Class PhysicsCalc

java.lang.Object
  java.awt.Component
    java.awt.Container
      java.awt.Window
        java.awt.Frame
          javax.swing.JFrame
            PhysicsCalc

All Implemented Interfaces:

`ActionListener`, `ImageObserver`, `MenuContainer`, `Serializable`, `EventListener`, `Accessible`, `RootPaneContainer`, `WindowConstants`

```
public class PhysicsCalc
extends JFrame
implements ActionListener
```

The PhysicsCalc program implements a GUI that creates a complex calculator using the Calculate class methods.

User can choose between the three formulas to solve a kinematics question that is asking for a specific variable through the use of Calculate, by inputting given variables from the question.

Since:

2020-12-0

See Also:

Serialized Form

- ***Nested Class Summary***

  Nested classes/interfaces inherited from class javax.swing.JFrame

  `JFrame.AccessibleJFrame`

  Nested classes/interfaces inherited from class java.awt.Frame

  `Frame.AccessibleAWTFrame`

  Nested classes/interfaces inherited from class java.awt.Window

  `Window.AccessibleAWTWindow`, `Window.Type`

Nested classes/interfaces inherited from class java.awt.Container

`Container.AccessibleAWTContainer`

Nested classes/interfaces inherited from class java.awt.Component

`Component.AccessibleAWTComponent`, `Component.BaselineResizeBehavior`, `Component.BltBufferStrategy`, `Component.FlipBufferStrategy`

- ### *Field Summary*

Modifier and Type	Field	Description
`private JTextField`	`answer`	Declares a JTextField variables to input numbers.
`private Calculate`	`calc`	Declares a Calculate variable to use methods.
`private double`	`missing`	Declares a missing variable to be set to the missing number in the equation.
`private JTextField`	`numberAcc`	Declares a JTextField variables to input numbers.
`private JTextField`	`numberDis`	Declares a JTextField variables to input numbers.

private JTextField	numberIniVel	Declares a JTextField variables to input numbers.
private JTextField	numberTime	Declares a JTextField variables to input numbers.
private JTextField	numberVel	Declares a JTextField variables to input numbers.

- **Fields inherited from class javax.swing.JFrame**
  `accessibleContext, rootPane, rootPaneCheckingEnabled`
  Fields inherited from class java.awt.Frame
  `CROSSHAIR_CURSOR, DEFAULT_CURSOR, E_RESIZE_CURSOR, HAND_CURSOR, ICONIFIED, MAXIMIZED_BOTH, MAXIMIZED_HORIZ, MAXIMIZED_VERT, MOVE_CURSOR, N_RESIZE_CURSOR, NE_RESIZE_CURSOR, NORMAL, NW_RESIZE_CURSOR, S_RESIZE_CURSOR, SE_RESIZE_CURSOR, SW_RESIZE_CURSOR, TEXT_CURSOR, W_RESIZE_CURSOR, WAIT_CURSOR`
  Fields inherited from class java.awt.Component
  `BOTTOM_ALIGNMENT, CENTER_ALIGNMENT, LEFT_ALIGNMENT, RIGHT_ALIGNMENT, TOP_ALIGNMENT`
  Fields inherited from interface java.awt.image.ImageObserver
  `ABORT, ALLBITS, ERROR, FRAMEBITS, HEIGHT, PROPERTIES, SOMEBITS, WIDTH`
  Fields inherited from interface javax.swing.WindowConstants
  `DISPOSE_ON_CLOSE, DO_NOTHING_ON_CLOSE, EXIT_ON_CLOSE, HIDE_ON_CLOSE`
- ***Constructor Summary***

Constructor	Description
`PhysicsCalc()`	This contructor creates a Content Pane with a Flow Layout to put the different JLabels, JTextFields, and JButtons into.

-

- ## *Method Summary*

All MethodsStatic MethodsInstance MethodsConcrete Methods

Modifier and Type	Method	Description
void	`actionPerformed(ActionEvent e)`	This method uses Action Listener for the buttons and calls on the equation methods to find the missing variable.
static void	`main(String[] args)`	Main method to create a new PhysicsCalc object, Runs a window for the calculator and sets parameters and colors.

