

ЛАБОРАТОРНАЯ РАБОТА № 1

ТЕМА: РАЗРАБОТКА ИНТЕРАКТИВНОГО ПРИЛОЖЕНИЯ С ПРИМЕНЕНИЕМ ОСНОВНЫХ КОМПОНЕНТОВ ДЛЯ ПОСТРОЕНИЯ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА

Общее задание:

Для реализации приложения необходимо использовать язык Java и одну из графических библиотек SWING/ SWT/ JavaFx. Тип графической библиотеки назначается преподавателем. **Нельзя пользоваться редакторами форм!!!!**

Общее задание для всех студентов.

Создать окно, которое содержит пять групп элементов управления. Для группировки использовать один из компонент JPanel(Swing)/ Composite(SWT)/ Pane(JavaFx).

Индивидуальное задание для студента

После выполнения 5 групп элементов, преподавателем выдается дополнительное 6-е задание из предложенного списка вариантов.

Описание 5 групп элементов для общего задания.

1) Группа состоит из элементов JTextField, JComboBox, JButton. Пользователь заносит текст в JTextField, затем нажимает элемент JButton, после чего текст из JTextField заносится в элемент JComboBox. Если вносимый текст уже существует в JComboBox, то при добавлении должно быть выведено диалоговое окно о невозможности добавления введенного текста.

2) Группа состоит из элементов JTextField, JButton1, JButton2. Пользователь заносит текст в JTextField, затем нажимает элемент JButton1, после чего текст из JTextField отображается на элементе JButton2. При нажатии на JButton2 тексты на кнопках меняются местами.

3) Группа состоит из элементов JTextField, JButton, JRadioButton1, JRadioButton2, JRadioButton3. Пользователь заносит имя одного из элементов типа JRadioButton в JTextField, затем нажимает элемент JButton, после чего соответствующий элемент помечается, причем со всех остальных пометки должны сниматься. При повторном нажатии на JButton метка с уже выделенного элемента JRadioButton не должна исчезать. Если введено имя JRadioButton, которого не существует, то должно быть выведено диалоговое окно с ошибкой.

4) Группа состоит из элементов JTextField, JButton, JCheckBox1, JCheckBox2, JCheckBox3. Пользователь заносит имя одного из элементов JCheckBox в JTextField, затем нажимает элемент JButton, после чего соответствующий элемент помечается, если он не был помечен или с соответствующего элемента снимается метка, если он был помечен. Метки оставшихся двух компонент не должны менять свое состояние. Если введено имя JCheckBox, которого не существует, то должно быть выведено диалоговое окно с ошибкой.

5) Группа состоит из элементов JTextField, JButton1, JButton2, JButton3, JTable. Элемент JTable содержит два столбца. Пользователь заносит текст в JTextField, затем нажимает элемент JButton1, после чего текст из JTextField заносится в первый столбец элемента JTable. Затем нажимает элемент JButton2, после чего выделенный текст из первого столбца JTable переносится во второй столбец элемента JTable в той же строке, затем нажимает элемент JButton3, после чего выделенный текст из второго столбца JTable переносится в первый столбец элемента JTable в той же строке. В элементе управления JTable в каждой строке один столбец всегда остается пустым. Если выделенная для переноса ячейка содержит пустое значение, то при нажатии на соответствующий элемент JButton2 или JButton3, ничего не должно происходить.

Все кнопки и элементы управления должны иметь осмысленные названия. Для размещения элементов управления в рамках групп необходимо использовать компоновщики, предоставляемые вашей библиотекой. Для вывода диалогов с сообщением об ошибке использовать статические методы класса JOptionPane.

Варианты индивидуального задания:

Общее замечание: Все индивидуальные варианты написана для библиотеки SWING. При реализации заменять компоненты библиотеки SWING на компоненты из вашей библиотеки. Например JButton(SWING) -> Button(SWT).

1. Реализовать бегущую строку в заголовке окна.
2. Реализовать бегущую строку во всех названия кнопок.
3. Реализовать бегущую строку во всех JLabel компонентах, если такие имеются.
4. Реализовать менеджер компоновки, который автоматически размещает элементы любой панели или окна по Эллипсу.
5. Изменить рисование стандартной кнопки на кнопку в виде круга. Кнопка должна менять цвет при попадании мыши в область выбранной фигуры. Все кнопки в окне меняются на новую.
6. Изменить рисование стандартной кнопки на кнопку в виде ромба. Кнопка должна менять цвет при попадании мыши в область выбранной фигуры. Все кнопки в окне меняются на новую.
7. Изменить рисование стандартной кнопки на кнопку в виде бесконечности. Кнопка должна менять цвет при попадании мыши в область выбранной фигуры. Все кнопки в окне меняются на новую.
8. Изменить рисование стандартной кнопки на кнопку в виде восьмерки. Кнопка должна менять цвет при попадании мыши в область выбранной фигуры. Все кнопки в окне меняются на новую.
9. Изменить рисование стандартной кнопки на кнопку в виде эллипса. Кнопка должна менять цвет при попадании мыши в область выбранной фигуры. Все кнопки в окне меняются на новую.
10. Реализовать группу 6 состоящую из следующих компонентов JTextField, JButton1, JButton2, JButton3. В JTextField будет вводиться число

