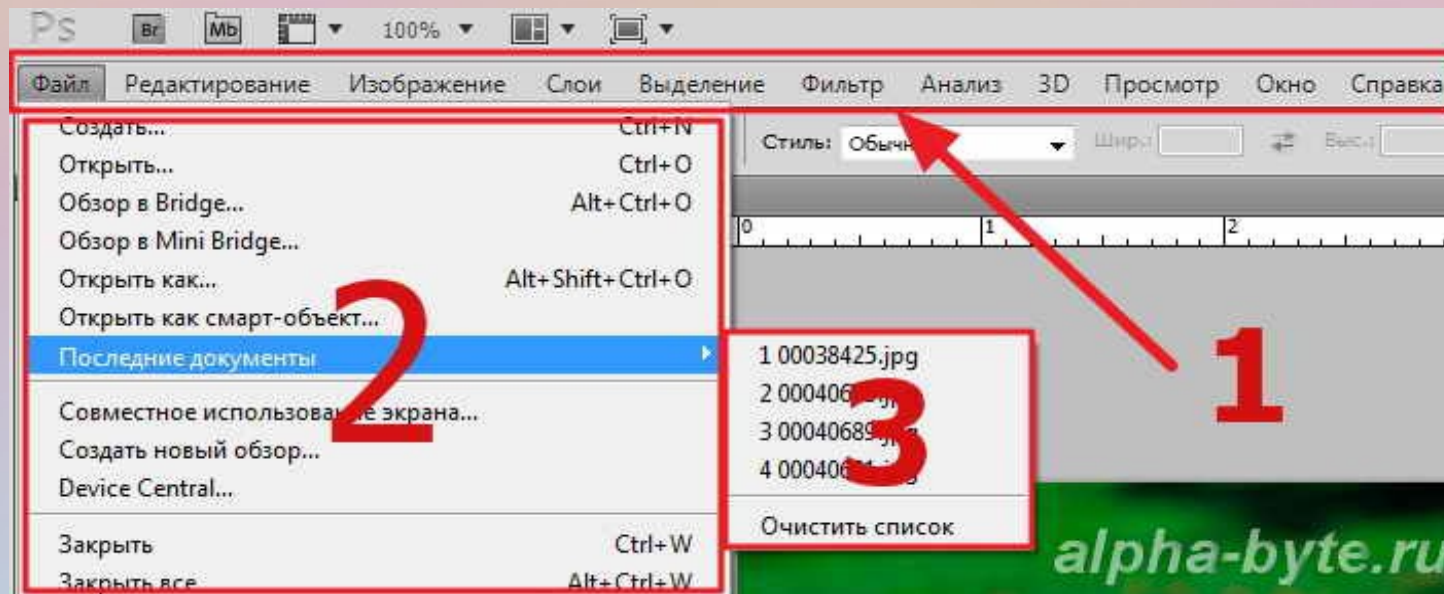


Меню

- Доступ к основным функциональным возможностям программы
- Структура: (1) главное меню; (2) меню; (3) подменю



Главное меню

Строка главного меню, содержит набор меню, элемент графа сцены

```
class MenuBar
```

```
MenuBar()
```

```
MenuBar(Menu... menus)
```

```
ObservableList<Menu> getMenu()
```

Пример:

```
MenuBar menuBar = new MenuBar();
```

```
menuBar.getMenus().addAll(menu1, menu2, menu3);
```

Управление главным меню

ObservableList<Menu> getMenu()

Методы класса ObservableList<Menu>

boolean add(Menu m)

void add(int index, Menu m)// начинается с 0

boolean addAll(Menu... elements)

boolean remove(Menu m)

Menu remove(int index)

Меню и подменю

Стандартное меню, содержащее отдельные пункты или другие подменю

```
class Menu extends MenuItem
```

Конструкторы:

```
Menu()
```

```
Menu(String text)
```

```
Menu(String text, Node graphic)
```

```
Menu(String text, Node graphic, MenuItem...  
items)
```

Методы класса Menu

Получение списка пунктов меню

`ObservableList<MenuItem> getItems()`

Добавление и другие действия с пунктами меню осуществляются с помощью методов списка

Пример:

```
Menu menu1 = new Menu("File");
```

```
MenuItem menu12 = new MenuItem("Open");
```

```
menu1.getItems().add(menu12);
```

Пункты меню

class MenuItem

Конструкторы:

MenuItem()

MenuItem(String text)

MenuItem(String text, Node graphic)

Определение обработчика нажатия на пункт меню:

void setOnAction(EventHandler<ActionEvent>
value)

Определение горячих клавиш и мнемоники (class MenuItem)

Горячие клавиши:

```
void setAccelerator (KeyCombination value)
```

```
open.setAccelerator  
    (KeyCombination.keyCombination  
    ("shortcut+O")); // <Ctrl+O>
```

```
// «alt+O» «shift+O»
```

Мнемоника:

```
Menu fileMenu = new Menu("_File"); //<Alt+F>
```


Пункт меню – кнопка переключатель

class CheckMenuItem extends MenuItem

Конструкторы

CheckMenuItem(String text)

CheckMenuItem()

Методы

boolean isSelected()

void setSelected(boolean value)

BooleanProperty selectedProperty()

Пункт меню – радио-кнопка

class RadioMenuItem extends MenuItem implements Toggle

Конструкторы

RadioMenuItem(String text)

RadioMenuItem()

Методы

void setToggleGroup(ToggleGroup value)

boolean isSelected()

void setSelected(boolean value)

BooleanProperty selectedProperty()

Разделитель

```
class SeparatorMenuItem
```

```
SeparatorMenuItem()
```

Пример:

```
fileMenu.getItems().addAll(open, save, new  
    SeparatorMenuItem(), exit);
```

Контекстное меню

class ContextMenu

Элементы контекстного меню – объекты MenuItem

Конструкторы:

ContextMenu()

ContextMenu(MenuItem... items)

Методы:

ObservableList<MenuItem> getItems()

void show(Node anchor, double screenX, double
screenY)

Вызов контекстного меню

Для любого элемента управления (пакет `javafx.scene.control`):

```
void setContextMenu (ContextMenu value)
```

Для других элементов, например корневого узла сцены (метод класса `Node`):

```
void setOnContextMenuRequested(EventHandler<? super  
ContextMenuEvent> value)
```

Пример:

```
elem.setOnContextMenuRequested((ContextMenuEvent e)  
-> { cmCopyImg.show(elem, e.getScreenX(),e.getScreenY());
```

Кнопки-меню

Классы `MenuButton` и `SplitMenuButton`

Представляют кнопку, при нажатии которой
появляется меню



Панель инструментов **ToolBar**

Содержит кнопки **Buttons**, переключатели **ToggleButton** и разделители **Separators**

Может располагаться по вертикали или по горизонтали

```
void setOrientation(Orientation value)
```

Orientation.HORIZONTAL

Orientation.VERTICAL