`o`
`i`
`d`

- ### Methods inherited from class javax.swing.JFrame

  `addImpl`, `createRootPane`, `frameInit`, `getAccessibleContext`, `getContentPane`, `getDefaultCloseOperation`, `getGlassPane`, `getGraphics`, `getJMenuBar`, `getLayeredPane`, `getRootPane`, `getTransferHandler`, `isDefaultLookAndFeelDecorated`, `isRootPaneCheckingEnabled`, `paramString`, `processWindowEvent`, `remove`, `repaint`, `setContentPane`, `setDefaultCloseOperation`, `setDefaultLookAndFeelDecorated`, `setGlassPane`, `setIconImage`, `setJMenuBar`, `setLayeredPane`, `setLayout`, `setRootPane`, `setRootPaneCheckingEnabled`, `setTransferHandler`, `update`

  ### Methods inherited from class java.awt.Frame

  `addNotify`, `getCursorType`, `getExtendedState`, `getFrames`, `getIconImage`, `getMaximizedBounds`, `getMenuBar`, `getState`, `getTitle`, `isResizable`, `isUndecorated`, `remove`, `removeNotify`, `setBackground`, `setCursor`, `setExtendedState`, `setMaximizedBounds`, `setMenuBar`, `setOpacity`, `setResizable`, `setShape`, `setState`, `setTitle`, `setUndecorated`

  ### Methods inherited from class java.awt.Window

  `addPropertyChangeListener`, `addPropertyChangeListener`, `addWindowFocusListener`, `addWindowListener`, `addWindowStateListener`, `applyResourceBundle`, `applyResourceBundle`, `createBufferStrategy`, `createBufferStrategy`, `dispose`, `getBackground`, `getBufferStrategy`, `getFocusableWindowState`, `getFocusCycleRootAncestor`, `getFocusOwner`, `getFocusTraversalKeys`, `getIconImages`, `getInputContext`, `getListeners`, `getLocale`, `getModalExclusionType`, `getMostRecentFocusOwner`, `getOpacity`, `getOwnedWindows`, `getOwner`, `getOwnerlessWindows`, `getShape`, `getToolkit`, `getType`, `getWarningString`, `getWindowFocusListeners`, `getWindowListeners`, `getWindows`, `getWindowStateListeners`, `hide`, `isActive`, `isAlwaysOnTop`, `isAlwaysOnTopSupported`, `isAutoRequestFocus`, `isFocusableWindow`, `isFocusCycleRoot`, `isFocused`, `isLocationByPlatform`, `isOpaque`, `isShowing`, `isValidateRoot`, `pack`, `paint`, `postEvent`, `processEvent`, `processWindowFocusEvent`, `processWindowStateEvent`, `removeWindowFocusListener`, `removeWindowListener`,

removeWindowStateListener, reshape, setAlwaysOnTop, setAutoRequestFocus, setBounds, setBounds, setCursor, setFocusableWindowState, setFocusCycleRoot, setIconImages, setLocation, setLocation, setLocationByPlatform, setLocationRelativeTo, setMinimumSize, setModalExclusionType, setSize, setSize, setType, setVisible, show, toBack, toFront

Methods inherited from class java.awt.Container

add, add, add, add, add, addContainerListener, applyComponentOrientation, areFocusTraversalKeysSet, countComponents, deliverEvent, doLayout, findComponentAt, findComponentAt, getAlignmentX, getAlignmentY, getComponent, getComponentAt, getComponentAt, getComponentCount, getComponents, getComponentZOrder, getContainerListeners, getFocusTraversalPolicy, getInsets, getLayout, getMaximumSize, getMinimumSize, getMousePosition, getPreferredSize, insets, invalidate, isAncestorOf, isFocusCycleRoot, isFocusTraversalPolicyProvider, isFocusTraversalPolicySet, layout, list, list, locate, minimumSize, paintComponents, preferredSize, print, printComponents, processContainerEvent, remove, removeAll, removeContainerListener, setComponentZOrder, setFocusTraversalKeys, setFocusTraversalPolicy, setFocusTraversalPolicyProvider, setFont, transferFocusDownCycle, validate, validateTree

