

GRAU D'ENGINYERIA INFORMÀTICA

PROGRAMACIÓ II

Bloc 3:

Solució exercici POE (2)

Laura Igual

Departament de Matemàtiques i Informàtica

Facultat de Matemàtiques i Informàtica

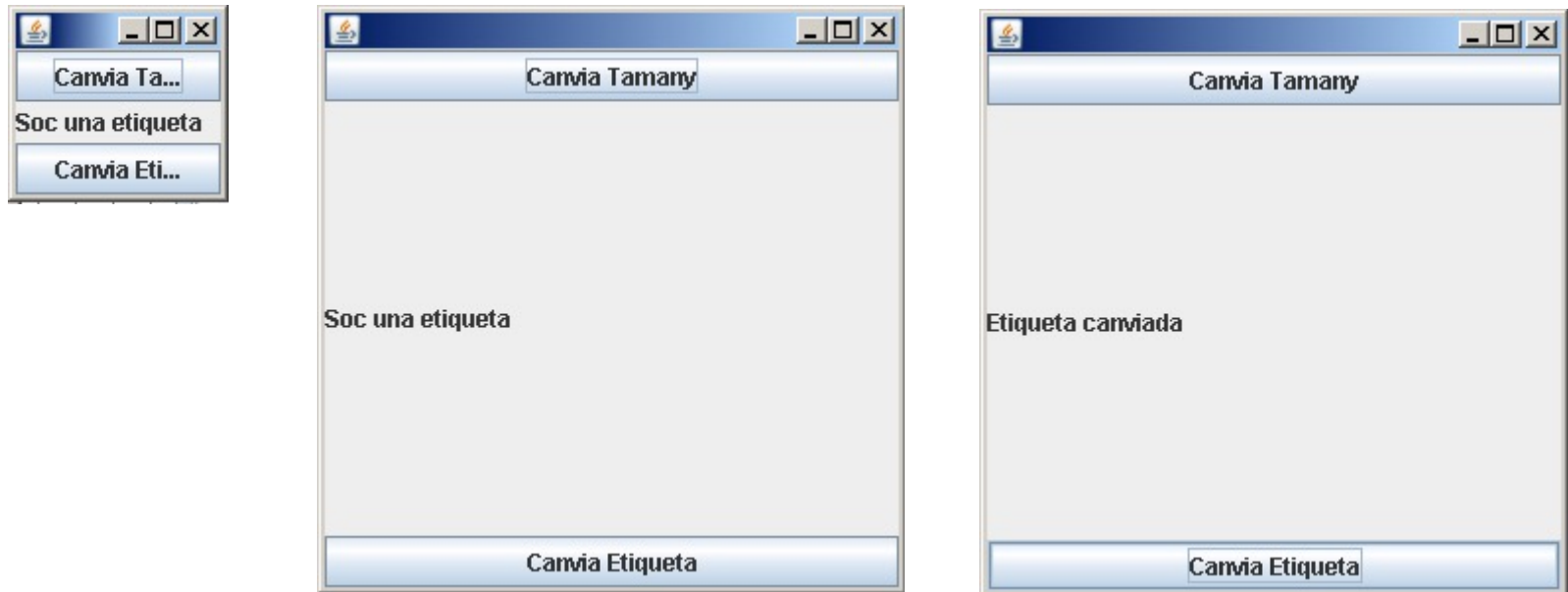
Universitat de Barcelona



Exemple 4

- Com gestionem events `ActionEvent` per a dos botons quan cada botó necessita fer una cosa diferent?

Exemple 4: Com gestionem events ActionEvent per a dos botons quan cada botó necessita fer una cosa diferent?



Pensar la viabilitat de les possibles implementacions i implementar-les:

1. La classe aplicació també fa el paper d'escoltador.
2. Separem l'escoltador en dues classes externes.
3. Utilitzem classes internes.

