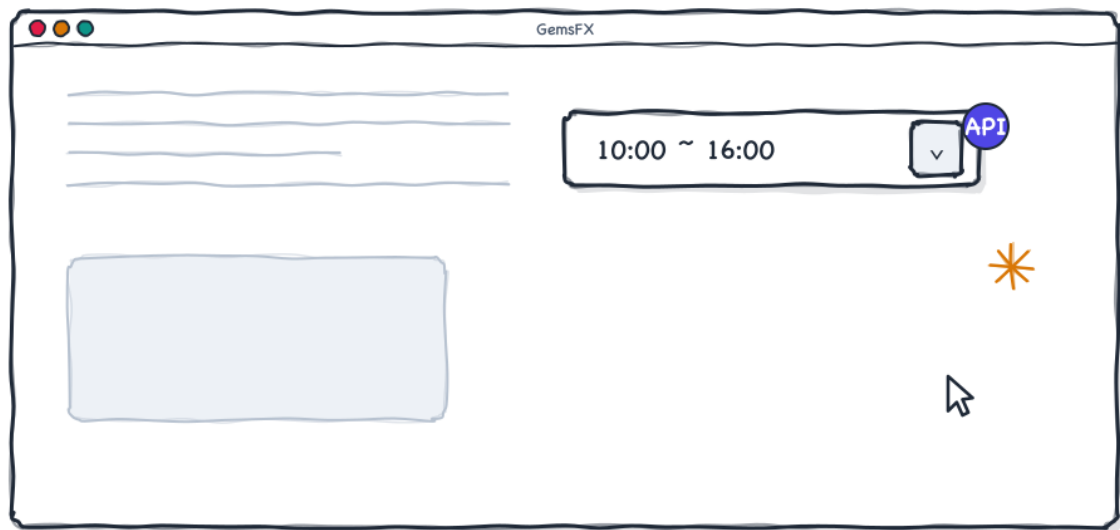


TimeRangePicker

Developer Manual

GemsFX • com.dlsc.gemsfx • JavaFX Custom Controls



Generated cartoon overview of TimeRangePicker.

TimeRangePicker is a SelectionBox of TimeRange items. It ships with common two-hour ranges, clear/select-all popup buttons and display text that merges consecutive selected ranges.

Table of Contents

1. Introduction	3
1.1 Key features	3
1.2 Maven dependency	3
2. Getting started	3
3. Anatomy	4
4. Control API	5
4.1 TimeRangePicker additions	5
4.2 SelectionBox API	5
4.3 TimeRange value object	5
5. Behaviour	6
5.1 Selection modes	6
5.2 Merging display text	6
5.3 Popup lifecycle	6
6. Styling	7
6.1 Style classes	7
6.2 Pseudo classes	7
6.3 Styleable CSS properties	8
7. Localization	8
8. Accessibility	8
9. Recipes	8
9.1 Use custom ranges	9
9.2 Enable multiple selection	9
9.3 Customize range text	9
9.4 Set a placeholder	9
9.5 Read selected ranges	9
9.6 Checklist	9

10. See also 10

1. Introduction

TimeRangePicker is part of the GemsFX module `com.dlsc.gemsfx`. This manual documents only behavior verified in the control, skin, CSS, resource bundles and demo app.

1.1 Key features

- Extends `SelectionBox`.
- Default items are 10:00-12:00, 12:00-14:00, 14:00-16:00 and 16:00-18:00.
- Supports SINGLE and MULTIPLE selection through the selection model.
- Clear and Select All buttons are added to the popup top area.
- Selected consecutive ranges are merged in the displayed text.

1.2 Maven dependency

```
<dependency>
  <groupId>com.dlsc.gemsfx</groupId>
  <artifactId>gemsfx</artifactId>
  <version>4.4.1</version>
</dependency>
```

