Assignment 1 :
In an outlet, there can be several counters, each one managed by a single
Sales person selling a specific product . A customer approaches any counter, depending on the product the customer wishes to purchase. The salesperson hands over the product and accepts the payment from the customer. Identify the classes, their attributes and operations in that classes.

Code:

```
import java.util.Scanner;

abstract class Pcounter {
	abstract void process();
}

class Counterone extends Pcounter {
	Scanner p1 = new Scanner(System.in);
	 int Payment, Receipt;

	@Override
	void process() {
		System.out.println("Cost of the Product is $100");
		System.out.println("Enter Payment Status"+"\n"+"1.Done"+"\n"+"2.Not Done");

		Payment = p1.nextInt();

		if (Payment == 1) {
			System.out.println("Payment done");
			System.out.println("Enter Receipt Status"+"\n"+"1.Done"+"\n"+"2.Not Done");

			Receipt = p1.nextInt();
			if (Receipt == 1) {
				System.out.println("Receipt Given");
				System.out.println("Your produt
is:"+ProductCounters.productno+"\n"+"Payment done"+"\n"+"Receipt Given");
			} else if (Receipt == 2) {
				System.out.println("Receipt Not Given");
			}
		} else if (Payment == 2) {
			System.out.println("Payment not done");
		}

	}

}

class Countertwo extends Pcounter {
	Scanner p2 = new Scanner(System.in);
	 int Payment, Receipt;

	@Override
	void process() {
		System.out.println("Cost of the Product is $200");
		System.out.println("Enter Payment Type Status"+"\n"+"1.Done"+"\n"+"2.Not Done");
		Payment = p2.nextInt();
		if (Payment == 1) {
			System.out.println("Payment done");
			System.out.println("Enter Receipt Status"+"\n"+"1.Done"+"\n"+"2.Not Done");
```

```
                        Receipt = p2.nextInt();
                        if (Receipt == 1) {
                                System.out.println("Receipt Given");
                                System.out.println("Your produt
is:"+ProductCounters.productno+"\n"+"Payment done"+"\n"+"Receipt Given");
                        } else if (Receipt == 2) {
                                System.out.println("Receipt not Given");
                        }
                } else if (Payment == 2) {
                        System.out.println("Payment not done");
                }
        }

}

class Counterthree extends Pcounter {
        Scanner p3 = new Scanner(System.in);
         int Payment, Receipt;

        @Override
        void process() {
                System.out.println("Cost of the Product is $300");
                System.out.println("Enter Payment Status"+"\n"+"1.Done"+"\n"+"2.Not Done");
                Payment = p3.nextInt();
                if (Payment == 1) {
                        System.out.println("Payment done");
                        System.out.println("Enter Receipt Status"+"\n"+"1.Done"+"\n"+"2.Not Done");
                        Receipt = p3.nextInt();
                        if (Receipt == 1) {
                                System.out.println("Receipt Given");
                                System.out.println("Your produt
is:"+ProductCounters.productno+"\n"+"Payment done"+"\n"+"Receipt Given");
                        } else if (Receipt == 2) {
                                System.out.println("Receipt not Given");
                        }
                } else if (Payment == 2) {
                        System.out.println("Payment not done");
                }
        }
}

public class ProductCounters {
static int product;
static String productno;
        public static void main(String[] args) {
                Scanner sc = new Scanner(System.in);
                System.out.println("Welcome to our Market"+"\n"+"Enter your CHoice");
                System.out.println("1.ProductOne"+"\n"+"2.ProductTwo"+"\n"+"3.ProductThree");
                 product = sc.nextInt();

                if (product == 1) {
                        Pcounter c1 = new Counterone();
                        productno="ProductOne";
                        c1.process();
                } else if (product == 2) {//here inheritence is used we can say that as same object is used to call all
the sub classes
```

```
                    Pcounter c1 = new Countertwo();
                    productno="ProductTwo";
                    c1.process();
                } else if (product == 3) {
                    Pcounter c1 = new Counterthree();
                    productno="ProductThree";
                    c1.process();
                }
        }

}
```

Problems @ Javadoc Declaration Console

```
<terminated> ProductCounters [Java Application] C:\Program Files\Java\jdk1
Welcome to our Market
Enter your CHoice
1.ProductOne
2.ProductTwo
3.ProductThree
1
Cost of the Product is $100
Enter Payment Status
1.Done
2.Not Done
1
Payment done
Enter Receipt Status
1.Done
2.Not Done
1
Receipt Given
Your produt is:ProductOne
Payment done
Receipt Given
```