Methods inherited from class java.awt.Component

action, add, addComponentListener, addFocusListener, addHierarchyBoundsListener, addHierarchyListener, addInputMethodListener, addKeyListener, addMouseListener, addMouseMotionListener, addMouseWheelListener, bounds, checkImage, checkImage, coalesceEvents, contains, contains, createImage, createImage, createVolatileImage, createVolatileImage, disable, disableEvents, dispatchEvent, enable, enable, enableEvents, enableInputMethods, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, getBaseline, getBaselineResizeBehavior, getBounds, getBounds, getColorModel, getComponentListeners, getComponentOrientation, getCursor, getDropTarget, getFocusListeners, getFocusTraversalKeysEnabled, getFont, getFontMetrics, getForeground, getGraphicsConfiguration, getHeight,

getHierarchyBoundsListeners, getHierarchyListeners, getIgnoreRepaint, getInputMethodListeners, getInputMethodRequests, getKeyListeners, getLocation, getLocation, getLocationOnScreen, getMouseListeners, getMouseMotionListeners, getMousePosition, getMouseWheelListeners, getName, getParent, getPropertyChangeListeners, getPropertyChangeListeners, getSize, getSize, getTreeLock, getWidth, getX, getY, gotFocus, handleEvent, hasFocus, imageUpdate, inside, isBackgroundSet, isCursorSet, isDisplayable, isDoubleBuffered, isEnabled, isFocusable, isFocusOwner, isFocusTraversable, isFontSet, isForegroundSet, isLightweight, isMaximumSizeSet, isMinimumSizeSet, isPreferredSizeSet, isValid, isVisible, keyDown, keyUp, list, list, list, location, lostFocus, mouseDown, mouseDrag, mouseEnter, mouseExit, mouseMove, mouseUp, move, nextFocus, paintAll, prepareImage, prepareImage, printAll, processComponentEvent, processFocusEvent, processHierarchyBoundsEvent, processHierarchyEvent, processInputMethodEvent, processKeyEvent, processMouseEvent, processMouseMotionEvent, processMouseWheelEvent, removeComponentListener, removeFocusListener, removeHierarchyBoundsListener, removeHierarchyListener, removeInputMethodListener, removeKeyListener, removeMouseListener, removeMouseMotionListener, removeMouseWheelListener, removePropertyChangeListener, removePropertyChangeListener, repaint, repaint, repaint, requestFocus, requestFocus, requestFocus, requestFocus, requestFocusInWindow, requestFocusInWindow, requestFocusInWindow, resize, resize, revalidate, setComponentOrientation, setDropTarget, setEnabled, setFocusable, setFocusTraversalKeysEnabled, setForeground, setIgnoreRepaint, setLocale, setMaximumSize, setMixingCutoutShape, setName, setPreferredSize, show, size, toString, transferFocus, transferFocusBackward, transferFocusUpCycle

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Methods inherited from interface java.awt.MenuContainer

getFont, postEvent

- ***Field Details***

- **calc**

  `private Calculate calc`

  Declares a Calculate variable to use methods.
- **numberDis**

  `private JTextField numberDis`

  Declares a JTextField variables to input numbers.
- **numberVel**

  `private JTextField numberVel`

  Declares a JTextField variables to input numbers.
- **numberIniVel**

  `private JTextField numberIniVel`

  Declares a JTextField variables to input numbers.
- **numberTime**

  `private JTextField numberTime`

  Declares a JTextField variables to input numbers.
- **numberAcc**

  `private JTextField numberAcc`

  Declares a JTextField variables to input numbers.
- **answer**

  `private JTextField answer`

  Declares a JTextField variables to input numbers.
- **missing**

  `private double missing`

  Declares a missing variable to be set to the missing number in the equation.

-
- ***Constructor Details***
  - **PhysicsCalc**

    `public PhysicsCalc()`

    This contructor creates a Content Pane with a Flow Layout to put the different JLabels, JTextFields, and JButtons into. Action listener allows the calculator to use methods from the Calulate class to solve the missing variable then display it.
-
- ***Method Details***

- **actionPerformed**

  `public void actionPerformed(ActionEvent e)`

  This method uses Action Listener for the buttons and calls on the equation methods to find the missing variable. and then displays the missing variable to the user like a calculator.

  Specified by:

  `actionPerformed` **in interface** `ActionListener`
- **main**

  `public static void main(String[] args)`

  Main method to create a new PhysicsCalc object, Runs a window for the calculator and sets parameters and colors. To use the calculator, simply input your variables and press the corresponding button for the specific equation.
-

Class Calculate

java.lang.Object
  Calculate

```
public class Calculate
extends Object
```

Calculate for Physics!

