



Instituto Profesional AIEP Spa.

Dirección Nacional de Educación Continua

Programa Talento Digital 2021

Curso desarrollo de aplicaciones móviles Android Trainee

Módulo 1: Programación básica en java

EJERCICIO DE EVALUACIÓN

Descripción del ejercicio

En este ejercicio se evaluará la comprensión de la lógica adquirida respecto de los flujos y tomas de decisión que presenta un algoritmo. En esta oportunidad tomaremos situaciones de la vida real y serán llevados a diagramas lógicos los cuales son un punto de partida para la comprensión de la lógica inicial estructurada y para la programación de estos.

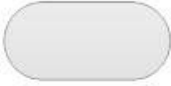

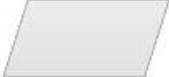
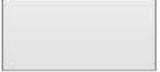

Objetivo

Leer los enunciados y representar estos de forma gráfica en diagramas de flujo simples que mostrarán la lógica de una forma inicial fácil de comprender e ir

Tiempo estimado: 50 min

Información proporcionada: Simbología de diagramas, Entidades participantes y enunciados.

Simbología de diagramas

Símbolo	Nombre	Función
	Inicio / Final	Representa el inicio y el final de un proceso
	Línea de Flujo	Indica el orden de la ejecución de las operaciones. La flecha indica la siguiente instrucción.
	Entrada / Salida	Representa la lectura de datos en la entrada y la impresión de datos en la salida
	Proceso	Representa cualquier tipo de operación
	Decisión	Nos permite analizar una situación, con base en los valores verdadero y falso

En este ejercicio se evaluará el desarrollo lógico en la comprensión de algoritmos y diagramado su representación gráfica:

En el siguiente ejercicio deberá representar con sus palabras paso a paso como es y en que consiste la adquisición y compra de un producto por parte de un cliente a una tienda.

Puede realizar la especificación de forma simple o más compleja dependiendo de su capacidad actual de análisis del algoritmo.

La forma de especificación determinará la complejidad del diagrama de flujo final

Caso Práctico

En la ciudad existe una tienda que requiere comenzar su proceso de automatización. Este comenzará por sus procesos para ellos más complejos a nivel operacional y que están utilizando mucho tiempo en obtener respuestas entre áreas para realizar finalmente las ventas.

La tienda en si posee 1 casa matriz y 1 sucursal. Cada una con su propia bodega.

Las ventas de la tienda en general funcionan de forma presencial ya que están recién migrando a comercio electrónico, pero pueden atender pedidos a través de WhatsApp y redes sociales para luego ser despachados a los domicilios de sus clientes.

El problema

Actualmente se lleva un registro de productos e inventario en tablas Excel la cual se tiene acceso y mantenimiento en cada tienda sin posibilidad de consulta remota.

Para realizar ventas de productos es necesario consultar si hay disponibilidad de stock suficiente, si no hay el personal se comunica telefónicamente con la otra tienda para preguntar si hay existencia y así poder concretar la venta.

El movimiento físico y despacho de productos no está considerada aún en esta solución.

Entidades Participantes

Estas entidades son aquellas que se relacionaran compartiendo y desplegando información necesaria a las áreas del sistema que lo requiera.

Cliente: Es aquel que dada la intención de compra o adquisición de producto genera las acciones de consulta por parte del personal de la tienda.

Tienda: Operada por su personal es quien realiza las consultas a bodega para verificar la cantidad suficiente de productos de acuerdo a lo solicitado por el cliente.

Bodega: Tiene la información de inventario y existencias que entrega a la tienda para las consultas que las tiendas realizan para las ventas a los clientes.

1- Análisis y Diseño

En palabras simples describe el análisis del problema y su posible solución en términos de automatización utilizando las entidades participantes.

En primera instancia, el problema mencionado es que no tienen una base de datos única para ambas tiendas en la cual hacer las consultas sobre productos existentes y poder concretar la venta de forma más rápida. Para ello se propone migrar la información desde los Excel a una base de datos solida con servidores externos o implementar y montarla en la casa matriz con acceso desde otras sucursales. De esta forma si el negocio se sigue expandiendo, los inventarios estarán alojados en un solo lugar y esto facilitaría la consulta de productos, acelerando también el proceso de venta. Posterior a eso, y con el stock actualizado es posible la implementación de venta online a través de el sitio web, haciendo que el cliente pueda comprar desde un ordenados, sin tener que acudir a Whatsapp o RRSS.

Este análisis inicial establecerá las bases sobre las relaciones entre las entidades y como se comportarán entre sí.