Exemple 4: Opició A

- Implementar dos mètodes actionPerformed()

```
class LaMevaGUI implements ActionListener {  
    // codi aquí  
    public void actionPerformed(ActionEvent ev) {  
        frame.setSize(300, 300);  
    }  
    public void actionPerformed(ActionEvent ev) {  
        etiqueta.setText("Etiqueta canviada");  
    }  
} // fi de la classe LaMevaGUI
```

Però això és impossible

Exemple 4: Opició B

```
public class LaMevaGUI implements ActionListener {
    private JFrame frame;
    private JLabel etiqueta;
    private JButton boto1, boto2;
    public LaMevaGUI(){
        frame = new JFrame();
        etiqueta = new JLabel("Soc una etiqueta");
        boto1 = new JButton("Canvia Tamany");
        boto2 = new JButton("Canvia Etiqueta");
    }
    public void go() {
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.getContentPane().add(BorderLayout.NORTH, boto1);
        frame.getContentPane().add(BorderLayout.SOUTH, boto2);
        frame.getContentPane().add(BorderLayout.CENTER, etiqueta);
        boto1.addActionListener(this);
        boto2.addActionListener(this);

        frame.setSize(100, 100);
        frame.setVisible(true);
    }
    public void actionPerformed(ActionEvent ev) {
        if(ev.getSource().equals(boto1) ){
            frame.setSize(300, 300);
        }else{
            etiqueta.setText("Etiqueta canviada");
        }
    }
    public static void main(String[] args) {
        LaMevaGUI gui = new LaMevaGUI();
        gui.go();
    }
} // fi de la classe LaMevaGUI }
```

- Registrar el mateix listener amb dos botons

Registre'm el mateix listener



Mirem quin event és



Opció correcta, però pot arribar a ser complicada de gestionar

Exemple 4: Opició C

- Crear dues classes ActionListener separades

```
class LaMevaGUI {  
    private JFrame frame;  
    private JLabel etiqueta;  
    public static void main (String[] args){  
        LaMevaGUI gui = new LaMevaGUI2();  
        gui.go();  
    }  
    public void go(){  
        frame = new JFrame();  
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        JButton boto1 = new JButton("Canvia Tamany");  
        JButton boto2 = new JButton("Canvia Etiqueta");  
        etiqueta = new JLabel("Soc una etiqueta");  
        frame.getContentPane().add(BorderLayout.NORTH, boto1);  
        frame.getContentPane().add(BorderLayout.SOUTH, boto2);  
        frame.getContentPane().add(BorderLayout.CENTER, etiqueta);  
        boto1.addActionListener(new Boto1Listener());  
        boto2.addActionListener(new Boto2Listener());  
        frame.setSize(100, 100);  
        frame.setVisible(true);  
    }  
} // fi de la classe
```

```
public class Boto1Listener implements  
ActionListener {  
    public void actionPerformed(ActionEvent ev){  
        frame.setSize(300, 300);  
    }  
} // fi de la classe
```

```
public class Boto2Listener implements  
ActionListener {  
    public void actionPerformed(ActionEvent ev){  
        etiqueta.setText("Etiqueta canviada");  
    }  
} // fi de la classe
```

Problema! Aquesta classe no té
referència a la variable etiqueta i
frame

Exemple 4: Opició D

- Crear dues classes ActionListener separades

Arreglant el problema d'abans

```
public class LaMevaGUI2 {  
    private JFrame frame;  
    private JLabel etiqueta;  
    public static void main (String[] args){  
        LaMevaGUI2 gui = new LaMevaGUI2();  
        gui.go();  
    }  
    public void go(){  
        frame = new JFrame();  
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        JButton boto1 = new JButton("Canvia Tamany");  
        JButton boto2 = new JButton("Canvia Etiqueta");  
        boto1.addActionListener(new Boto1Listener(this));  
        boto2.addActionListener(new Boto2Listener(this));  
        etiqueta = new JLabel("Soc una etiqueta");  
        frame.getContentPane().add(BorderLayout.NORTH, boto1);  
        frame.getContentPane().add(BorderLayout.SOUTH, boto2);  
        frame.getContentPane().add(BorderLayout.CENTER, etiqueta);  
        frame.setSize(100, 100);  
        frame.setVisible(true);  
    }  
}
```

```
    public JLabel getEtiqueta(){  
        return etiqueta;  
    }  
    public void setEtiqueta(JLabel etiqueta){  
        this.etiqueta = etiqueta;  
    }  
    public JFrame getFrame(){  
        return frame;  
    }  
    public void setFrame(JFrame frame){  
        this.frame = frame;  
    }  
} // fi de la classe
```

Exemple 4: Opició D

- Crear dues classes ActionListener separades

```
public class Boto1Listener implements ActionListener {  
    private LaMevaGUI2 gui;  
  
    public Boto1Listener(LaMevaGUI2 gui){  
        this.gui = gui;  
    }  
  
    public void actionPerformed(ActionEvent ev) {  
        gui.getFrame().setSize(300, 300);  
    }  
} // fi de la classe
```

Atenció! Aquestes classes han de rebre l'objecte LaMevaGUI2 per poder accedir als atributs de la mateixa.

