Adding radio buttons

The Swing `JRadioButton` class creates a radio button component that can be added to a graphical interface. This can be used to allow the user to select an item from a group of radio buttons.

The `JRadioButton` object is created with the `new` keyword and its constructor takes a `String` argument specifying text to be displayed alongside that radio button. It can also take a second `true` argument to make a radio button be selected by default.

A `ButtonGroup` object logically groups a number of radio buttons so that only one button in that group can be selected at any time. Each radio button is added to the `ButtonGroup` object by specifying its name as the argument to the group’s `add()` method.

1. Edit a copy of `Window.java` from page 135, changing the class name in the declaration, the constructor, and the instance statement from “Window” to “Radios”

2. Before the `Radios()` constructor, create three `JRadioButton` objects – with one selected by default
   ```java
   JRadioButton rad1 = new JRadioButton( "Red", true ) ;
   JRadioButton rad2 = new JRadioButton( "Rosé" ) ;
   JRadioButton rad3 = new JRadioButton( "White" ) ;
   ```

3. Next create a `ButtonGroup` object with which to group the radio buttons
   ```java
   ButtonGroup wines = new ButtonGroup() ;
   ```

4. In the `Radios()` constructor method, insert statements to add each `JRadioButton` component to the `JButtonGroup` objects.
   ```java
   wines.add( rad1 ) ;
   wines.add( rad2 ) ;
   wines.add( rad3 ) ;
   ```

5. Insert statements to add the `JRadioButton` components to the `JPanel` container
   ```java
   pnl.add( rad1 ) ;
   pnl.add( rad2 ) ;
   pnl.add( rad3 ) ;
   ```

6. Save the program as `Radios.java` then compile and run the program, selecting any one radio button after the default