

# Class Evento

java.lang.Object  
Evento

```
public class Evento
extends java.lang.Object
```

Clase que representa un evento con un número limitado de lugares disponibles. Mantiene un arreglo fijo de usuarios aceptados hasta llenar la capacidad.

## Constructor Summary

Constructors	
Constructor	Description
<code>Evento(int lugaresDisponibles)</code>	Constructor que crea un evento con la capacidad especificada.

## Method Summary

All Methods	Instance Methods	Concrete Methods
Modifier and Type	Method	Description
void	<code>agregarUsuario(Usuario usuario)</code>	Intenta agregar un usuario a los aceptados.
int	<code>getLugaresDisponibles()</code>	Obtiene el número total de lugares disponibles para el evento.
int	<code>getNumAceptados()</code>	Obtiene el número actual de usuarios aceptados al evento.
Usuario[]	<code>getUsuariosAceptados()</code>	Obtiene el arreglo de usuarios que han sido aceptados al evento.
void	<code>setLugaresDisponibles(int lugaresDisponibles)</code>	Establece el número total de lugares disponibles para el evento.
void	<code>setNumAceptados(int numAceptados)</code>	Establece el número actual de usuarios aceptados al evento.
void	<code>setUsuariosAceptados(Usuario[] usuariosAceptados)</code>	Establece el arreglo de usuarios aceptados al evento.

## Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

## Constructor Detail

Evento
<pre>public Evento(int lugaresDisponibles)</pre> <p>Constructor que crea un evento con la capacidad especificada.</p> <p><b>Parameters:</b> lugaresDisponibles - el número máximo de usuarios que pueden ser aceptados.</p>

## Method Detail

### agregarUsuario

```
public void agregarUsuario(Usuario usuario)
```

Intenta agregar un usuario a los aceptados. Si no hay lugar, no hace nada.

**Parameters:**

usuario - el usuario a agregar

### getLugaresDisponibles

```
public int getLugaresDisponibles()
```

Obtiene el número total de lugares disponibles para el evento.

**Returns:**

el número máximo de usuarios permitidos en el evento

### setLugaresDisponibles

```
public void setLugaresDisponibles(int lugaresDisponibles)
```

Establece el número total de lugares disponibles para el evento.

**Parameters:**

lugaresDisponibles - el nuevo número de lugares disponibles

### getUsuariosAceptados

```
public Usuario[] getUsuariosAceptados()
```

Obtiene el arreglo de usuarios que han sido aceptados al evento.

**Returns:**

un arreglo con los usuarios aceptados

### setUsuariosAceptados

```
public void setUsuariosAceptados(Usuario[] usuariosAceptados)
```

Establece el arreglo de usuarios aceptados al evento.

**Parameters:**

usuariosAceptados - el nuevo arreglo de usuarios aceptados

### getNumAceptados

```
public int getNumAceptados()
```

Obtiene el número actual de usuarios aceptados al evento.

**Returns:**

la cantidad de usuarios actualmente aceptados

### **setNumAceptados**

```
public void setNumAceptados(int numAceptados)
```

Establece el número actual de usuarios aceptados al evento.

**Parameters:**

numAceptados - la nueva cantidad de usuarios aceptados