2- Requerimientos

Establece una lista de requerimientos funcionales que serán las características principales para la posterior implementación y automatización de las áreas.

1. Consultar producto

2. Consultar stock

3. Consulta lugar donde se encuentra el producto

6. Realizar compra

7. Registro Cliente

8. Agregar Productos

9. _____

10. _____

3- Definición de clases

Completa y establece los **atributos** para las **entidades participantes**:

Cliente:

- idCliente
- rutCliente
- nombreCliente
- correoCliente
- direccionCliente
- telefonoCliente

Tienda:

- nombreTienda
- direccionTienda
- telefonoTienda
- correoTienda
-

Bodega:

- nombreBodega
- direcciónBodega
- capacidadBodega

Producto:

- nombreProducto
- clasificación
- cantidad
- precio
- lugar

4- Completa y establece los métodos ó funciones para las entidades participantes:

Completa y establece los **métodos o funciones** para las **entidades participantes**:

Cliente:

- consultaProducto()
- datosCliente()
- relalizarCompra()
- seleccionarDespacho()
- registrarse()
-

Tienda:

- Direccion()
- generarVenta()
- enviarProducto()
- entregarProducto()

Bodega:

- StockProducto()
- UbicacionProducto()

Producto:

- datosProducto()

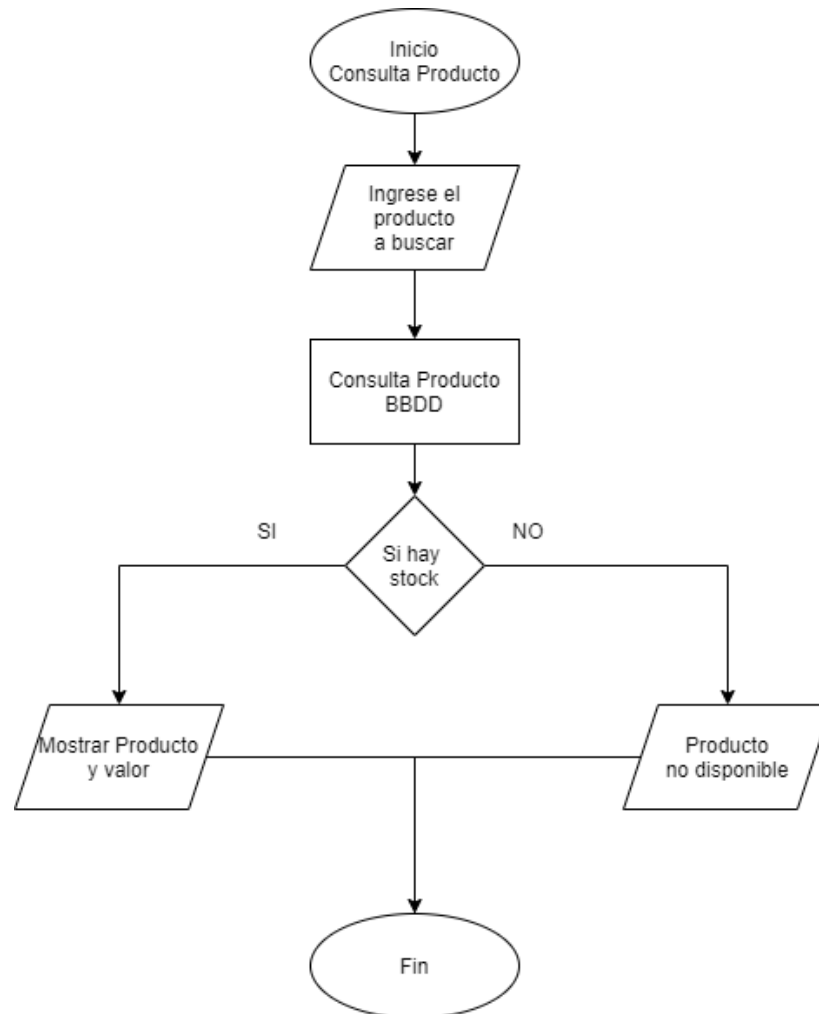
3- Expresa gráficamente las **entidades participantes** como **clase** con su nombre, atributos y métodos:

Cliente	Tienda	Bodega	Producto
-Id: String -Rut: String -Nombre: String -Correo: String -Direccion: String -Teléfono: String	-nombre : String -direccion: String -telefono: String -correo: String	-nombre: String -direccion: String -capacidad: int	-nombre: String -clasificacion: String -cantidad: int -precio int: -lugar String
+ registrarse() + datosCleinte() + realizarCompra() + selecDespacho()	+infoTienda() +generarVenta() +enviarProducto() +entregaProducto()	+stockProducto() +ubicacionproducto()	+datosProducto()

