



Instituto Politécnico Nacional
Escuela Superior de Cómputo



Programación Orientada a Objetos

Reporte de Práctica #2 Arreglos de Objetos

Profesor: Roberto Tecla Parra

Alumno: Calva Hernández José Manuel

Grupo: 2CM3

Reservaciones de asientos en un camión

-Tablero de botones de 4X11 (numerar ,de forma consecutiva los asientos, es decir que la etiqueta de cada botón sea un número)

-Reservar el asiento cuando se hace click sobre el botón que lo representa y cambiar el color y etiqueta de dicho botón .

-Se pulsa un botón especial (el 45=44+1) para mostrar cuales asientos se han reservado.

-Los números de los asientos reservados se pueden almacenar en una variable de tipo String la cual hará el papel de acumulador. Cada que se hace clic en un botón se puede agregar el número de asiento a la cadena.

Objetivos

- Aprender el uso de arreglos de objetos en Java.
- Hacer uso de un ciclo simple (for) para ir agregando cada uno de los objetos.
- Mediante botones elegir la imagen que se quiere que aparezca en una etiqueta.
- Aprender cómo se extraen cosas de los objetos para ponerlos en otros mediante el uso de get y set.
- Hacer uso de una Applet para visualizar mejor el procedimiento de las acciones.

Desarrollo

Primero importamos las librerías de Java que nos permitirían posteriormente realizar eventos en nuestro programa y la de Applet para poder ver la función del mismo en una pequeña aplicación.

```
import java.awt.event.*;  
import javax.swing.*;  
import java.applet.*;  
import java.awt.*;  
import javax.imageio.*;  
import java.io.*;  
import java.awt.image.*;
```

Después establecimos nuestra clase Camion como una extensión de JApplet implementando la acción de escucha, posteriormente declaramos los objetos correspondientes, un arreglo de botones tipo JButton para simular los asientos, además de uno adicional que simulará el botón de reservador, para terminar, una etiqueta tipo JLabel, en la cual serán puestas los asientos que previamente fueron seleccionadas por los botones.

```
public class Camion extends JApplet implements ActionListener{

    JButton[] b = new JButton[44]; // Asientos del camión
    JButton reservado; // Botón que muestra los reservados
    JLabel mensaje; // Despliegue de los reservados
    String cadena = ""; // Cadena que contiene los asientos que han sido reservados
    public Camion(){}
}
```

Posteriormente se declaró como pública a Camion, se creó un método de inicialización (init) que nos permitió darle los parámetros iniciales a los objetos, se creó una cuadrícula de 44 espacios donde los botones serían puestos en cada uno con un ciclo for que iría incrementando de 1 en 1 unidad y a su vez a cada botón se le agregaría la acción escucha y sería agregado. Al concluir se agregaría al final el botón de Reservado.

```
public void init(){

    reservado = new JButton("Reservados: ");
    mensaje = new JLabel("");
    setLayout(new GridLayout(12,4,15,5)); // Layout del programa, 11 filas + botón final
    for(int i = 0; i<44; i++){ // Acciones sobre botones
        b[i] = new JButton(""+(1+i));
        add(b[i]);
        b[i].addActionListener(this);
    }
    add(reservado); // Acción sobre el botón para reservados
    reservado.addActionListener(this);
    add(mensaje); // Mensaje de reservados
    setSize(650,700);
}
}
```

Luego se estableció la acción evento, que permitiría darle a cada botón una acción, la cual consistía en que al oprimir cualquier botón este cambiaría de color para simular la reservación y se añadiría su número a la cadena de reservaciones.

```
public void actionPerformed(ActionEvent e){

    JButton seleccionado=(JButton)e.getSource(); // Con getSource se lee
    if(seleccionado == reservado){ // Si se seleccionó reservado, mostrar asientos reservados
        mensaje.setText(""+cadena);
    }
    else{ // en otro caso, colorear el botón y agregar a la cadena
        seleccionado.setBackground(Color.lightGray);
        seleccionado.setForeground(Color.red);
        cadena = cadena + " " + seleccionado.getText();
    }
}
}
```

Por último, creamos el html para poder crear la simulación como un Applet.

```
<html>
<body>
<applet code="Camion.class" width=256 height=256></applet>
</body>
</html>
```

Para ejecutar el Applet, debemos de seguir las instrucciones mostradas a continuación:



Conclusiones

Para agregar una serie de objetos del mismo tipo, de una forma más rápida, es mediante un ciclo, además es necesario establecer un nombre adecuado para cada objeto para que sean más rápido de reconocer. Es importante establecer las acciones escucha y evento para visualizar que los objetos cumplan sus actividades debidas. Si no se inicializa un programa, al igual que los objetos que serán "seteados" en el mismo, no podrán proceder las actividades de cada uno de ellos, además de que no aparecerá ningún objeto en la aplicación JApplet (nula visualización).