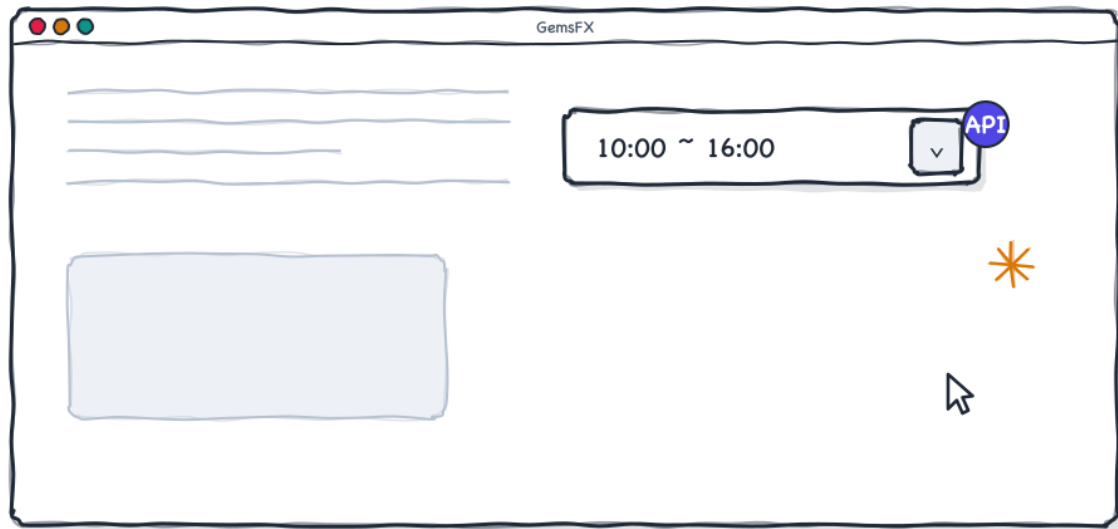
Use package `com.dlsc.gemsfx`.

2. Getting started

Create the control, set the properties that define its valid values, and listen to the value or selection property that the control owns.

```
TimeRangePicker picker = new TimeRangePicker(
    new TimeRangePicker.TimeRange(LocalTime.of(8, 0), LocalTime.of(10, 0)),
    new TimeRangePicker.TimeRange(LocalTime.of(10, 0), LocalTime.of(12, 0)),
    new TimeRangePicker.TimeRange(LocalTime.of(13, 0), LocalTime.of(15, 0))
);
picker.getSelectionModel().setSelectionMode(SelectionMode.MULTIPLE);
picker.getSelectionModel().selectIndices(0, 1);
```

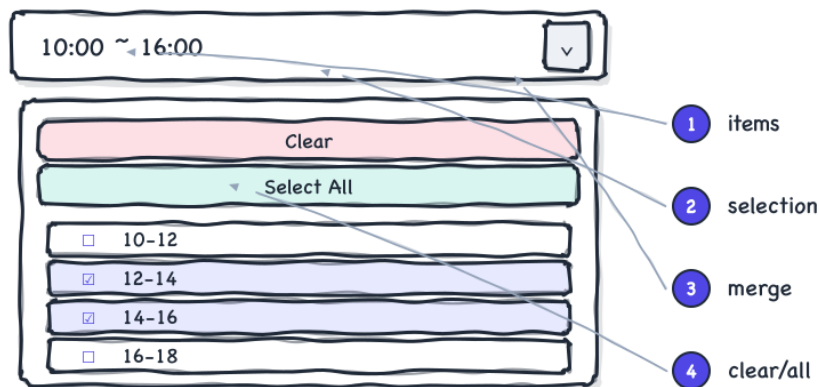
A compact setup for `TimeRangePicker`.



A first look at the control in a simple application window.

3. Anatomy

The skin builds the visible nodes below the public control. The table lists the style classes and nodes that are useful when reading the source or writing CSS.



The main parts of the control.

Part	Style / node	Description
Root	combo-box-base combo-box selection-box time-range-picker	SelectionBox root plus TimeRangePicker style class.
Display label	display-label	Shows prompt text or converted selected ranges.
Arrow button	arrow-button / arrow	Opens the SelectionPopup.
Top area	extra-buttons-box	Clear and Select All buttons.

Part	Style / node	Description
Options	options-box	Radio buttons in SINGLE mode or check boxes in MULTIPLE mode.
TimeRange	startTime/endTime	Immutable value object; constructor swaps endpoints if start is after end.

4. Control API

4.1 TimeRangePicker additions

Property	Type	Default	Description
items	ListProperty<TimeRange>	four default ranges	Inherited items list populated by the constructor.
selectedItemsConverter	ObjectProperty<StringConverter<List<TimeRange>>>	merging converter	Formats selected ranges; multiple consecutive ranges are merged.
top	ObjectProperty<Node>	extra buttons box	Popup top node with Clear and Select All buttons.

4.2 SelectionBox API

Property	Type	Default	Description
selectionModel	ObjectProperty<MultipleSelectionModel<TimeRange>>	CustomMultipleSelectionModel	Controls selected item(s) and SelectionMode.
currentSelectionMode	ReadOnlyObjectProperty<SelectionMode>	selection model mode	Convenience read-only mirror of the model mode.
promptText	StringProperty	null	Text shown when no range is selected.
itemConverter	ObjectProperty<StringConverter<TimeRange>>>	null	Formats individual ranges for popup and fallback display.
graphic	ObjectProperty<Node>	null	Optional graphic in the display label.
placeholder	ObjectProperty<Node>	null	Shown in popup when items is empty.
autoHideOnSelection	BooleanProperty	true	Controls whether popup actions hide the popup.
readOnly	BooleanProperty	false	Prevents interaction and hides arrow when true. Styleable.
animationEnabled	BooleanProperty	false	Fades popup open/closed when true. Styleable.