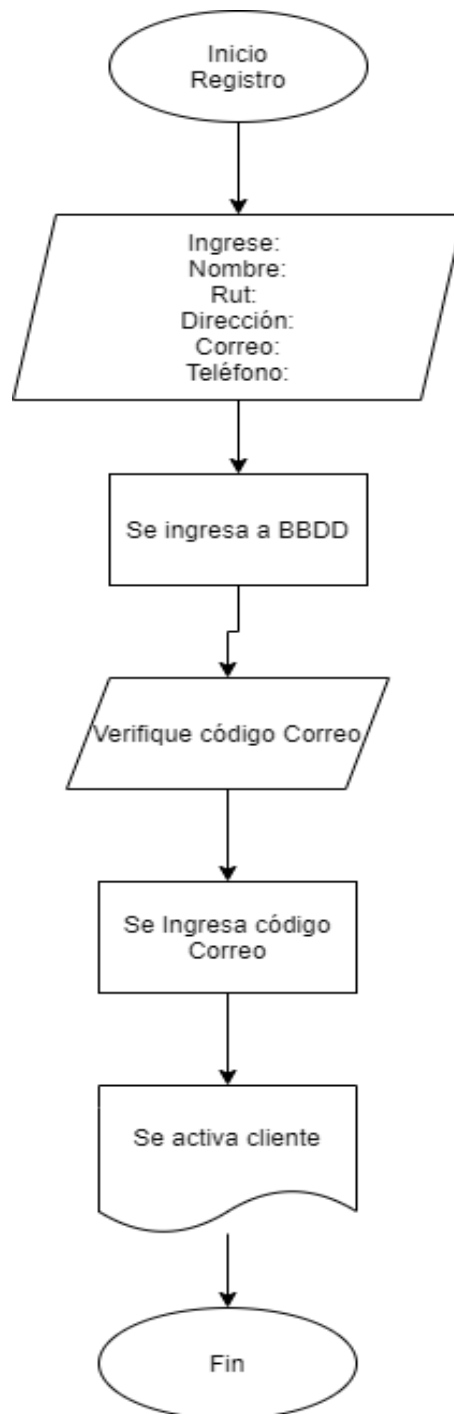
4 - Selecciona 3 Requerimientos Funcionales y exprésalos en un diagrama de flujo

(R1) Requerimiento 1:

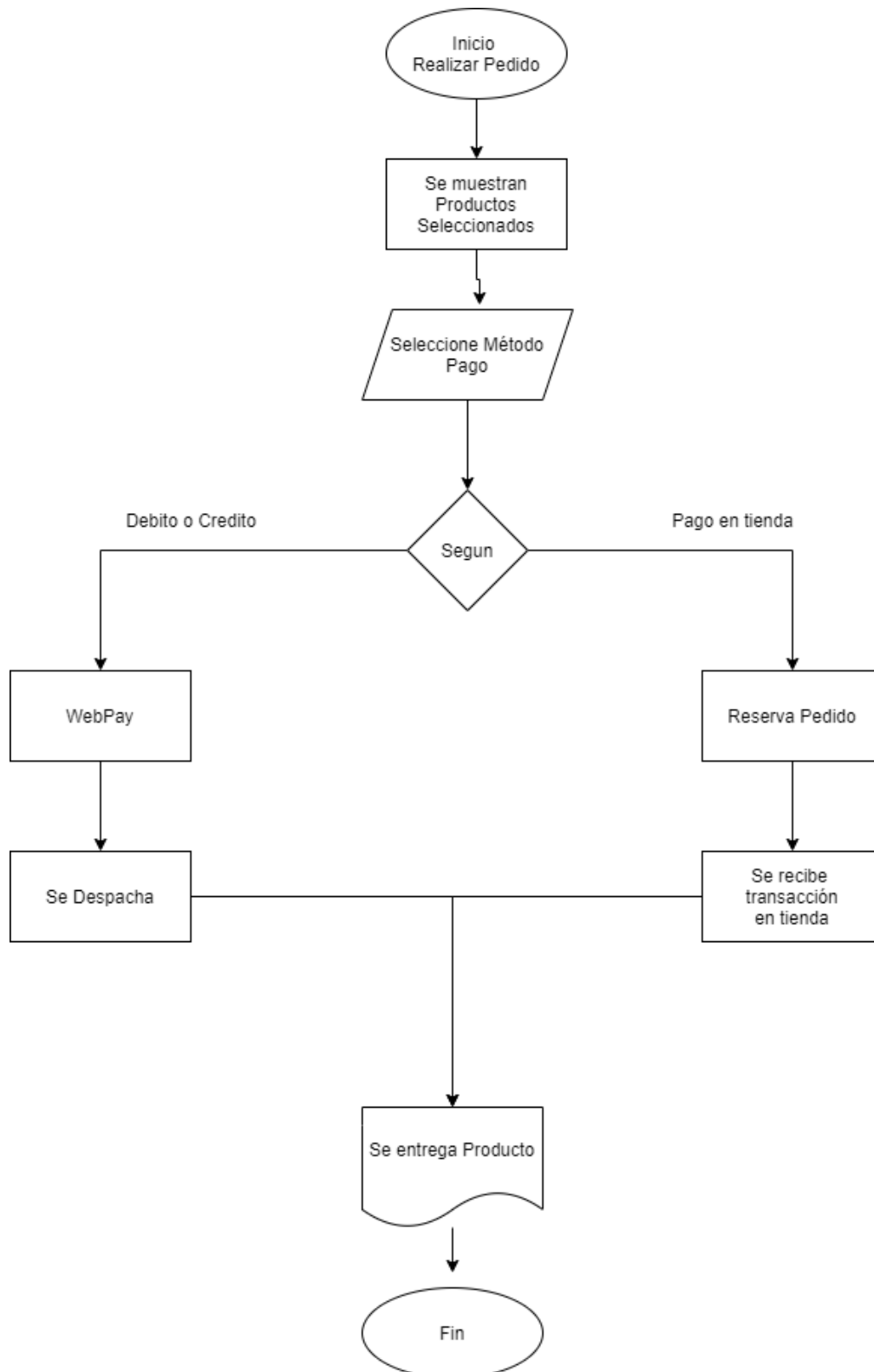
CONSULTA PRODUCTO:



- **(R2) Requerimiento 2: REGISTRO CLIENTE**



- **(R3) Requerimiento 3:
REALIZAR COMPRA**



5- Realiza la codificación simple de los diagramas realizados en lenguaje de Pseudo-código con las siguientes consideraciones:

- Mostrar entidades y procesos.
- Definir inicio y fin del proceso.
- Utilizar estructuras de control para tomas de decisión.

Algoritmo RegistroCliente

```
    Escribir "Ingrese Nombre"
    leer nombre
    Escribir "Ingrese rut"
    leer rut
    Escribir "Ingrese correo"
    leer correo
    Escribir "ingrese dirección"
    leer direccion
    Escribir "ingrese Teléfono"
    leer telefono
    Escribir "sus datos han sido ingresados"
    Escribir "hemos enviado un código a su correo"
    Escribir "para activar su cuenta ingrese código"
    a=1
    Repetir
    Escribir "*Revisa correo* Código:123"
    leer codigo
    Si (codigo=123)
        Escribir "Su cuenta ha sido activada"
        a=2
    SiNo
        Escribir "EL codigo ingresado no es correcto"
        Escribir "Intentar nuevamente? Si=1 No=2"
        Leer a
    FinSi
    Hasta Que a=2
```

FinAlgoritmo

• R1

Algoritmo RealizarCompra

```
    Escribir "Cuandos prouctos lleva?"
    Leer cantidad
    Escribir "Usted ha seleccionado los siguientes productos"
    Para i<-1 Hasta cantidad Con Paso 1 Hacer
        Escribir "producto ",i
    Fin Para
    Escribir "Ingrese tipo pago: Transferencia=1 Efectivo=2"
    leer tp
    si tp=1
        a=1
        Hacer
            Escribir "*Se envía a su casa comercial* www.tarjeta.cl"
            Escribir "El precio es: ",(5000*cantidad)
            Escribir "Ingrese valor"
            leer pago
            si pago>=(5000*cantidad)
                Escribir "Gracias por su compra"
                Escribir "Su vuelto es: ",pago-(5000*cantidad)
                Escribir "*Se hace evió producto*"
                a=2
            SiNo
                Escribir "Su pago es insuficiente para su compra"
                Escribir "Desea intenarlo nuevamente? 1=Si 2=No"
                Leer a
            FinSi
        Hasta Que a=2
    SiNo
        si tp=2
            a=1
            Hacer
                Escribir "*Se envía a caja tienda*"
                Escribir "El precio es: ",(5000*cantidad)
                Escribir "Con cuanto paga?"
                leer pago
                si pago>=(5000*cantidad)
                    Escribir "Gracias por su compra"
                    Escribir "Su vuelto es: ",pago-(5000*cantidad)
                    Escribir "*Se entrega producto*"
                    a=2
                SiNo
                    Escribir "Su pago es insuficiente para su compra"
                    Escribir "Desea intenarlo nuevamente? 1=Si 2=No"
                    Leer a
                FinSi
            Hasta Que a=2
        FinSi
    FinSi
```

