

Multiple Choices: Right answer +4 points, wrong -1 point, empty +0. *There is just one right answer per question*

Name and Surname

D.N.I :

1. Which of the following sentences is right?

- a. A source can handle only one listener at the same time
- b. A source can handle more than one listener of the same type at the same time.**
- c. A source can handle more than one listener at the same time, but only if they process different events
- d. A source that register a two listeners for the same event (p1 and later, p2), will keep only the last one (p2).
- e. None of the answers is right

Comentado [DFL1]: ok

2. We want to implement an event handler for the Key Pressed event. Given that the KeyListener interface has three methods, ...

- a. We can extend the KeyListener or implement the KeyAdapter, both strategies will work.
- b. The straight implementation of the listener limit us to have no more than these three methods in our listener, so we extend the adapter, and that way we can create more private methods to handle the logic of the listener.
- c. We cannot use the adapter class, it does not exist for key events
- d. We can implement the KeyListener or extend the KeyAdapter, both strategies will work.**
- e. We can implement both the KeyListener or the KeyAdapter, both strategies will work.

Comentado [DFL2]:

3. The picture shows a screen shot of an Internet connection speed test. Which principles have the designers applied here?



- a. Consistency
- b. Recoverability
- c. Familiarity
- d. Observability
- e. More than one answers are right.**

Comentado [DFL3]: ok

4. Regarding the text components, the fact is that...

- a. If we want to associate a mnemonic to a JTextField, we must modify its own *displayedMnemonic* property
- b. A Label must be unabled whenever its associated text component is unabled.**
- c. A *PasswordField* allows the same edition possibilities as a text field (including cut, paste, copy, etc.).
- d. Every answer is right
- e. None of the answers is right.

Comentado [DFL4]: ok

5. Given the following code, and considering the receiver or listener is correctly registered in the source object, if we want the implement the same behavior over text2 and text1, be best strategy would be:

```
class ProcesaFoco1 extends FocusAdapter{
    public void focusGained (FocusEvent e){
        text1.setText(""); }
}
```

- a. Add a new class (for example, ProcesaFoco2) and implement focusGained as in ProcesaFoco1 but replacing text1 with ((JTextField)e.getSource()). Create an object of this new class and register it in the new source object.

- b. Replace the *text1* with ((JTextField) e.getSource()) and register the same receiver object in the new source object.
- c. Replace in *ProcesaFoco1* *text1* with (e.getSource()) and register the new receiver in the new source object.
- d. Add a new class (for example, *ProcesaFoco2*) and implement *focusGained* as in *ProcesaFoco1* but replacing *text1* with *text2*. Create an object of this class and register it in the new source object.
- e. Replace the name of the class with *ProcesaFoco*, *text1* with *JTextField* and register the existing receiver in the new source object.

Comentado [DFL5]: Ésta

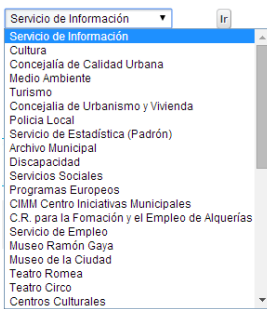
6. Given the following code, where **KeyEvent.VK\_COMMA** is the char corresponding to the comma symbol, and considering the receiver or listener is correctly registered in the source object, we can state that:

```
class ProcesaTecla extends KeyAdapter
{
    public void keyTyped (KeyEvent e){
        comprueba(e);
    }
}

private void comprueba(KeyEvent e) {
    char tecla = e.getKeyChar();
    if (tecla == KeyEvent.VK_COMMA)
        e.consume();
}
```

- a. The code is correct and will ignore every comma the user writes in the source object.
- b. The code is not correct because the method *consume* must be applied over the object *tecla*.
- c. The code is correct and will ignore every key stroke different than comma pressed on the event receiver object
- d. The code is not correct because the method *consume* cannot be invoked over a *KeyEvent* object
- e. Both answers c and d are correct

Comentado [DFL6]: ok



7. The combobox of the picture...

- a. It does not satisfy all the recommendations related to its content, given that its elements are not correctly organized.
- b. It does not need keyboard access support given that it already has an "Ir" (Go to...) button
- c. a) and b) are correct.
- d. It satisfies all the recommendations related to its content, given that its elements are capitalized.
- e. None of the answers is right.

