# Что такое GUI?

Это интерфейс, который позволяет пользователям взаимодействовать с программным обеспечением через графические элементы, такие как окна, кнопки, текстовые поля, меню и другие визуальные компоненты.

# Какие два типа компонентов GUI существует?

Контейнеры - компоненты, которые могут содержать другие компоненты. Контейнеры используются для организации и размещения элементов интерфейса.

Компоненты - элементы интерфейса, которые непосредственно взаимодействуют с пользователем. Они могут обрабатывать ввод и отображать информацию.

# Что такое элементы управления и что такое контейнеры?

Элементы управления / управляемые компоненты — это компоненты графического интерфейса, которые позволяют пользователям взаимодействовать с приложением. Они могут обрабатывать ввод пользователя и выполнять определенные действия.

Контейнеры — это специальные компоненты, которые могут содержать другие элементы управления и организовывать их в определенном порядке. Контейнеры помогают структурировать интерфейс, группируя элементы и управляя их размещением.

# Какие классы элементов управления существуют?

1. Button - кнопки
2. Label - надписи
3. TextField - однострочное текстовое поле
4. TextArea - многострочное текстовое поле
5. CheckBox - флаги
6. RadioButton - переключатели
7. List - выпадающий список с возможностью выбрать несколько элементов
8. Choice - выпадающий список с возможностью выбрать один элемент
9. Scrollbar – полоса прокрутки

# Что необходимо сделать, чтобы ввести компонент в контейнер?

1. Создать контейнер
2. Создать компонент
3. Добавить компонент в контейнер: Используйте метод add() контейнера

# Как можно перехватить и обработать события, пришедшие от компонентов?

Чтобы перехватить и обработать события, пришедшие от компонентов в Java апплетах, необходимо следовать следующим шагам:

1. Импортировать нужные классы
2. Создать компоненты
3. Создать слушатели событий: реализовать интерфейсы слушателей событий, такие как ActionListener, KeyListener, MouseListener
4. Добавить слушатели к компонентам: использовать метод addActionListener() (или аналогичный метод для других типов слушателей) для добавления слушателя к компоненту.
5. Обработать события - реализовать методы обработки событий, такие как actionPerformed() для ActionListener, чтобы определить, что должно произойти при возникновении события.

# Какие события от компонентов можно обработать при помощи метода action()

1. Нажатие кнопки
2. Выбор элемента в списке (List)
3. Выбор элемента в меню (MenuItem)
4. Двойное нажатие (Double Click)
5. Изменение состояния (Toggle Button)

# Если событие не обрабатывается методом action(), то какой метод используется для обработки события?

1. Для клавиатурных событий (интерфейс KeyListener):
   1. keyPressed: вызывается при нажатии клавиши.
   2. keyReleased: вызывается при отпускании клавиши.
   3. keyTyped: вызывается при вводе символа.
2. Для мышиных событий (интерфейс MouseListener):
   1. mouseClicked - вызывается при щелчке мыши.
   2. mousePressed - вызывается при нажатии кнопки мыши.
   3. mouseReleased - вызывается при отпускании кнопки мыши.
   4. mouseEntered - вызывается, когда курсор входит в область компонента.
   5. mouseExited - вызывается, когда курсор покидает область компонента.
3. Для изменения состояния компонентов (интерфейс ItemListener):
   1. itemStateChanged - вызывается при изменении состояния элемента
4. Для обработки действий, связанных с изменением текста (интерфейс DocumentListener)

# Какие типы переключателей существуют?

1. RadioButton - тип переключателя, который позволяет пользователю выбрать только один вариант из набора вариантов
2. CheckBox - тип переключателя, который позволяет пользователю выбрать один или несколько вариантов из набора вариантов.
3. ToggleButton - тип переключателя, который позволяет пользователю переключать между двумя состояниями: включенным и выключенным.
4. Choice - тип переключателя, который позволяет пользователю выбрать один вариант из набора вариантов, представленных в виде раскрывающегося списка
5. List - тип переключателя, который позволяет пользователю выбрать один или несколько вариантов из набора вариантов, представленных в виде списка.
6. ComboBox - тип переключателя, который позволяет пользователю выбрать один вариант из набора вариантов, представленных в виде раскрывающегося списка.
7. ButtonGroup - тип переключателя, который позволяет группировать несколько кнопок вместе, чтобы пользователь мог выбрать только одну из них.

# Как несколько переключателей объединить в группу?

Можно объединить несколько переключателей в группу с помощью класса ButtonGroup. Этот класс позволяет группировать несколько кнопок вместе, чтобы пользователь мог выбрать только одну из них.

# Чем отличаются выпадающие и раскрывающиеся списки? Как осуществляется в них выбор элементов?

Выпадающий список (ComboBox) — это элемент управления, который отображает только один элемент списка по умолчанию, а остальные элементы списка скрыты. Пользователь может нажать на кнопку со стрелкой вниз, чтобы раскрыть список и выбрать один из элементов. После выбора элемента список закрывается, и выбранный элемент отображается в поле списка.

Раскрывающийся список (List) — это элемент управления, который отображает все элементы списка в виде списка. Пользователь может выбрать один или несколько элементов из списка.

# При помощи какого метода обрабатываются сообщения от линейки прокрутки?

При помощи метода AdjustmentListener обрабатываются сообщения от линейки прокрутки.

AdjustmentListener - это интерфейс, который предоставляет метод adjustmentValueChanged(), который вызывается при изменении значения линейки прокрутки.

# Порождают ли события компоненты класса Label и для чего используются эти компоненты? В чем преимущества их использования перед обычным выводом текста методами класса Graphics?

Компоненты класса Label не порождают события, поскольку они не являются интерактивными элементами. Они используются для отображения статического текста или изображений в приложении.

# Что такое текстовые поля и текстовые области? Чем они отличаются?

Текстовые поля (TextField) - это компоненты, которые позволяют пользователю вводить одну строку текста.

Текстовые области (TextArea) - это компоненты, которые позволяют пользователю вводить несколько строк текста.