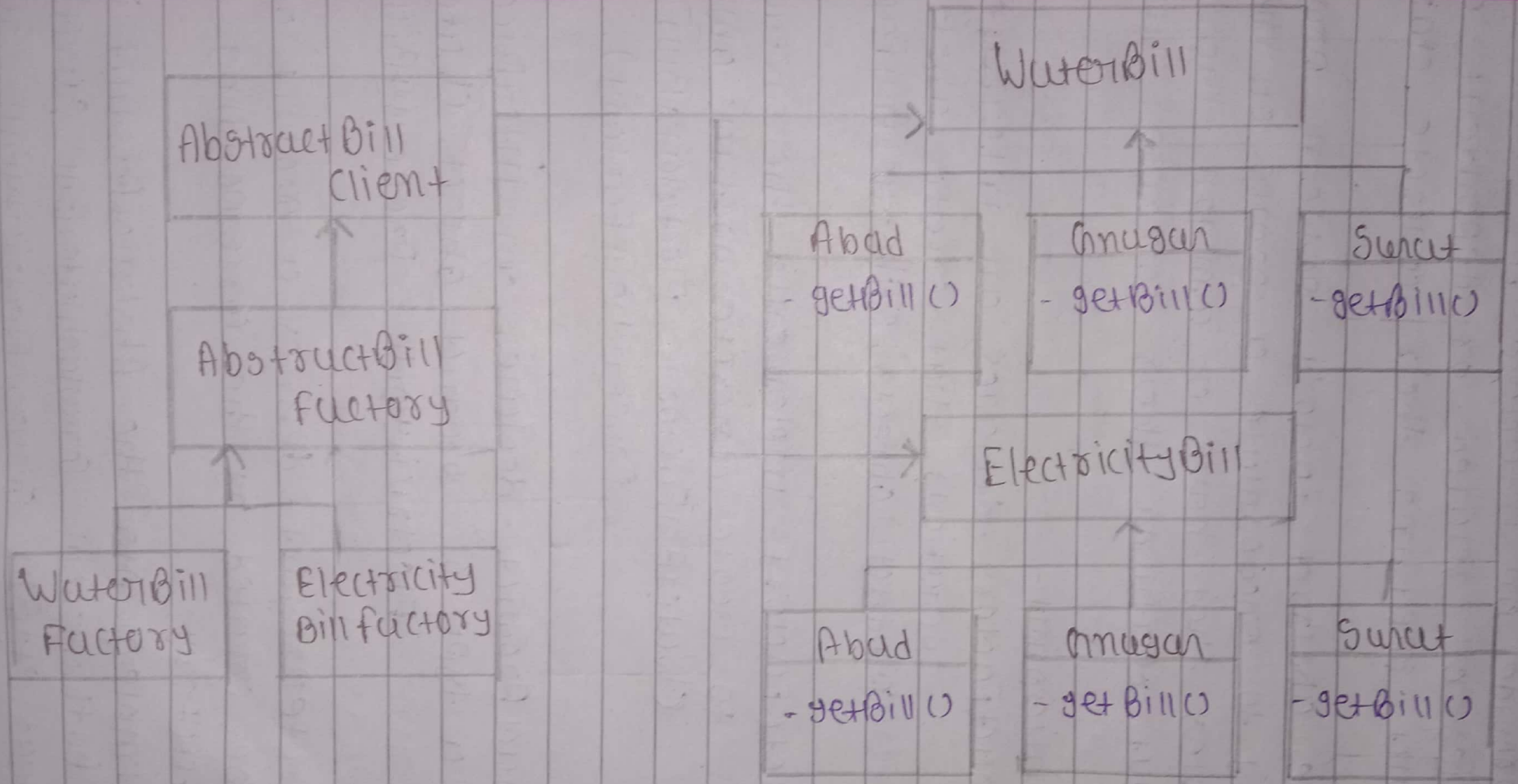
* ElectricityBill -

```
public double getElectricityBill();
public void setElectricityCharge(double c);
public void setElectricityUnit(double unt);
public void setNoofMonth(int n);
public String TecrmCond();
```

* PropertyBill -

```
public double getPropertyBill();
public void setPropertyRcnt(double rcnt);
public void setNoofMonth(int n);
public String TecrmCond();
```

c) structure -



7) Intent -

- provide an interface for creating families of related or dependent objects without specfying their concrete classes.

- The abstract factory pattern provides a way to encapsulate a group of individual factories that have a common theme without specfying their concrete classes.

8) Participant -

- AbstractFactory - declares 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 product object to be created by the corresponding concrete factory.

- implements the AbstractProduct interface (AbadWaterBill, AnagarWaterBill, SuratWaterBill, AbadElectricityBill, AnagarElectricityBill, SuratElectricityBill, AbadPropertyBill, AnagarPropertyBill)

- Client -
Uses only interfaces declared by AbstractFactory and AbstractProduct classes.

9] Known As -
Kit

10] Collaborations -
- To create different product, the client should use a different concrete factory.

11] Applicability -
- Use the Abstract Factory Pattern when

- A system should be independent of how its products 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 isolates concrete classes from the client.

- You use the Abstract Factory to contro the classes of objects the client creates

- Product names are isolated in the implementation 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 implement the extensions.
- Client has to be extended to use the new product.