

الكونستركتور مع تعريفه

```
public JList()
```

1

ينشئ كائن من الكلاس `JList` يمثل `List` فارغة, أي لا تحتوي أي عنصر.

```
public JList(E[]Data)
```

2

ينشئ كائن من الكلاس `JList` يمثل `List` فيها خيارات جاهزة.

مكان الباراميتر `listData` نمرر مصفوفة من أي نوع.

عناصر هذه المصفوفة هي التي ستظهر كخيارات في القائمة.

قد يرمي إستثناء في حال كانت المصفوفة `listData` فارغة, أي قيمها تساوي `null`.

```
public JList(ListModel<E> dataModel)
```

3

ينشئ كائن من الكلاس `JList` يمثل `List` فيها خيارات جاهزة.

مكان الباراميتر `dataModel` نمرر كائن من الكلاس `ListModel`.

عناصر الكائن `dataModel` هي التي ستظهر كخيارات في القائمة.

يرمي الإستثناء `IllegalArgumentException` في حال كان كائن الـ `dataModel` فارغاً, أي قيمته تساوي `null`.

الدالة مع تعريفها	
<div>public void addListSelectionListener(ListSelectionListener listener)</div> <div>تستخدم لتنفيذ أوامر معينة عند النقر على أي خيار موجود في كائن الـ <code>JList</code> الذي قام بإستدعائها. مكان الباراميتر <code>listener</code> نضع كائن من الكلاس <code>ActionListener</code> و نفعل بداخله <code>Override</code> لدالة إسمها <code>actionPerformed(ActionEvent ae)</code>.</div>	1
<div>public E getSelectedValue()</div> <div>ترجع قيمة العنصر الذي قام المستخدم باختياره في كائن الـ <code>JList</code> الذي قام بإستدعائها. في حال قام المستخدم باختيار أكثر من عنصر فإنها ترجع قيمة العنصر الذي يملك أصغر <code>index</code> بينهم. في حال لم يقم المستخدم باختيار أي عنصر فإنها ترجع القيمة <code>null</code>.</div>	2
<div>public int getSelectedIndex()</div> <div>ترجع رقم يمثل <code>index</code> العنصر الذي قام المستخدم باختياره في كائن الـ <code>JList</code> الذي قام بإستدعائها. في حال قام المستخدم باختيار أكثر من عنصر ترجع <code>index</code> أصغر عنصر بينهم.</div>	3
<div>public int[] getSelectedIndices()</div> <div>ترجع مصفوفة أرقام، كل رقم فيها يمثل <code>index</code> عنصر من العناصر التي قام المستخدم باختيارها في كائن الـ <code>JList</code> الذي قام بإستدعائها.</div>	4
<div>public void setModel(ListModel<E> model)</div> <div>تستخدم لتحديد <code>Model</code> كائن الـ <code>JList</code> الذي قام بإستدعائها. مكان الباراميتر <code>model</code> نضع كائن من الكلاس <code>ListModel</code>. ترمي الإستثناء <code>IllegalArgumentException</code> في حال كان كائن الـ <code>dataModel</code> فارغاً، أي قيمته تساوي <code>null</code>.</div>	5
<div>public ListModel<E> getModel()</div> <div>ترجع نسخة من كائن الـ <code>ListModel</code> الذي يستخدمه كائن الـ <code>JList</code> الذي قام بإستدعائها.</div>	6
<div>public void setLayoutOrientation(int layoutOrientation)</div> <div>تستخدم لتحديد طريقة ظهور العناصر بداخل كائن الـ <code>JList</code> الذي قام بإستدعائها. مكان الباراميتر <code>layoutOrientation</code> نضع إحدى الثوابت الموجودة في الكلاس <code>ListModel</code> المخصصة لتحديد طريقة ظهور العناصر بداخل كائن الـ <code>JList</code>. الثوابت التي يسمح باستخدامها هي: <div><code>JList.VERTICAL</code></div><div><code>JList.VERTICAL_WRAP</code></div><div><code>JList.HORIZONTAL_WRAP</code></div> إفتراضياً، الثابت الذي يستخدم هو <code>JList.VERTICAL</code>. ترمي الإستثناء <code>IllegalArgumentException</code> في حال كان الثابت المستخدم غير مناسب للدالة.</div>	7
<div>public void setSelectionMode(int selectionMode)</div> <div>تستخدم لتحديد طريقة ظهور العناصر بداخل كائن الـ <code>JList</code> الذي قام بإستدعائها. مكان الباراميتر <code>layoutOrientation</code> نضع إحدى الثوابت الموجودة في الكلاس <code>ListSelectionMode</code> المخصصة لتحديد طريقة ظهور العناصر بداخل كائن الـ <code>JList</code>. الثوابت التي يسمح باستخدامها هي: <div><code>ListSelectionMode.SINGLE_SELECTION</code></div><div><code>ListSelectionMode.SINGLE_INTERVAL_SELECTION</code></div><div><code>ListSelectionMode.MULTIPLE_INTERVAL_SELECTION</code></div> إفتراضياً، الثابت الذي يستخدم هو <code>ListSelectionMode.MULTIPLE_INTERVAL_SELECTION</code>. ترمي الإستثناء <code>IllegalArgumentException</code> في حال كان الثابت المستخدم غير مناسب للدالة.</div>	8