



PARADIGMAS DE PROGRAMACION

Alumno:

López Hernández Joshua Alejandro

Profesor:

Cuauhtemoc Alba Rodriguez

Turno Matutino

Carrera: ICO 3°



Clase Tiempo2:

```
J Tiempo2.java > Tiempo2
1 public class Tiempo2 {
2     private int hora; //0-23(Almacena la hora en formato 24h)
3     private int minuto; //0-59 (Almacena los minutos)
4     private int segundo; //0-59 (Almacena los segundos)
5
6
7     //Constructor sin argumentos. Inicializa el tiempo a 00:00:00 llamando al constructor con tres parámetros.
8
9     public Tiempo2() {
10         this(hora:0, minuto:0, segundo:0); //Invoca al constructor principal con todos los valores en cero
11     }
12
13
14     //Constructor con solo la hora y la hora se establece (0-23)
15     //Inicializa los minutos y segundos a 0
16
17     public Tiempo2(int hora) {
18         this(hora, minuto:0, segundo:0); // Invoca al constructor principal con minutos y segundos en cero
19     }
20
21
22     //Constructor con hora y minutos.
23     //La hora a establecer (0-23)
24     //Los minutos a establecer (0-59)
25     //Inicializa segundos a 0.
26
27     public Tiempo2(int hora, int minuto) {
28         this(hora, minuto, segundo:0); // Invoca al constructor principal con segundos en cero
29     }
30
31
32     //Constructor principal con hora, minutos y segundos.
33     //La hora a establecer (0-23)
34     //Los minutos a establecer (0-59)
35     //Los segundos a establecer (0-59)
36     //IllegalArgumentException en caso de algún parámetro está fuera de rango
37
38     public Tiempo2(int hora, int minuto, int segundo) {
39         // Validación de parámetros
40         if (hora < 0 || hora >= 24)
41             throw new IllegalArgumentException(s:"hora debe estar entre 0 y 23");
42
43         if (minuto < 0 || minuto >= 60)
44             throw new IllegalArgumentException(s:"minuto debe estar entre 0 y 59");
45
46         if (segundo < 0 || segundo >= 60)
47             throw new IllegalArgumentException(s:"segundo debe estar entre 0 y 59");
48
49         //La asignación de valores si son válidos
50         this.hora = hora;
51         this.minuto = minuto;
52         this.segundo = segundo;
53     }
54
55
56     //Obtiene la hora actual
57     //La hora (0-23)
58
59     public int obtenerHora() {
60         return hora;
61     }
62 }
```

```
63 //Obtiene Los minutos actuales.
64 //Los minutos (0-59)
65
66 public int obtenerMinuto() {
67     return minuto;
68 }
69
70 //Los segundos actuales.
71 //Los segundos (0-59)
72
73 public int obtenerSegundo() {
74     return segundo;
75 }
76
77
78 //Convierte el tiempo a formato universal (HH:MM:SS).
79 //El String con el tiempo en formato 24h
80
81 public String aStringUniversal() {
82     return String.format(format: "%02d:%02d:%02d", obtenerHora(), obtenerMinuto(), obtenerSegundo());
83 }
84
85
86 //Convierte el tiempo a formato estándar (H:MM:SS AM/PM).
87 //String con el tiempo en formato 12h
88
89 public String toString() {
90     return String.format(format: "%d:%02d:%02d %s",
91         ((obtenerHora() == 0 || obtenerHora() == 12) ? 12 : obtenerHora() % 12),
92         obtenerMinuto(),
93         obtenerSegundo(),
94         (obtenerHora() < 12 ? "AM" : "PM"));
95 }
96 }
```

Clase PruebaAccesoMiembros:

```
1 public class PruebaTiempo2 {  
    Run | Debug  
2     public static void main(String[] args) {  
3         // Creación de objetos para ser usados con diferentes constructores  
4         Tiempo2 t1 = new Tiempo2(); //00:00:00 (Constructor sin argumentos)  
5         Tiempo2 t2 = new Tiempo2(hora:2); //02:00:00 (Constructor con solo hora)  
6         Tiempo2 t3 = new Tiempo2(hora:21, minuto:34); //21:34:00 (Constructor con hora y minuto)  
7         Tiempo2 t4 = new Tiempo2(hora:12, minuto:25, segundo:42); // 12:25:42 (Constructor completo)  
8         Tiempo2 t5 = new Tiempo2(t4); // 12:25:42 (Constructor copia, este constructor no existe en la clase original)  
9  
10        //Mostrar Los tiempos creados  
11        System.out.println(x:"Se construyo con:");  
12        mostrarTiempo(encabezado:"t1: todos los argumentos predeterminados", t1);  
13        mostrarTiempo(encabezado:"t2: se especifico hora; minuto y segundo predeterminados", t2);  
14        mostrarTiempo(encabezado:"t3: se especificaron hora y minuto; segundo predeterminado", t3);  
15        mostrarTiempo(encabezado:"t4: se especificaron hora, minuto y segundo", t4);  
16        mostrarTiempo(encabezado:"t5: se especifico el objeto Tiempo2 llamado t4", t5);  
17  
18        //La prueba de manejo para excepciones con valores inválidos  
19        try {  
20            Tiempo2 t6 = new Tiempo2(hora:27, minuto:74, segundo:99); // Valores inválidos  
21        } catch (IllegalArgumentException e) {  
22            System.out.printf(format:"%nExcepcion al inicializar t6: %s%n", e.getMessage());  
23        }  
24    }  
25  
26  
27    //Método auxiliar para mostrar un objeto Tiempo2 en ambos formato. Encabezado Texto descriptivo del tiempo  
28    //t Objeto Tiempo2 a mostrar  
29  
30    private static void mostrarTiempo(String encabezado, Tiempo2 t) {  
31        System.out.printf(format:"%s%n %s%n %s%n",  
32            encabezado,  
33            t.aStringUniversal(), // Formato 24h  
34            t.toString()); // Formato 12h  
35    }  
36 }
```

NO
COP
YAR