



PROGRAMACIÓN ORIENTADA A OBJETOS

Segundo Parcial 2022

HOJA 1 / 4



- 1) Analice la siguiente problemática y responda las preguntas.
 - a) ¿Qué patrón de comportamiento visto en la cursada se puede aplicar?
 - b) Aplique el patrón y resuelva la problemática.

La empresa <Objetos limpios> dedicada a la venta de productos de limpieza ofrece tres tipos de formas de pago: en un pago (llamado Pago completo), en 12 cuotas (llamado Pago en cuotas) y por Mercado Pago (llamado Mercado pago). A partir del monto a cobrar y dependiendo del tipo de pago elegido se calcula el monto final en base a lo siguiente:

- Pago completo: el monto no sufre variaciones. No tiene recargos.
- En cuotas: el monto lleva un recargo del 70%.
- Mercado Pago: el monto tiene un recargo del 50%.

La selección de método de pago debe ser realizada por el cliente a la hora de pagar (en tiempo de ejecución). Demostrar que la solución funciona ejemplificando y mostrando el resultado en consola, ejemplo de salida esperada:

```
Bienvenido al sistema de pago de Objetos Limpios:
```

```
En su carrito de compras hay:
```

```
Detergente patito: $317.1
```

```
Esponja irrompible: $73.6
```

```
Bolsa de residuos alimentada: $650.0
```

```
Seleccione la forma de pago:
```

```
1- Un pago (+0%)
```

```
2- 12 Cuotas (+70%)
```

```
3- Mercado Pago (+50%)
```

```
2
```

```
Usted esta pagando: 1769.19
```

Consideraciones:

Indentar bien el código, sea limpio, no deje códigos comentados o comentarios autogenerados.

Aplique las convenciones de java para nombrar clases, atributos, etc.

Respete el encapsulamiento.

No se olvide lo aprendido de algorítmica básica.

Si tiene dudas de algo, justifique.



PROGRAMACIÓN ORIENTADA A OBJETOS

Segundo Parcial 2022

HOJA 2 / 4



- 2) Se tiene una API encargada de enviar mensajes de las distintas empresas, para ello tienen un sistema de notificaciones push que crean mensajes cada 3 segundos.
- Revise el código y marque los errores encontrados y por qué lo son.
 - Agregue una excepción que indique cuando falle el envío de un mensaje, esto puede suceder con una probabilidad del 20%.

```
public class Mensaje {

    protected String empresa;
    protected String titulo;
    protected String cuerpo;

    public Mensaje(String empresa,String titulo, String cuerpo) {
        this.titulo = titulo;
        this.cuerpo = cuerpo;
    }

    public String toString(){
        return this.empresa+"-----"+this.titulo +"("+this.cuerpo+")";
    }
}

public class WebNotification {

    private LinkedList<Mensaje> mensajes;

    public synchronized void encolar(Mensaje mensaje) {
        while(this.mensajes.size()>=5) {
            try {
                wait();
            }catch (Exception e) {
                e.printStackTrace();
            }
        }
        if(mensaje != null) {
            System.out.println("Se almacenó para enviar el mensaje = "+mensaje.toString());
        }
        this.mensajes.add(mensaje);
    }

    public Mensaje enviar() {
        while(this.mensajes.isEmpty()) {
            try {
                wait();
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    }
}
```



PROGRAMACIÓN ORIENTADA A OBJETOS

Segundo Parcial 2022

HOJA 3 / 4



```

    }

    Mensaje mensaje = mensajes.removeFirst();
    if(mensaje != 0) {
        System.out.println("Envio mensaje = "+mensaje.toString());
    }
    notifyAll();
    return mensaje;
}

}

public class Celular{

    private String modelo;
    private WebNotification apiNotificacion;

    public Celular(String modelo, WebNotification api) {
        this.modelo = modelo;
        this.apiNotificacion = api;
    }

    public void imprimir(Mensaje mensaje) {
        System.out.println("Celular "+this.modelo+"\n"+mensaje.toString());
    }

    public void run() {
        Integer contador = 0;
        while (contador < 4) {
            Mensaje msj = this.apiNotificacion.enviar();
            if(msj == null) {
                this.imprimir(msj);
            }
        }
    }

}

public class Empresa implements Thread{

    private WebNotification apiNotificacion;
    private LinkedList<Mensaje> mensajes = new LinkedList<>();
    private String nombre;

    public Empresa(String nombre, WebNotification api) {
        this.nombre = nombre;
        this.apiNotificacion = api;
    }

    public void enviar() {
        Random rnd = new Random();
        try {
            Thread.sleep(9000);
        } catch (Exception e) {

```



PROGRAMACIÓN ORIENTADA A OBJETOS

Segundo Parcial 2022

HOJA 4 / 4



```
        e.printStackTrace();
    }
    this.apiNotification.encolar(new Mensaje(this.nombre,"titulo"+rnd.nextInt(100),
"cuervo"+rnd.nextInt(100)));
}

    public void run() {
        for(int i = 0;i<5;i++) {
            this.enviar();
        }
        this.apiNotification.encolar(null);
    }
}

public class main {

    public static void main(String[] args) {
        WebNotification api = new WebNotification();
        new Empresa("Movistar",api).run();
        new Empresa("Claro",api);
        new Celular("s21", api).start();
    }
}
```