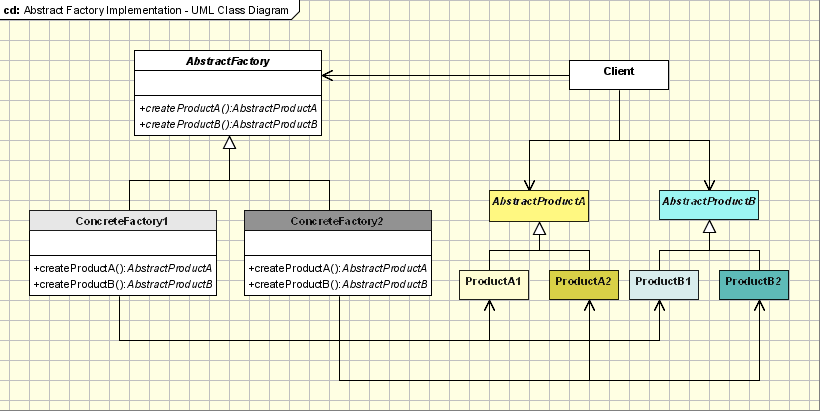
Abstract Factory

GOF : Provide an interface for creating families of related or dependent objects without specifying their concrete classes.

# **Class Diagram**



# **Java Code**

**public** **abstract** **class** AbstractFactory {

**abstract** AbstractProductA createProductA();

//Create another product

// abstract AbstractProductB createProductB();

}

**abstract** **class** AbstractProductA {

**public** **abstract** **void** operation1();

}

**class** ConcreteFactory1 **extends** AbstractFactory {

@Override

AbstractProductA createProductA() {

**return** **new** ProductA1();

}

//Create for another Product

}

**public** **class** ProductA1 **extends** AbstractProductA {

@Override

**public** **void** operation1() {

System.***out***.println("Operation1 executed ...");

}

}

**public** **class** FactoryMaker {

**private** **static** AbstractFactory *absFactory* = **null**;

**public** **static** AbstractFactory getFactory(String choice) {

**if** (choice.equals("a")) {

*absFactory* = **new** ConcreteFactory1();

} **else** {

// return another ConcreteFactory

}

**return** *absFactory*;

}

}

**public** **class** Client {

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

AbstractFactory absFact = FactoryMaker.*getFactory*("a");

AbstractProductA prodA = absFact.createProductA();

prodA.operation1();

}

}