The Calculate program implements an application that finds the variable in a given case for a standard kinematics question.

User can choose between the three formulas to solve a kinematics question that is asking for a specific variable through the use of algebra. I set the variables as object Double's instead of primitive doubles because I could set the object variables as null for the missing variable a student needed to solve.

Since:

2020-12-03

- ***Field Summary***

Modifier and Type	Field	Description
private Double	acceleration	Acceleration variable can have magnitude and direction.
private Double	displacement	Displacement variable can have magnitude and direction.
private Double	iniVelocity	Initial velocity variable can have magnitude and direction.
private Double	time	Time variable can only have magnitude, no negatives.
private Double	velocity	Velocity variable can have magnitude and direction.

- ### *Constructor Summary*

Constructor	Description
Calculate()	Basic constructor, sets all instance variables to null when ran.

-
- ### *Method Summary*

All Methods Static Methods Instance Methods Concrete Methods

Mo	Method	Description
do	**displacementWithoutVel**()	Method for the physics displacement equation: Δx = v0t+1/2at^2 No parameters needed and finds the missing variable when called upon if it is null for each case.
st	**main**(**String**[] args)	Main method tests the methods to make sure they are working correctly when called upon with different objects.
vo	**setAcc**(**Double** accele	Sets the acceleration when ran.

	ration)	
vo	**setDis**(**Double** displacement)	Sets the displacement when ran.
vo	**setIniVel**(**Double** iniVelocity)	Sets the initial velocity when ran.
vo	**setTime**(**Double** time)	Sets the time when ran.
vo	**setVel**(**Double** velocity)	Sets the velocity when ran.
do	**velWithoutDis**()	Method for the physics velocity equation: v = v0+at No parameters needed and finds the missing variable when called upon if it is null for each case.

do	**velWithoutTime**()	Method for the physics velocity equation: v^2 = v0^2+2a(Δx) No parameters needed and finds the missing variable when called upon if it is null for each case.

- **Methods inherited from class java.lang.Object**
  `clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`
- ***Field Details***
  - **displacement**
    `private Double displacement`
    Displacement variable can have magnitude and direction.
  - **velocity**
    `private Double velocity`
    Velocity variable can have magnitude and direction.
  - **iniVelocity**
    `private Double iniVelocity`
    Initial velocity variable can have magnitude and direction.
  - **acceleration**
    `private Double acceleration`
    Acceleration variable can have magnitude and direction.
  - **time**
    `private Double time`
    Time variable can only have magnitude, no negatives.
- ***Constructor Details***
  - **Calculate**
    `public Calculate()`
    Basic constructor, sets all instance variables to null when ran.
- ***Method Details***

- **setDis**

  `public void setDis(Double displacement)`

  Sets the displacement when ran.

  Parameters:

  `displacment` -
- **setVel**

  `public void setVel(Double velocity)`

  Sets the velocity when ran.

  Parameters:

  `velocity` -
- **setIniVel**

  `public void setIniVel(Double iniVelocity)`

  Sets the initial velocity when ran.

  Parameters:

  `iniVelocity` -
- **setAcc**

  `public void setAcc(Double acceleration)`

  Sets the acceleration when ran.

  Parameters:

  `acceleration` -
- **setTime**

  `public void setTime(Double time)`

  Sets the time when ran.

  Parameters:

  `time` -
- **velWithoutTime**

  `public double velWithoutTime()`

  Method for the physics velocity equation: v^2 = v0^2+2a(Δx) No parameters needed and finds the missing variable when called upon if it is null for each case.

  Returns:

  value of missing variable
- **velWithoutDis**

  `public double velWithoutDis()`

  Method for the physics velocity equation: v = v0+at No parameters needed and finds the missing variable when called upon if it is null for each case.

  Returns:

  value of missing variable

- **displacementWithoutVel**

  `public double displacementWithoutVel()`

  Method for the physics displacement equation: Δx = v0t+1/2at^2 No parameters needed and finds the missing variable when called upon if it is null for each case.

  Returns:

  value of missing variable

- **main**

  `public static void main(String[] args)`

  Main method tests the methods to make sure they are working correctly when called upon with different objects. and different questions.

-