

Enseignement secondaire général  
Division technique générale – Section technique générale  
Examen 1GIG

Liste des composants et classes connus

Liste des composants (propriétés, événements et méthodes) et classes à connaître pour l'épreuve en informatique à l'examen de fin d'études secondaires générales - division technique générale.

Package	Classe	Details	Remarques / Constantes
javax.swing	JFrame	<u>Méthodes</u> - setTitle(...) / getTitle() <u>NetBeans Object Inspector Property</u> - title	
	JButton JLabel JTextField	<u>Méthodes</u> - setText(...) / getText() - setVisible(...) - setEnabled(...) <u>Événement</u> - actionPerformed <u>NetBeans Object Inspector Property</u> - icon	- le libellé <i>JLabel</i> peut aussi être utilisé pour visualiser des images via la propriété « icon » de l'inspecteur objet de NetBeans (le composant <i>JTextField</i> ne possède pas de propriété « icon »).
	JSlider	<u>Méthodes</u> - setMinimum(...) / getMinimum() - setMaximum(...) / getMaximum() - setValue(...) / getValue() <u>Événement</u> - stateChanged	
	JPanel	<u>Méthodes</u> - setVisible(...) - setBackground(...) / getBackground() - getWidth() / getHeight() - paintComponent(Graphics g) - repaint() <u>Événements</u> - MousePressed / MouseReleased - MouseDragged / MouseMoved	- <i>JPanel</i> est utilisé pour regrouper d'autres composants visuels et pour réaliser des dessins. - Lors de la réalisation de dessins, la méthode <b>public void paintComponent (Graphics g)</b> est à surcharger.
javax.swing	JList	<u>Méthodes</u> - setListData(...) - getSelectedIndex() / setSelectedIndex(...) <u>Événement</u> - valueChanged <u>NetBeans Object Inspector - Properties</u> - model - selectionMode (SINGLE) <u>NetBeans Object Inspector - Code</u> - Type Parameters : - (vide)	- <i>JList</i> est utilisé surtout pour afficher le contenu d'une liste <i>ArrayList</i> .
java.awt.event	ActionEvent	- Ce type d'objet est uniquement utilisé dans les méthodes de réaction ajoutées de manière automatique à l'aide de NetBeans.	
	MouseEvent	<u>Méthodes</u> - getX() / getY() - getPoint() - getButton()	<u>Constantes</u> - BUTTON1 - BUTTON2 - BUTTON3

Package	Classe	Details	Remarques
javax.swing	Timer	<u>Constructeur</u> - Timer(int,ActionListener) <u>Méthodes</u> - start() - stop() - setDelay(...) - isRunning()	
		- Comme <i>ActionListener</i> on utilisera de préférence celui d'un bouton. <u>Exemple :</u> <pre>timer = new Timer(1000,stepButton.getActionListeners()[0]);</pre>	
java.awt	Graphics	<u>Méthodes</u> - drawLine(...) - drawOval(...) / fillOval(...) - drawRect(...) / fillRect(...) - drawString(...) - setColor(...) / getColor()	
	Color	<u>Constructeurs</u> - Color(...)	
	Point	<u>Constructeurs</u> - Point(...) <u>Attributs (publics)</u> - x et y <u>Méthodes</u> - getLocation() / setLocation(...)	
java.util	ArrayList	<u>Méthodes</u> - add(...) - clear() - contains(...) - get(...) - indexOf(...) - remove(...) - set(...) - size() - isEmpty() - toArray()	- Object[] toArray() est employé uniquement pour passer les contenus d'une liste à la méthode <i>setListData(...)</i> d'une <i>JList</i> .
java.lang	String	<u>Méthodes</u> - equals(...) / compareTo(...) - contains(...) - valueOf(...)	
	Integer Double	<u>Méthodes</u> - equals(...) / compareTo(...) - valueOf(...)	
	Math	<u>Méthodes</u> - abs(...) - round(...) - random() - sqrt(...) - pow(...) - sin(...), cos(...), tan(...)	<u>Constante:</u> - PI
	System	<u>Méthode</u> - out.print() - out.println()	