Potser fa **difícil** entendre el codi.

```
public class Boto2Listener implements ActionListener {  
    private LaMevaGUI2 gui;  
  
    public Boto2Listener(LaMevaGUI2 gui){  
        this.gui = gui;  
    }  
  
    public void actionPerformed(ActionEvent ev) {  
        gui.getEtiqueta().setText("Etiqueta canviada");  
    }  
} // fi de la classe
```


Exemple 4: Opició E

- Crear dues classes ActionListener **internes amb nom**.

```

import java.awt.BorderLayout;
import javax.swing.*;
import java.awt.event.*;
public class LaMevaGUI {
    JFrame frame;
    JLabel etiqueta;
    public static void main(String[] args){
        LaMevaGUI gui = new LaMevaGUI();
        gui.go();
    }
    public void go() {
        frame = new JFrame();
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        JButton botoTamany = new JButton("Canvia Tamany");
        botoTamany.addActionListener(new
BotoTamanyListener());
        JButton botoEtiqueta = new JButton("Canvia Etiqueta");
        botoEtiqueta.addActionListener(new
BotoEtiquetaListener());
        etiqueta = new JLabel("Soc una etiqueta");
        frame.getContentPane().add(BorderLayout.NORTH,
botoTamany);
        frame.getContentPane().add(BorderLayout.SOUTH,
botoEtiqueta);
        frame.getContentPane().add(BorderLayout.CENTER,
etiqueta);
        frame.setSize(100, 100);
        frame.setVisible(true);
    }
} // Fi mètode go

```

Classes internes

```

class BotoTamanyListener implements ActionListener
{
    public void actionPerformed(ActionEvent ev) {
        frame.setSize(300, 300);
    }
}
class BotoEtiquetaListener implements ActionListener
{
    public void actionPerformed(ActionEvent ev) {
        etiqueta.setText("Etiqueta canviada");
    }
}
} // Fi de la classe LaMevaGUI

```



Exemple 4: Opició E

Exemple 4: Opició F

- Crear dues classes ActionListener **internes anònimes**.
- Aquesta és l'opció més comuna quan tenim moltes components en la GUI.

```

import java.awt.BorderLayout;
import javax.swing.*;
import java.awt.event.*;

public class LaMevaGUI4 {
    JFrame frame;
    JLabel etiqueta;
    public static void main(String[] args){
        LaMevaGUI4 gui = new LaMevaGUI4();
        gui.go();
    }
    public void go() {
        frame = new JFrame();
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        JButton botoTamany = new JButton("Canvia Tamany");
        botoTamany.addActionListener(new ActionListener(){
            public void actionPerformed(ActionEvent ev) {
                frame.setSize(300, 300);
            }
        });
        JButton botoEtiqueta = new JButton("Canvia Etiqueta");
        botoEtiqueta.addActionListener(new ActionListener(){
            public void actionPerformed(ActionEvent ev) {
                etiqueta.setText("Etiqueta canviada");
            }
        });
    }
}

```

```

etiqueta = new JLabel("Soc una etiqueta");

frame.getContentPane().add(BorderLayout.NORTH,
    botoTamany);
frame.getContentPane().add(BorderLayout.SOUTH,
    botoEtiqueta);

frame.getContentPane().add(BorderLayout.CENT
    ER, etiqueta);

frame.setSize(100, 100);
frame.setVisible(true);
} // Fi mètode go
} // Fi de la classe LaMevaGUI

```

Classes internes anònimes

Fa **més fàcil** entendre el codi.

Exemple 4: Opició F



Comentaris sobre les classes internes

- S'ha d'implementar tots els mètodes de la interfície
- Si el codi utilitzat per a implementar la manipulació d'events té unes poques línies es sol utilitzar una **classe interna anònima**.
- Una **classe interna amb nom** serà útil quan la volem instanciar més d'una vegada.
- No hi ha garantia de quin Listener és notificat primer.
- No escriure codi contant amb un ordre específic.