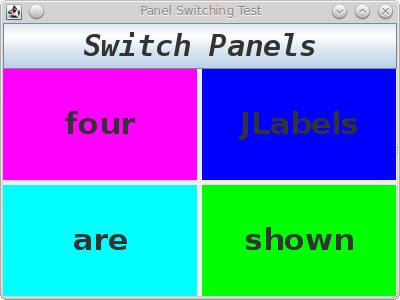
## Lab 7

Some of you might want to switch window components in your your team projects, in particular JPanels. For example, you might want a start screen with buttons, which you want to replace with a graphics screen. This lab will show you one way to do this following the Model-View-Controller design pattern.

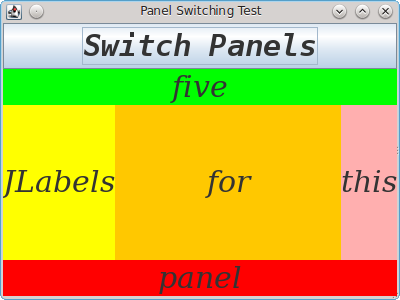
The name of your Eclipse project should be abc123-lab7, where you replace abc123 with your abc123 id. The name of the file that contains the main method should remain PanelSwitcher.java. The remaining Java files should also keep their original names. To submit the project, export the project and upload the zip file to Blackboard.

### Task

Your task to modify the code in [lab7.zip](http://cs.utsa.edu/~cs3443/laboratories/lab7.zip) so that the Switch Panels button will switch back and forth between



and



PanelSwitcherView.java creates two JPanels and a JButton and contains a method displayPanel to display one of the two panels. However, the JButton doesn't do anything because it is not registered with any event handler.

PanelSwitcherModel.java stores a 1 or a 2, which is intended to indicate which panel should be displayed.

PanelSwitcher.java contains a main method that creates PanelSwitcherView and PanelSwitcherModel objects. However, it cannot create a PanelSwitcherController object because you haven't written that code yet.

You need to write PanelSwitcherController.java and any additional code in the other classes so that a PanelSwitcherController object will handle events from the JButton by updating the model and the view. [Hint: This class should implement ActionListener.] If you find the carefully calibrated color coordination too disturbing, you may change the background colors of the JLabels.

### Comments

Add javadoc comments as needed for any additions or changes.

### UML Class Diagram

Create a UML class diagram using Violet.

### Rubric

* An incorrect submission will possibly get zero points. A project that does not compile will receive at most 50 points total.
* (80 pts.) If the JButton can be used to switch back and forth between the two panels. This will be only 40 points if the JButton can switch to one panel, but can't switch back.
* (10 pts.) There are javadoc comments: at least one for each class and comments for each method, constructor and constant.
* (10 pts.) A UML class diagram is included that has all the classes (including the PanelSwitcher class) and all the constructors, methods, variables and constants of each class.