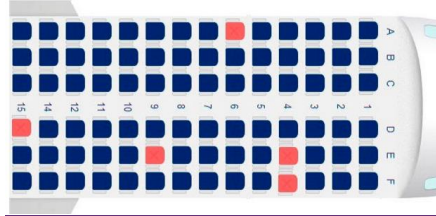
Comentado [DFL7]: Ésta

8. Regarding the Tooltips...

- a. Lighten the work of the long term memory
- b. Lighten the work of the short term memory
- c. Attending to the recommendations discussed in the class, there should be a configuration option to disable them.
- d. A and C are correct.
- e. B and C are correct.

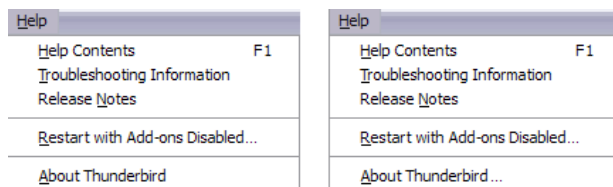
Comentado [DFL8]:

9. The picture shows a sketch of the seats of a plane. The blue seats are available, the red seats are already booked, and the size of the font used to identify the rows is 12 points. Regarding the design recommendations related to the human visual system restrictions, this interface:



- The font size should be smaller than 12 points to avoid user's visual sensorial memory saturation.
- Satisfies all the recommendations
- It should use any different extra code (besides the color) to avoid user's perception confusion.
- More than one answer is right.
- None of the answers is right.

Comentado [DFL9]:



10. Considering the usability recommendations, and that they are the same for buttons and menu items, regarding "About" item (that shows an *informative dialog*), we can state that:

- The item that uses "..." is satisfying the recommendation.
- The item that does not use "..." is satisfying the recommendation.
- Both are correct, given that the only applicable recommendation to this case is related to mnemonics, and both satisfy it.
- Both are correct, given that both of them show the new dialog to the user.
- None of the answers is right.

Comentado [DFL10]: Ok

#### 11. Toggle buttons ...

- Cannot be used in toolbars because they cannot have mnemonics.
- Should **never** be used to represent exclusive options, we must use radio buttons for that
- Should **never** be used to represent non-exclusive options
- More than one answer is right.
- None of the answers is right.

Comentado [DFL11]: ok

#### 12. In relation to Swing and AWT...

- Swing and AWT contain the same visual components, but in Swing these components have been improved by adding new attributes (border, icon, tooltip, etc.).
- Both specifications are part of the JFC (Java Foundation Classes)
- AWT includes the package "event" that includes the resources to work with events
- Every answer is right.
- Only B and C answers are right.

Comentado [DFL12]:

#### 13. Considering that the object pT is correctly registered as listener in an object area, if we need to stop pT from listening events generated from area...

- area.consume(pT);
- pT.consume(area);
- pT.addKeyListener(null);
- area.removeKeyListener(pT);
- pT.removeKeyListener(area);

Comentado [.13]:

#### 14. Which of the following factors help to improve the usability of an application?:

- Updating the application frequently using any interaction technique to increase users attention capability
- To avoid as far as possible the use of metaphors of the real world.
- The immediate notification to the user of any change occurred in the system
- The sensorial capacity of the user while he/she is using the application.
- More than one answer is right.

Comentado [DFL14]:

**15. Regarding the default button of a window...**

- a. As any other command button, we must declare its mnemonic
- b. As any other command button, it should have an associated shortcut
- c. It is mandatory in every dialog, but the cancel button is not mandatory
- d. It should **always** be the located in the last position of the lower row of buttons of the dialog..
- e. None of the answers is right.

Comentado [DFL15]: