


**Nombre:**..... **Fecha:**.....

**e-mail:**.....

1 La clase Profesor deriva de la clase Persona, complete la siguiente declaración:

```
public class Profesor ... {
}
```

2 Las clases java se agrupan en paquetes, cómo se llama el paquete en donde se encuentran las clases System, String, StringBuffer

3 Dado el siguiente código:

```
String str1 = new String("hola mundo!");
String str2 = "hola mundo cruel";
String str3 = "hola mundo cruel";
String str4 = str2;
```

Responda: ¿Cuántos objetos String son creados?

4 Dado el código anterior, indique el contenido de las áreas de memoria: Heap, Stack o Pila, Constant, Static

5 Implemente la siguiente clase:

Boleteria
-valor : double
-valorPromo : double
Boleteria()
+getValor(): double
+getValorPromo(): double
+setValor(v:double):void
+setValorPromo(v:double):void
+ticket( fecha:String, n:int):double
+ticketPromo( fecha:String, n:int):double
+resumen():void
+total():double
+totalTicket():double
+totalTicketPromo():double

Ud puede agregar los métodos y atributos que desee para lograr que la clase funcione como se indica a continuación:

**Boleteria** representa una boletería de un cine. Hay entradas comunes y entradas en promoción. Cada vez que se llama al método **ticket()** representa la venta de una entrada común. Cada vez que se llama al método **ticketPromo()** representa la venta de una entrada en promoción. El argumento **fecha**

es la fecha en que se vendió el ticket, el argumento **n** es la cantidad de ticket a vender, **ticket()** devuelve **n\*valor** y **ticketPromo()** devuelve **n\*valorPromo**. El método **total()** devuelve el total general que ha recaudado la boletería. El método **totalTicket()** devuelve el total recaudado en entradas comunes. El método **totalTicketPromo()** devuelve el total recaudado en entradas en promoción. El método **resumen()** imprime en pantalla un reporte que indique cada uno de tickets comunes o de promoción vendidos, la cantidad, el subtotal, el total general de tickets, el total general de tickets de promoción y el total general de ventas.

Criterio de aprobación: punto 5 correcto y tener el 60% del examen correcto



## RESOLUCION

1  
 public class Profesor **extends** Persona {

...

}

2

java.lang

3

```
String str1 = new String("hola mundo!");
String str2 = "hola mundo cruel";
String str3 = "hola mundo cruel";
String str4 = str2;
```

¿Cuántos objetos `String` son creados? 2

4 dado el código anterior...

contenido de las áreas de memoria: Heap, Stack o Pila, Constant, Static

**Heap:** 1 objeto `String` "hola mundo!"

**Stack:** 4 referencias: `str1`, `str2`, `str3`, `str4`

**Constant:** 1 objeto "hola mundo cruel"

**Static:** no hay nada

5

```
import java.io.*;
/**
 * Representa la boletería de un cine
 * 1er parcial POO ISFT 189
 *
 * @author G. Cherencio
 * @version 1.0
 */
public class Boleteria {
    // instance variables - replace the example below with your own
    private double valor = 0.0;
    private double valorPromo = 0.0;
    private double totalPromo = 0.0;
    private double totalComun = 0.0;
    private StringBuffer sb = new StringBuffer();

    /**
     * Constructor nulo Boleteria
     * Asume un valor de entradas de $100 para entradas comunes y
     * $ 75 para entradas en promocion
     */
    public Boleteria() {
        abrir(100.0,75.0);
    }
    /**
     * Constructor Boleteria
```



```

* Asume que el valor de las entradas de promocion son
* un 20% menos que las entradas comunes
* @param valorComun valor de un ticket entrada comun
*/
public Boleteria(double valorComun) {
    abrir(valorComun);
}
/**
 * Constructor Boleteria
 * @param valorComun valor de un ticket entrada comun
 * @param valorPromo valor de un ticket entrada en promocion
 */
public Boleteria(double valorComun,double valorPromo) {
    abrir(valorComun,valorPromo);
}
//setters
public void setValor(double v){
    valor=v;
}
public void setValorPromo(double vp){
    valorPromo=vp;
}
//getters
public double getValor (){
    return valor;
}
public double getValorPromo (){
    return valorPromo;
}
//custom
/**
 * Abre la Boleteria para comenzar a trabajar
 * @param vcomun valor de un ticket entrada comun
 * @param vpromo valor de un ticket entrada en promocion
 */
public void abrir(final double vcomun,final double vpromo) {
    setValor(vcomun);
    setValorPromo(vpromo);
    inicializo();
}
/**
 * Abre la Boleteria para comenzar a trabajar
 * Asume que los tickets de promocion valen un 20% menos que los
 * tickets comunes
 * @param vcomun valor de un ticket entrada comun
 * @param vpromo valor de un ticket entrada en promocion
 */
public void abrir(final double vcomun) {
    setValor(vcomun);
    setValorPromo(vcomun * 0.8);
    inicializo();
}
// inicializo totales y limpio reporte

```



```

private void inicializo() {
    totalPromo = 0.0;
    totalComun = 0.0;
    sb.delete(0,sb.length()); // borro el reporte, si es que habia uno!
}
/**
 * Vende un ticket de entrada comun
 * @param fecha fecha en formato dd/mm/yyyy de venta de la entrada
 * @param n cantidad de tickets vendidos en esta operacion
 * @return devuelve el monto total cobrado por esta entrada
 */
public double ticket(String fecha,int n) {
    totalComun+=getValor()*n;

sb.append(fecha+"\t"+"C\t"+n+"\t"+getValor()+"\t"+(getValor()*n)+"\n");
    return getValor()*n;
}
/**
 * Vende un ticket de entrada en promocion
 * @param fecha fecha en formato dd/mm/yyyy de venta de la entrada
 * @param n cantidad de tickets vendidos en esta operacion
 * @return devuelve el monto total cobrado por esta entrada
 */
public double ticketPromo(String fecha,int n) {
    totalPromo+=getValorPromo()*n;

sb.append(fecha+"\t"+"P\t"+n+"\t"+getValorPromo()+"\t"+(getValorPromo()*n)+"\n");
}
    return getValorPromo()*n;
}
/**
 * Imprime en consola el resumen de ventas de la boleteria
 */
public void resumen() {
    PrintStream p = System.out;
    p.println("\t\t\tCINE APOCALIPSIS\n");
    p.println("Fecha\t\tTipo\tCant.\tValor\tSubtotal");
    p.println(sb.toString()); // detalle del resumen
    p.println("\t\tTotal Tickets Comunes:\t"+totalTicket());
    p.println("\t\tTotal Tickets Promo  :\t"+totalTicketPromo());
    p.println("\t\tTotal General      :\t"+total());
}
/**
 * Total general vendido
 * @return devuelve el total general cobrado por esta boleteria (desde
 * que se abrio)
 */
public double total() {
    return totalTicket()+totalTicketPromo();
}
/**
 * Total vendido de entradas comunes
 * @return devuelve el total cobrado por esta boleteria (desde

```



**POO 1er Parcial**



**PROVINCIA DE BUENOS AIRES**  
**Dirección General de Cultura y Educación**  
Dirección de Educación Superior

```
* que se abrio) de las ventas de tickets comunes
*/
public double totalTicket() {
    return totalComun;
}
/**
 * Total vendido de entradas en promocion
 * @return devuelve el total cobrado por esta boleteria (desde
 * que se abrio) de las ventas de tickets en promocion
 */
public double totalTicketPromo() {
    return totalPromo;
}
}
```