

Предмет	МДК 03.01 ТРПО
Дата	11.11.2020
Группа	474
Студенты	Цветков М. В.
Работа	№ 4

1. Настройка интерфейса.

Графический интерфейс состоит из двух текстовых полей (для обозначения полей ввода), поля ввода для обработки данных, выпадающий список с названиями изображений и одного листа для вывода информации.

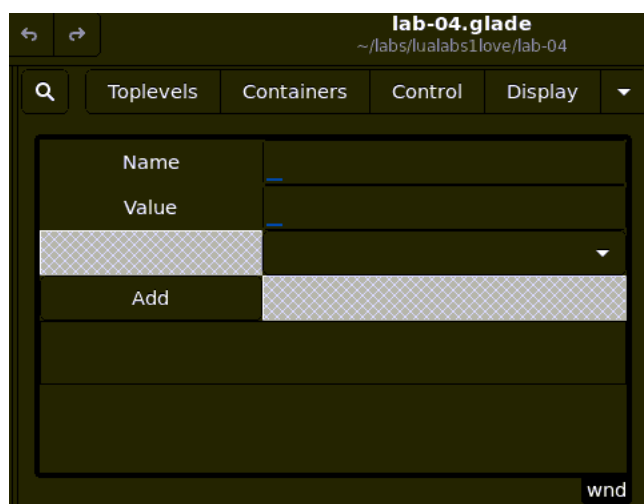


Рисунок 1 – Графический интерфейс

2. Добавление модели в выпадающий список

Для добавления модели в выпадающий список сначала нужно ее создать и заполнить. Заполненная модель на рисунке 2.

Add and remove columns:

Column type	Column name
gchararray	Name
gint64	ID
< define a new column >	

Data:

Add remove and edit rows of data (you can optionally use Ctrl+N to add new rows and the Delete key to remove the selected row)

Add and remove rows:

+

-





Name		ID
apple		1
orange		2
blueberry		3
cherry		4

Рисунок 2 – Модель данных

Первым делом добавляем модель в источник для списка (рисунок 3).

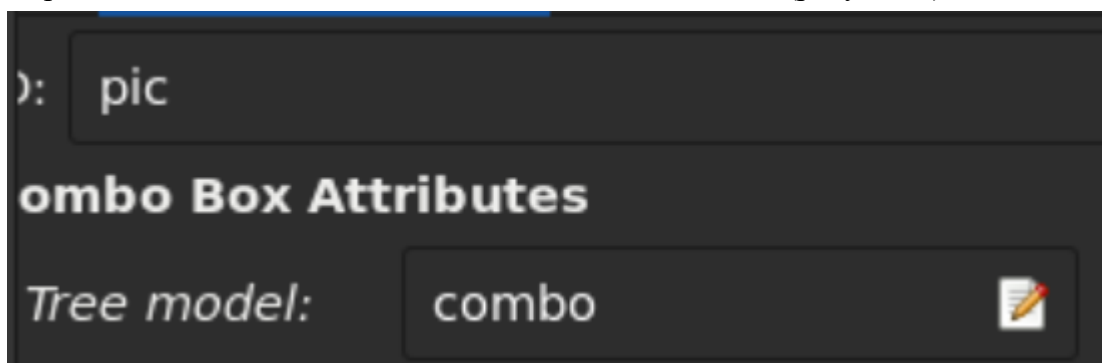


Рисунок 3 – Модель данных в выпадающем списке.

В редакторе выпадающего списка добавляем запись, чтобы данные в нашей модели отображались корректно.



Рисунок 4 – Настройка для отображения.

Создаем модель для списка. Определяем ей поля как строка, число и картинка

ID:

Columns:

Define columns for your liststore; giving them meaningful names will help you to retrieve them when setting cell renderer attributes (press the Delete key to remove the selected column)

Add and remove columns:

Column type	Column name
<input type="text" value="gchararray"/>	<input type="text" value="Name"/>
<input type="text" value="gint64"/>	<input type="text" value="Value"/>
<input type="text" value="GdkPixbuf"/>	<input type="text" value="Pic"/>
<input type="text" value=" < define a new column >"/>	