JCheckBox, которые будут сгенерированы в окне, после нажатия на кнопку JButton1. При нажатии JButton2, JCheckbox будут выделяться с задержкой в 500мс по алгоритму: выделяется 1-й, затем последний, затем 2-й, затем предпоследний и т.д. Когда все элементы JCheckbox будут выделены, они по такому же алгоритму начинают терять выделение, начиная с 1-го элемента. При нажатии JButton3 выделение будет останавливаться. При повторном нажатии на JButton2 выделение будет продолжаться с того места, где было остановлено выделение компонентов в предыдущий раз. При повторном нажатии на JButton1 генерируется новое заданное количество JCheckbox.

11. Реализовать группу 6 состоящую из следующих компонентов JTextField, JButton1, JButton2, JButton3. В JTextField будет вводиться число JCheckBox, которые будут сгенерированы в окне, после нажатия на кнопку JButton1. При нажатии JButton2, JCheckbox будут выделяться с задержкой в 500мс по алгоритму: выделяется 1-й и последний, затем 2-й и предпоследний и т.д. Когда все элементы JCheckbox будут выделены, они по такому же алгоритму начинают терять выделение, начиная с 1-го элемента и последнего компонента. При нажатии JButton3 выделение будет останавливаться. При повторном нажатии на JButton2 выделение будет продолжаться с того места, где было остановлено выделение компонентов в предыдущий раз. При повторном нажатии на JButton1 генерируется новое заданное количество JCheckbox.

12. Реализовать группу 6 состоящую из следующих компонентов JTextField, JButton1, JButton2, JButton3. В JTextField будет вводиться число JCheckBox, которые будут сгенерированы в окне, после нажатия на кнопку JButton1. При нажатии JButton2, JCheckbox будут выделяться с задержкой в 500мс по алгоритму: выделяется 1-й, затем 2-й, затем 3-й и т.д. Когда все элементы JCheckbox будут выделены, они по такому же алгоритму начинают терять выделение, начиная с 1-го элемента. При нажатии JButton3 выделение будет останавливаться. При повторном нажатии на JButton2 выделение будет продолжаться с того места, где было остановлено выделение компонентов в предыдущий раз. При повторном нажатии на JButton1 генерируется новое заданное количество JCheckbox.

13. Реализовать группу 6 состоящую из следующих компонентов JTextField, JButton1, JButton2, JButton3. В JTextField будет вводиться число JRadioButton, которые будут сгенерированы в окне, после нажатия на кнопку JButton1. При нажатии JButton2, JRadioButton будут выделяться с задержкой в 500мс по алгоритму: выделяется 1-й, затем последний, затем 2-й, затем предпоследний и т.д. Когда все элементы JRadioButton будут обработаны, они по такому же алгоритму начинают выделяться, начиная с 1-го элемента. При нажатии JButton3 выделение будет останавливаться. При повторном нажатии на JButton2 выделение будет продолжаться с того места, где было остановлено выделение компонентов в предыдущий раз. При повторном нажатии на JButton1 генерируется новое заданное количество JRadioButton.

14. Реализовать группу 6 состоящую из следующих компонентов JTextField, JButton1, JButton2, JButton3. В JTextField будет вводиться число

JRadioButton, которые будут сгенерированы в окне, после нажатия на кнопку JButton1. При нажатии JButton2, JRadioButton будут выделяться с задержкой в 500мс по алгоритму: выделяется 1-й, затем 2-й, затем 3-й и т.д. Когда все элементы JRadioButton будут обработаны, они по такому же алгоритму начинают выделяться, начиная с 1-го элемента. При нажатии JButton3 выделение будет останавливаться. При повторном нажатии на JButton2 выделение будет продолжаться с того места, где было остановлено выделение компонентов в предыдущий раз. При повторном нажатии на JButton1 генерируется новое заданное количество JRadioButton.