FinAlgoritmo

• R2

Main: TiendaDobleBodega

Clases: Bodega

 Cliente

 Producto

 Tienda

Metodos: MostrarDatos (Bodega,Cliente,Producto,Tienda)

 Venta (Producto)

 IngresarCliente (Main)

 IngresarProducto (Main)

 IngresarTiendayBodega (Main)

 RealizarCompra (Main)

6- Establece la estructura básica de una aplicación JAVA

7- Define en código JAVA la función o método principal de los requerimientos seleccionados más arriba.

```
public class TiendaDobleBodega {

    public static void main(String[] args) {
        Scanner key=new Scanner (System.in);
        ArrayList<Cliente> cliente = new ArrayList<Cliente>();
        ArrayList<Tienda> tienda = new ArrayList<Tienda>();
        ArrayList<Bodega> bodega = new ArrayList<Bodega>();
        ArrayList<Producto> producto = new ArrayList<Producto>();
        String nombre, rut,correo, telefono, calle, numCalle,ID,tipo;
        int cantidad, precio,q;
        int a=1;
        int s=0;
        do {
            System.out.println("Qué desea hacer?");
            System.out.println("1= Ingresar Cliente");
            System.out.println("2= Ver Clientes");
            System.out.println("3= Ingresar Tiendas");
            System.out.println("4= Ver Tiendas");
            System.out.println("5= Ingresar productos");
            System.out.println("6= Ver Productos");
            System.out.println("7= Venta Producto");
            s=key.nextInt();
            key.nextLine();
            switch (s){
                case 1:
                    a=1;
            do {
                System.out.println("Nombre...");
                nombre=key.nextLine();
                System.out.println("Rut...");
                rut=key.nextLine();
                System.out.println("Correo...");
                correo=key.nextLine();
                System.out.println("Telefono...");
                telefono=key.nextLine();
                System.out.println("Calle...");
                calle=key.nextLine();
                System.out.println("Numero de Calle...");
                numCalle=key.nextLine();
                System.out.println("ID...");
                ID=key.nextLine();
```

```

        cliente.add(new Cliente(nombre, rut, correo, telefono, calle, numCalle, ID));

        System.out.println("");
        System.out.println("Ingresar otro cliente? 1=Si 2=No");
        a=key.nextInt();
        key.nextLine();
    } while (a==1);
    break;
    // Se muestrasn los clientes
        case 2:
            for (int i=0;i<cliente.size();i++){
                System.out.println("Cliente .- "+i);
                System.out.println("");
                cliente.get(i).MostrarDatos();
                System.out.println("");
                System.out.println("...");
            }
            break;
        case 3:
            // Se ingresan tiendas solo para la inicialización
do{
    //for (int i=0;i<=1;i++){
        System.out.println("Nombre Tienda...");
        nombre=key.nextLine();
        System.out.println("Calle...");
        calle=key.nextLine();
        System.out.println("Numero de Calle...");
        numCalle=key.nextLine();
        System.out.println("Telefono...");
        telefono=key.nextLine();
        System.out.println("Ingrese Correo...");
        correo=key.nextLine();

        tienda.add(new Tienda(nombre, calle, numCalle, telefono, correo));

        System.out.println("Ingrese capacidad de bodega");
        long capacidad=key.nextLong();

        bodega.add(new Bodega(("Bodega "+nombre), capacidad));

        System.out.println("");
        System.out.println("Ingresar otra tienda? 1=Si 2=No");
        a=key.nextInt();
        key.nextLine();
    }