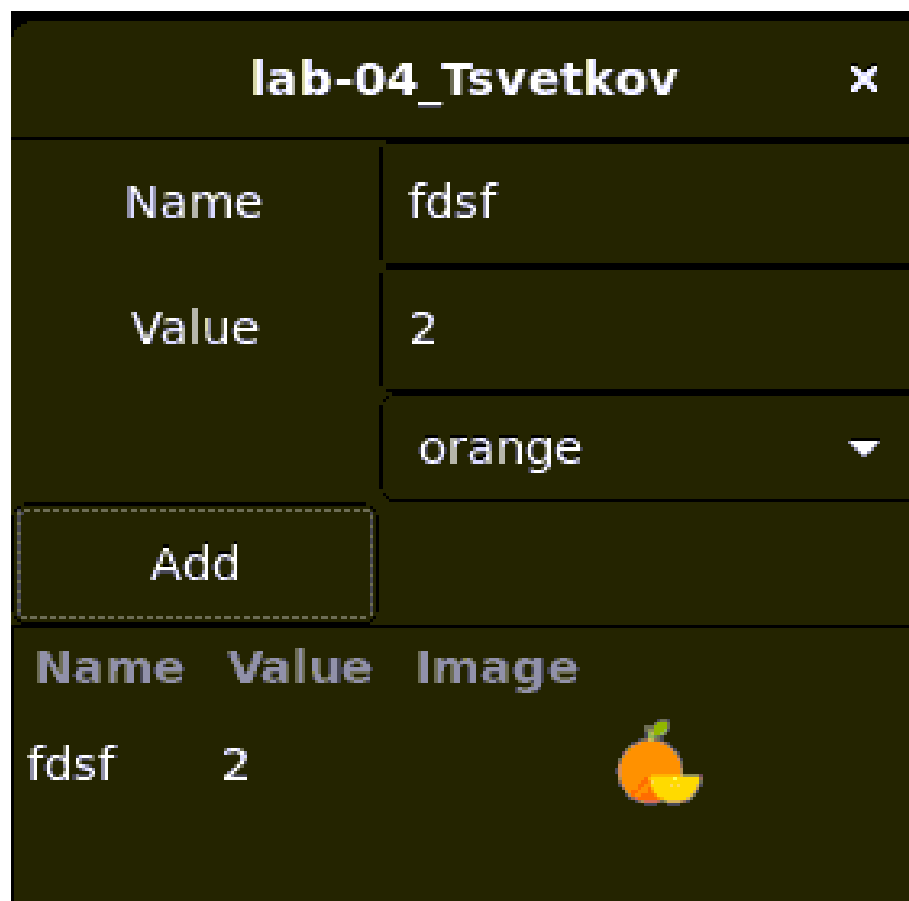
Рисунок 5 - Модель для списка

Для программного заполнения модели был написан следующий код (рисунок 6).

```
lab-04.lua
~/labs/lualabs1love/lab-04

8 bld=gtk.Builder()
9 bld:add_from_file('lab-04.glade')
10 ui=bld.objects
11 prov=gtk.CssProvider()
12 prov:load_from_path('style.css')
13
14 ctx=gtk.StyleContext()
15 scr=gdk.Screen.get_default()
16 ctx.add_provider_for_screen(scr,prov,gtk.STYLE_PROVIDER_PRIORITY_APPLICATION)
17
18 function ui.add_btn:on_clicked(...)
19 i=ui.md_it:append()
20 ui.md_it[i]={[1]=ui.name.text, [2]=tonumber(ui.val.text),
21             [3]=pixbuf.new_from_file(image_arr[ui.pic.active+1])}
22 end
23 rdr_txt=gtk.CellRendererText{}
24 rdr_pix=gtk.CellRendererPixbuf{}
25 c1=gtk.TreeViewColumn{title='Name', {rdr_txt,{text=1}}}
26 c2=gtk.TreeViewColumn{title='Value', {rdr_txt,{text=2}}}
27 c3=gtk.TreeViewColumn{title='Image', {rdr_pix,{pixbuf=3}}}
28
29 ui.list:append_column(c1)
30 ui.list:append_column(c2)
31 ui.list:append_column(c3)
32 ui.wnd.title='lab-04_Tsvetkov'
33 ui.wnd.on_destroy=gtk.main_quit
34 ui.wnd:show_all()
35 gtk.main()
```

Рисунок 6 – Программное заполнение и отображение листа через модель.



Рабочее приложение на рисунке 7.