15. Реализовать группу 6 состоящую из следующих компонентов JTextField1, JTextField2, JButton1, JButton2, JButton3. В JTextField1 и JTextField2 будет вводиться число строк и столбцов, которые задают размер сгенерированной в будущей таблице, после нажатия на кнопку JButton1. Также при нажатии на JButton1, 2 случайные ячейки таблицы будут заполняться некоторым текстом. При нажатии JButton2, текст из ячеек будет перемещаться в другие ячейки с задержкой в 500мс по алгоритму: сначала на две ячейки вниз, затем на одну ячейку вправо. Если текст выходит за границы в столбце, то он появляется в этом же столбце в 1-й ячейке сверху. Если текст выходит за границу справа, то он появляется в этой же строке в 1-й ячейке слева. Если в ячейку приходит текст из двух ячеек, то в ячейке показывается только 1 текст. При нажатии JButton3 перемещение текста будет останавливаться. При повторном нажатии на JButton2 перемещение текста будет продолжаться с того места, где было остановлено. При повторном нажатии на JButton1 генерируется новая таблица.

16. Реализовать группу 6 состоящую из следующих компонентов JTextField1, JTextField2, JButton1, JButton2, JButton3. В JTextField1 и JTextField2 будет вводиться число строк и столбцов, которые задают размер сгенерированной в будущей таблице, после нажатия на кнопку JButton1. Также при нажатии на JButton1, 4 случайные ячейки таблицы будут заполняться некоторым текстом. При нажатии JButton2, текст из ячеек будет перемещаться в другие ячейки с задержкой в 500мс по алгоритму: сначала на две ячейки вправо, затем на одну ячейку вверх. Если текст выходит за границы в столбце, то он появляется в этом же столбце в 1-й ячейке снизу. Если текст выходит за границу справа, то он появляется в этой же строке в 1-й ячейке слева. Если в ячейку приходит текст из двух ячеек, то в ячейке показывается только 1 текст. При нажатии JButton3 перемещение текста будет останавливаться. При повторном нажатии на JButton2 перемещение текста будет продолжаться с того места, где было остановлено. При повторном нажатии на JButton1 генерируется новая таблица.

17. Реализовать группу 6 состоящую из следующих компонентов JTextField1, JTextField2, JButton1, JButton2, JButton3. В JTextField1 и JTextField2 будет вводиться число строк и столбцов, которые задают размер сгенерированной в будущей таблице, после нажатия на кнопку JButton1. Также при нажатии на JButton1, 1 случайная ячейка таблицы будет заполняться некоторым текстом. При нажатии JButton2, текст из ячейки

будет перемещаться в другие ячейки с задержкой в 500мс по алгоритму змейки: текст двигается вправо до конца строки, затем перемещается на одну ячейку вниз и двигается влево до конца строки, затем смещается вниз на одну ячейку и снова двигается вправо. Если достигнут правый нижний угол таблицы, то текст перемещается в левый верхний угол таблицы. При нажатии JButton3 перемещение текста будет останавливаться. При повторном нажатии на JButton2 перемещение текста будет продолжаться с того места, где было остановлено. При повторном нажатии на JButton1 генерируется новая таблица.

18. Реализовать группу 6 состоящую из следующих компонентов JTextField1, JTextField2, JButton1, JButton2, JButton3. В JTextField1 и JTextField2 будет вводиться число строк и столбцов, которые задают размер сгенерированной в будущей таблице, после нажатия на кнопку JButton1. Также при нажатии на JButton1, 1 случайная ячейка таблицы будет заполняться некоторым текстом. При нажатии JButton2, текст из ячейки будет перемещаться в другие ячейки с задержкой в 500мс по алгоритму змейки: текст двигается вниз до конца столбца, затем перемещается на одну ячейку вправо и двигается вверх до конца столбца, затем смещается вправо на одну ячейку и снова двигается вниз. Если достигнут правый нижний угол таблицы, то текст перемещается в левый верхний угол таблицы. При нажатии JButton3 перемещение текста будет останавливаться. При повторном нажатии на JButton2 перемещение текста будет продолжаться с того места, где было остановлено. При повторном нажатии на JButton1 генерируется новая таблица.

19. Реализовать две горячие комбинации клавиш. Например, ctrl+R, ctrl+S. При нажатии на первую комбинацию(ctrl+R): основное окно приложения, разбивается на 5 окон в каждом из которых расположено по одной группе элементов. Через 1 секунду 5 новых окон размещаются по кругу на экране. Затем 5 окон переставляются по очереди между собой по часовой стрелке с задержкой в 1 секунду. После окончания перестановки через 1 секунду на экране снова появляется основное окно, затем описанный процесс повторяется вновь. При нажатии второй комбинации клавиш(ctrl+S) описанный процесс останавливается в любой момент времени. Если повторно нажать вторую комбинацию (ctrl+R), то процесс продолжится с того места, где был остановлен.