```



```

        while (a==1);
            break;
// Se muestran las tiendas y sus bodegas
        case 4:
            for (int i=0;i<tienda.size();i++){
                System.out.println("");
                System.out.println("Tienda .- "+i);
                System.out.println("");
                tienda.get(i).MostrarDatos();
                System.out.println("...");
                bodega.get(i).MostrarDatos();
            }
            break;
// Ingreso de Productos
        case 5:
            do {
                System.out.println("Nombre Producto...");
                nombre=key.nextLine();
                System.out.println("Tipo Producto...");
                tipo=key.nextLine();
                System.out.println("Cantidad...");
                cantidad=key.nextInt();
                key.nextLine();
                System.out.println("Ingrese valor del producto...");
                precio=key.nextInt();

                // Al producto se le asocia a una bodega al momento de ingresarlo

                System.out.println("A qué Bodega desea ingresar este producto?");
                for (int i=0;i<tienda.size();i++){
                    System.out.println("");
                    System.out.println("Bodega .- "+i);
                    System.out.println("");
                    System.out.println(""+bodega.get(i).getNombre());
                }
                q=key.nextInt();

                producto.add(new
                Producto(nombre,tipo,cantidad,precio,(bodega.get(q).getNombre())));

                System.out.println("");
                System.out.println("Ingresar Otro producto? 1=Si 2=No");
                a=key.nextInt();
                key.nextLine();
            }while (a==1);
            break;

```

```

// Se Muestran los productos
    case 6:
        for (int i=0;i<producto.size();i++){
            System.out.println("");
            System.out.println("Producto .- "+i);
            System.out.println("");
            producto.get(i).MostrarDatos();
        }
        break;
    case 7:
        System.out.println("Qué producto desea?");
        for (int i=0;i<producto.size();i++){
            System.out.println("");
            System.out.println("Producto .- "+i);
            System.out.println("");
            producto.get(i).MostrarDatos();
        }
        q=key.nextInt();
        System.out.println("Cuantos lleva?");
        int c=key.nextInt();
        System.out.println("El valor es: "+(producto.get(q).getPrecio()*c);
        System.out.println("...");
        System.out.println("Con qué cancela? 1=Transferencia 2=Efectivo");
        int m=key.nextInt();
        if (m==1){
            producto.get(q).Venta(c);
            System.out.println("");
            System.out.println("Gracias por preferirnos");
        } else {
            if (m==2){
                System.out.println("Con cuanto paga?");
                int ef=key.nextInt();
                if (ef>=((producto.get(q).getPrecio()*c)){
                    producto.get(q).Venta(c);
                    System.out.println("Su vuelto es: "+(ef-
((producto.get(q).getPrecio()*c)));
                    System.out.println("");
                    System.out.println("Gracias por preferirnos");
                } else {
                    System.out.println("Su efectivo no alcanza para realizar
su compra");
                }
            }
        }
        break;
    default : System.out.println("No valido");

```

```

    }
    System.out.println("Continuar? 1=Si 2=No");
    a=key.nextInt();
    key.nextLine();
    }while (a==1);
    System.out.println("*Apagando sistema*");
    System.out.println("¡Buenas Noches!");
    }
}

```

Metodos MostrarDatos de clases:

Cliente:

```

public void MostrarDatos(){
    System.out.println("Nombre "+nombre);
    System.out.println("Rut: "+rut);
    System.out.println("Correo: "+correo);
    System.out.println("Telefono: "+telefono);
    System.out.println("Calle: "+calle);
    System.out.println("Numero: "+numeroCalle);
    System.out.println("ID: "+ID);
}

```

Tienda:

```

public void MostrarDatos(){
    System.out.println("Nombre Tienda: "+nombre);
    System.out.println("Direccion: Calle "+calle+" N° "+numeroCalle);
    System.out.println("Contacto----- ");
    System.out.println("Teléfono: "+telefono);
    System.out.println("Correo: "+correo);
}

```

Bodega:

```

public void MostrarDatos(){
    System.out.println("Nombre Bodega: "+nombre);
    System.out.println("Capacidad :"+capacidadProductos);
}

```

Producto:

```
public void MostrarDatos(){
    System.out.println("Nombre: "+nombre);
    System.out.println("Tipo: "+tipo);
    System.out.println("Cantidad: "+cantidad);
    System.out.println("Lugar: "+lugar);
    System.out.println("Precio: $" + precio);
}
```

Metodo Venta en clase Producto:

```
public void Venta(int c){
    if (c<=cantidad){
        System.out.println("Usted ha comprado lo siguiente...");
        System.out.println("");
        System.out.println("Producto: "+nombre);
        System.out.println("Unidades: "+c);
        System.out.println("Precio Final "+(c*precio));
        cantidad=cantidad-c;
    }
    else {
        System.out.println("Lo sentimos.. su petición excede nuestro stock");
    }
}
```

R1, R2 y R3