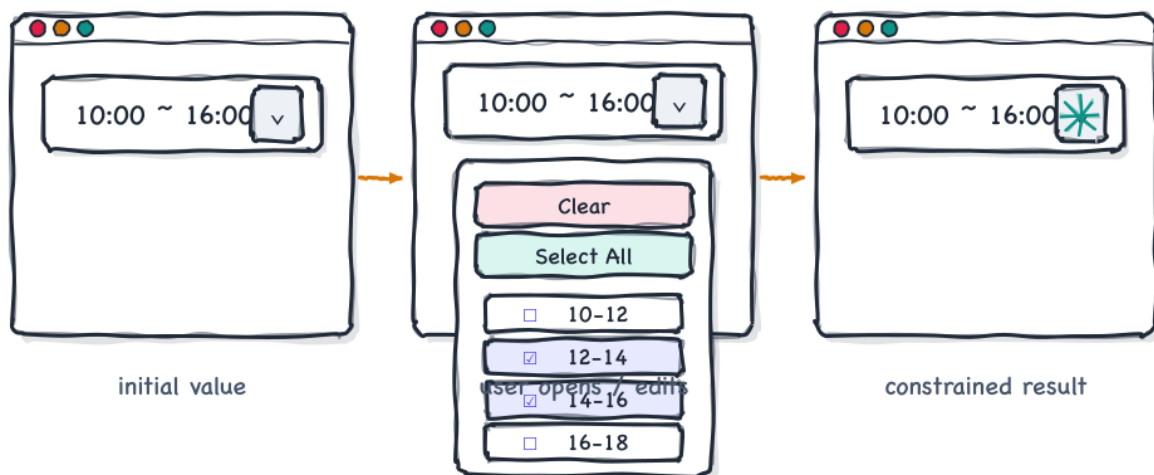
4.3 TimeRange value object

Property	Type	Default	Description
startTime	LocalTime	constructor argument	Start of the range.
endTime	LocalTime	constructor argument	End of the range.
toString()	String	start + " ~ " + end	Default text when no item converter is installed.

5. Behaviour

5.1 Selection modes

SINGLE mode creates radio buttons and hides the Select All button. MULTIPLE mode creates check boxes and shows Select All.



Interaction and value-flow behaviour.

5.2 Merging display text

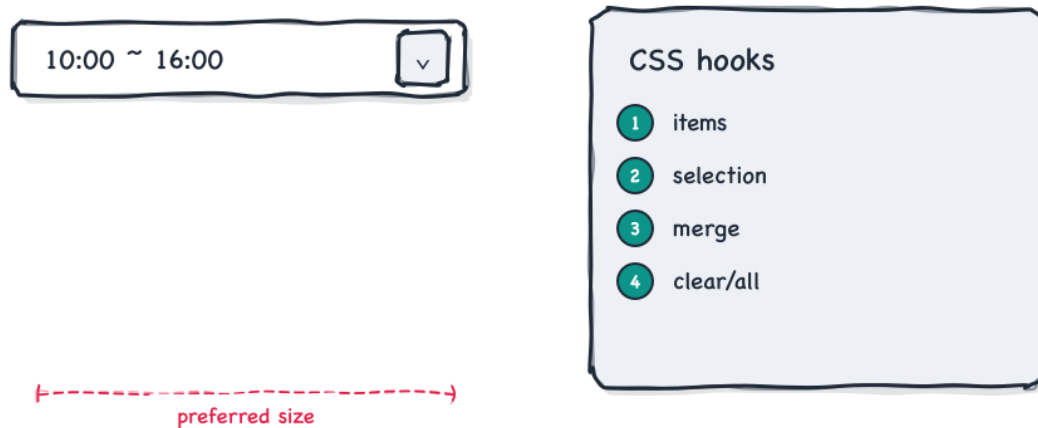
For multiple selections, selected ranges are sorted by start time. Adjacent ranges whose endTime equals the next startTime are merged before display.

5.3 Popup lifecycle

SelectionBox exposes show(), hide() and WindowEvent handlers for showing, shown, hiding and hidden. The popup auto-fixes, auto-hides and hides on ESCAPE.

6. Styling

The stylesheet is loaded by the control or its base class. The rows below list style hooks verified in the CSS and skin source.



Style classes and pseudo-class states.

6.1 Style classes

Style class	Where used
time-range-picker	Additional root class set by TimeRangePicker.
selection-box combo-box combo-box-base	Inherited SelectionBox root classes.
display-label	Main display label.
arrow-button arrow	Popup arrow.
selection-popup	Popup root.
content	Popup BorderPane.
extra-buttons-box extra-button clear-button select-all-button	Quick action buttons.
options-scroll-pane options-box	Scrollable options container.
item single-item multiple-item item-N	Radio/check option classes.