20. Реализовать две горячие комбинации клавиш. Например, ctrl+R, ctrl+S. При нажатии на первую комбинацию(ctrl+R): основное окно приложения, разбивается на 5 окон в каждом из которых расположено по одной группе элементов. Затем новые окна по одному начинают показываться на экране с задержкой в 1 секунду, начиная с окна, содержащего 1-ю группу компонент. Когда процесс дошел до окна с 5-й группой компонент, то все начинает повторяться с окна, содержащего 1-ю группу компонент. При нажатии второй комбинации клавиш(ctrl+S) описанный процесс останавливается в любой момент времени. Если

повторно нажать вторую комбинацию (ctrl+R), то процесс продолжится с того места, где был остановлен.

21. Реализовать две горячие комбинации клавиш. Например, ctrl+R, ctrl+S. При нажатии на первую комбинацию(ctrl+R): основное окно приложения, разбивается на 5 окон в каждом из которых расположено по одной группе элементов. Через 1 секунду 5 новых окон размещаются по кругу на экране. Затем 5 окон по очереди пропадают с экрана по часовой стрелке с задержкой в 1 секунду. После того, как на экране не осталось окон, через 1 секунду на экране появляется основное окно, затем описанный процесс повторяется вновь. При нажатии второй комбинации клавиш(ctrl+S) описанный процесс останавливается в любой момент времени. Если повторно нажать вторую комбинацию (ctrl+R), то процесс продолжится с того места, где был остановлен.

22. Реализовать две горячие комбинации клавиш. Например, ctrl+R, ctrl+S. При нажатии на первую комбинацию(ctrl+R): основное окно приложения, разбивается на 5 окон в каждом из которых расположено по одной группе элементов. Через 1 секунду после нажатия первой комбинации клавиш (ctrl+R), главное окно пропадает с экрана, затем через 1 секунду на экране начинают по очереди показываться новые окна. Новые окна появляются с задержкой в 1 секунду начиная с окна, содержащего 1-ю группу компонент. Через 1 секунду после того, как на экране появилось окно с 5-й группой компонент, 5 окон прячутся и на экране отображается основное окно, затем процесс повторяется вновь. При нажатии второй комбинации клавиш(ctrl+S) описанный процесс останавливается в любой момент времени. Если повторно нажать вторую комбинацию (ctrl+R), то процесс продолжится с того места, где был остановлен.

23. Разукрасить фон 5 групп элементов в 5 различных цветов. Затем реализовать две горячие комбинации клавиш. Например, ctrl+R, ctrl+S. При нажатии на первую комбинацию(ctrl+R) цвета групп начинают по очереди меняться местами: 1-й становится 2-м, 2-й становится 3-м и т.д., последний становится 1-м. Смена цвета происходит с задержкой в 1 секунду. При нажатии второй комбинации клавиш(ctrl+S) описанный процесс останавливается в любой момент времени. Если повторно нажать вторую комбинацию (ctrl+R), то процесс продолжится с того места, где был остановлен.

24. Реализовать две горячие комбинации клавиш. Например, ctrl+R, ctrl+S. При нажатии на первую комбинацию(ctrl+R) группы компонент начинают по очереди меняться местами: 1-я становится 2-й, 2-я становится 3-ей и т.д., последняя становится 1-ой. Смена происходит с задержкой в 1 секунду. При нажатии второй комбинации клавиш(ctrl+S) описанный процесс останавливается в любой момент времени. Если повторно нажать вторую комбинацию (ctrl+R), то процесс продолжится с того места, где был остановлен.

25. Реализовать две горячие комбинации клавиш. Например, ctrl+R, ctrl+S. При нажатии на первую комбинацию(ctrl+R) компоненты внутри

группы компонент начинают по очереди меняться местами. Алгоритм применяется для всех групп компонент. Пример одной итерации для компонентов из группы 1: JTextField занимает место JComboBox, JComboBox занимает место JButton, JButton занимает место JTextField. Смена происходит с задержкой в 1 секунду. При нажатии второй комбинации клавиш(ctrl+S) описанный процесс останавливается в любой момент времени. Если повторно нажать вторую комбинацию (ctrl+R), то процесс продолжится с того места, где был остановлен.

26. Реализовать две горячие комбинации клавиш. Например, ctrl+R, ctrl+S. При нажатии на первую комбинацию(ctrl+R) компоненты всего окна по очереди меняются местами. Пример одной итерации: JTextField занимает место JComboBox, JComboBox занимает место JButton, JButton занимает место JTextField из группы 2 и т.д. JTable из групп 5 занимает место JTextField из групп 1. Смена происходит с задержкой в 1 секунду. При нажатии второй комбинации клавиш(ctrl+S) описанный процесс останавливается в любой момент времени. Если повторно нажать вторую комбинацию (ctrl+R), то процесс продолжится с того места, где был остановлен.