**Лабораторная работа 4**

**РАБОТА С ГРАФИКОЙ. РАЗРАБОТКА АППЛЕТОВ**

**Цель работы:** научиться создавать приложения с использованием апплетов

**Задача 1.** Смоделировать работу световой колонны, управ­ляемой с помощью кнопок состояния системы.

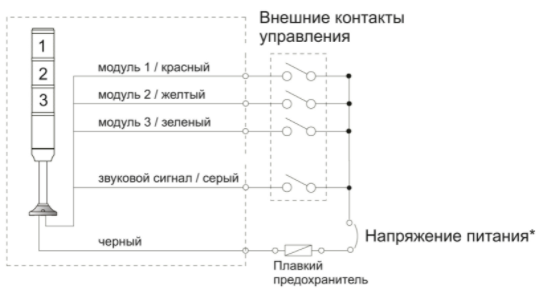
 

Рис1. Световая колонна

Создайте форму для работы световой колонны и запрограммируйте работу

обработку нажатия кнопок, меняя стиль заливки объектов.

Вариант: красный/желтый/зеленый;

**Задача 2.**

Смоделировать работу любого из светодиодов, управляемой с помощью одной кнопки «Вкл/Выкл»

**Задача 3.**

Смена формы фигуры.

Создать фигуру и три кнопки. Нажатие на каждую из них кнопок должно приводить к смене формы фигуры.

Надписи на кнопках должны соответствовать выполняемым действиям.

**Задача 4.**

Смоделируйте движение фигуры влево/вправо при нажатии кнопок.

**Задача 5.**

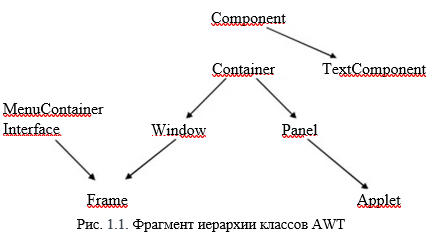
Смоделируйте постепенное надувание шарика, сохраняя нижнюю точку неподвижной.

*Апплеты -* это небольшие приложения. автоматически устанавливаются и выполняются как часть Web-документа. После того, как апплет прибывает к клиенту, он имеет ограниченный доступ к ре­сурсам системы, которые использует для создания произвольного мультиме­дийного интерфейса пользователя и выполнения комплексных вычислений без риска заражения вирусами или нарушения целостности данных. Работа с ап­плетами поддерживается пакетом *AWT.* Этот пакет достаточно большой, что позволяет работать не только с апплетами. Пакет *AWT* содержит многочислен­ные классы и методы, которые позволяют создавать окна и управлять ими.

**Пакет AWT**

*AWT - Abstract Window Toolkit* - абстрактный оконный интерфейс. Ос­новное назначение пакета AWT: поддержка окон апплета и создание обычных GUI-приложений. Классы пакета AWT определяют интерфейсные окна и работу с визуальными компонентами окон и содержатся в пакете java.awt.

На рис. 1.1 представлен фрагмент иерархии классов *AWT.* Практически все классы пакета *AWT* являются потомками абстрактного класса *Component*



Рассмотрим особенности программирования апплетов. В последних лабо­раторных работах рассмотрим более подробно другие классы пакета *AWT.*

**Класс Applet**

Для работы с апплетами предназначен класс *Applet*, который определяет методы, представленные в табл. 1.1. *Applet* обеспечивает всю необходимую поддержку для выполнения апплетов, такую как запуск и остановка. Он также реализует методы, которые загружают и показывают изображения, и методы, которые загружают и проигрывают аудиоклипы.

Рассмотрим особенности программирования апплетов.

Методы, определенные в классе Applet

|  |  |
| --- | --- |
| Метод | Описание |
| void destroy() | Освобождает все ресурсы, занятые апплетом. Вызы­вается браузером непосредственно перед тем, как ап­плет завершается.  Метод *destroy()* вызывается, когда среда решает, что апплет должен быть полностью удален из памяти. В этот момент следует освободить любые ресурсы, ко­торые апплет может использовать |
| String getParame- ter(String paramName) | Возвращает параметр, указанный в *paramName.* Если указанный параметр не найден, возвращается nu*ll* (пустой указатель) |
| void init() | Вызывается, когда апплет начинает выполнение. Это первый метод, который вызывается для любого ап­плета. В нем необходимо инициализировать пере­менные. Вызывается этот метод один раз в течение времени выполнения апплета |
| boolean isActive() | Возвращает *true,* если апплет был запущен. Возвраща­ет *false,* если апплет был остановлен |
| void resize(Dimension dim) | Изменяет размеры апплета согласно измерениям, ука­занным в *dim* |
| void start() | Вызывается, чтобы перезапустить апплет после его остановки. В то время как *init()* вызывается один раз (когда апплет загружается), *start()* запускается каждый раз, когда *HTML*-документ апплета отображается на эк­ране. Так, если пользователь покидает *Web*-страницу и возвращается обратно, апплет возобновляет выполне­ние в *start().* |
| void stop() | Метод *stop()* вызывается, если Web-браузер покидает HTML-документ, содержащий апплет, при переходе к другой странице. Когда вызывается *stop(),* апплет, ве­роятно, продолжает выполняться. Следует использо­вать *stop()* для приостановки потоков, не требующих выполнения, если апплет невидим. Их можно переза­пустить вызывом *start()*, когда пользователь возвра­щается к странице.  Метод *stop()* всегда вызывается перед *destroy()* |

Инициализация и завершение апплета

Важно понять порядок, в котором вызываются различные методы аппле­та. Когда апплет начинает выполняться. *AWT* вызывает методы в такой после­довательности:

1) *init();* 2) *start();* 3) *paint();*

При завершении апплета имеет место следующая последовательность вызовов:

1) *stop();* 2) *destroy();*

**Запуск апплетов**

Апплет можно запускать как из HTML-документа, так и из программы просмотра апплета. Для этого используется тег *<applet>* языка *HTML.* Про­грамма просмотра апплета выполняет каждый *<applet*>-тег*,* который она нахо­дит, в отдельном окне, в то время как Web-браузеры могут допускать много апплетов на одной странице.

Синтаксис тега *<applet>.* Параметры в квадратных скобках не обязательны.

</applet>

[CODEBASE = codebaseURL]

CODЕ = appletFile

[ALT = altemateText]

[NAME = appletlnstanceName]

[WIDTH = pixels HEIGHT = pixels]

[ALIGN = alignment]

[VSPACE = pixels] [HSPACE = pixels]

…

</applet>