6.2 Pseudo classes

Pseudo class / marker	Meaning
empty	No selected item(s).
showing	Popup is showing.
single	Selection mode is SINGLE.

Pseudo class / marker	Meaning
multiple	Selection mode is MULTIPLE.
readonly	Read-only state in SelectionBoxSkin source.

6.3 Styleable CSS properties

Property	Type	Default	Description
-fx-read-only	Boolean	false	Inherited from SelectionBox.
-fx-animation-enabled	Boolean	false	Inherited from SelectionBox.

```
.time-range-picker {  
    -fx-read-only: false;  
    -fx-animation-enabled: true;  
}  
.selection-popup > .content .extra-buttons-box .clear-button {  
    -fx-font-weight: bold;  
}
```

Example CSS.

7. Localization

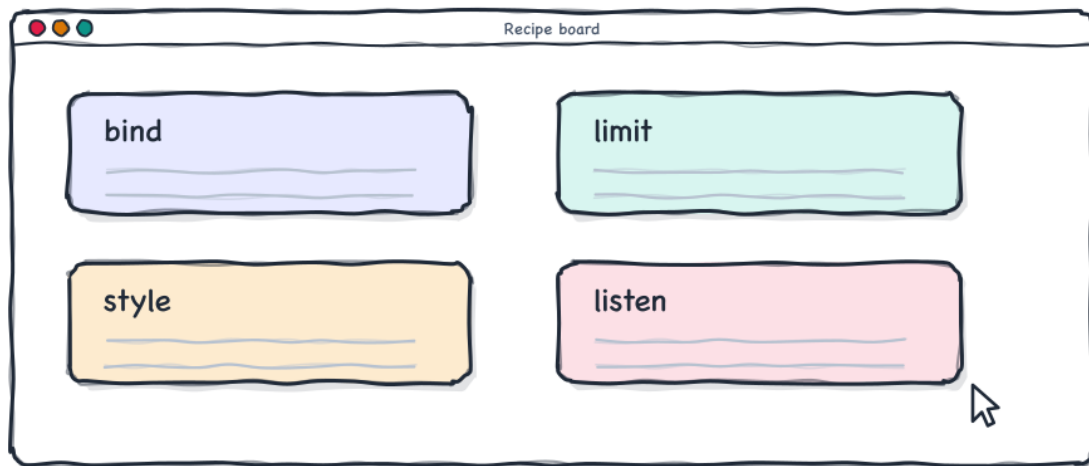
This control uses `ResourceBundleManager` for the following keys.

Key	English default
action.clear	Clear
action.select-all	Select All

8. Accessibility

`SelectionBox` sets `AccessibleRole.COMBO_BOX` and binds accessible text to the selected ranges, or `promptText` when the selection is empty. `TimeRangePicker` also sets `AccessibleRole.COMBO_BOX`.

9. Recipes



Common configuration recipes.

9.1 Use custom ranges

```
TimeRangePicker picker = new TimeRangePicker(  
    new TimeRangePicker.TimeRange(LocalTime.of(9, 0), LocalTime.of(12, 0)),  
    new TimeRangePicker.TimeRange(LocalTime.of(13, 0), LocalTime.of(17, 0))  
);
```

9.2 Enable multiple selection

```
picker.getSelectionModel().setSelectionMode(SelectionMode.MULTIPLE);
```

9.3 Customize range text

```
picker.setItemConverter(new StringConverter<>() {  
    @Override public String toString(TimeRangePicker.TimeRange range) {  
        return range.startTime() + " - " + range.endTime();  
    }  
    @Override public TimeRangePicker.TimeRange fromString(String text) { return null; }  
});
```

9.4 Set a placeholder

```
picker.setPromptText("Choose a time range");
```

9.5 Read selected ranges

```
List<TimeRangePicker.TimeRange> ranges =  
    new ArrayList<>(picker.getSelectionModel().getSelectedItems());
```

9.6 Checklist

- 1 Use the selection model to switch SINGLE/MULTIPLE.

- 2 Consecutive selected ranges merge only for display text.
- 3 The TimeRange constructor requires non-null endpoints and swaps reversed endpoints.
- 4 There is no time-range-picker.css; styling comes from selection-box.css plus the extra root class.

10. See also

Demo app: TimeRangePickerApp. Run it with:

```
mvn javafx:run -f gemsfx-demo/pom.xml -Dmain.class=com.dlsc.gemsfx.demo.TimeRangePickerApp
```

- Related GemsFX controls: TimePicker, DurationPicker, SelectionBox.
- API documentation: <https://dlsc-software-consulting-gmbh.github.io/GemsFX/api/>