



Layout Management

By,

Hitha Paulson

Assistant Professor, Dept. of Computer Science

LF College, Guruvayoor

Layout



- The way in which the controls of a container are arranged
- All the containers have default Layout
 - Eg: Applet – `FlowLayout.CENTER`
 - Eg: Frame – `BorderLayout.CENTER`
- **Layout Manager:** An instance of any class that implements the `LayoutManager` interface
- Layout can set by using `setLayout()` method
 - Eg: `setLayout(layoutobj)`
 - Layout can be “null”; No default Layout manager



- **Layout Manager will invoke**
 - Container is resized
 - Add Component
- **Layout Size**
 - `minimumLayoutSize()`
 - `preferredLayoutSize()`
- **Component Size**
 - `getPreferredSize()`
 - `getMinimumSize()`

FlowLayout

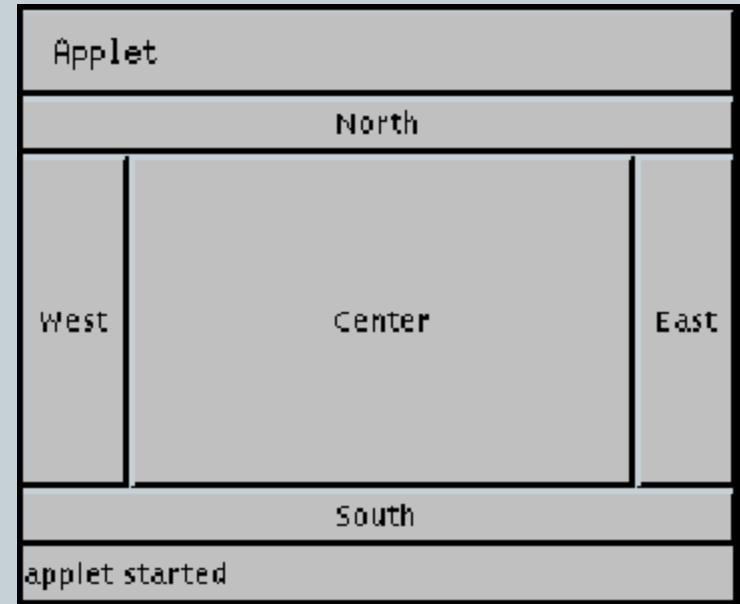


- Components are arranged as floating objects
- Small space will left between each component
- Class: FlowLayout
 - FlowLayout()
 - FlowLayout(int how)
 - ✦ how
 - FlowLayout.LEFT
 - FlowLayout.CENTER
 - FlowLayout.RIGHT
 - FlowLayout(int how, int hSpze, int vSpze)

BorderLayout



- The container is divided into five regions
 - BorderLayout.CENTER
 - BorderLayout.EAST
 - BorderLayout.WEST
 - BorderLayout.SOUTH
 - BorderLayout.NORTH
- Class: BorderLayout
 - BorderLayout()
 - BorderLayout(hSpz, vSpz)
- Adding Component
 - add(component, region)



Using Insets



- Insets: Space between the container that holds component and the window that contains it
- Override the `getInsets()` method to specify new dimension
- `Insets(int top, int left, int bottom, int right)`
- Eg:

```
public Insets getInsets()  
{  
    return new Insets(10,10,10,10);  
}
```

GridLayout



- Divides the container into two dimensional grid
- Class: GridLayout
 - GridLayout() – Single column
 - GridLayout(rows, cols)
 - GridLayout(rows, cols, hSpz, vSpz)
 - rows, cols can be zero, means it can have unlimited length

CardLayout



- More than one Layouts are managing at the same time
- Panel
 - It is container
 - not visible
 - Can set Layout
- Class: CardLayout
 - CardLayout()
 - CardLayout(hSpz, vSpz)
- Steps
 - Assign layout to top-level container in the program
 - Create another container to hold the first group of controls
 - Similarly create more containers to hold the subsequent group of controls
- Recalling Cards
 - Void first(Container deck)
 - Void last(Container deck)
 - Void next(Container deck)
 - Void previous(Container deck)
 - Void show(Container deck, String cardName)