CODENAME ONE

PSD to App

- Codename One allows Java developers to build **true native apps** for **all mobile devices**
- Its **free** & Open Source!

Demos

MWC Events
Sunday February 26th
Monday February 27th
Tuesday February 28th
Wednesday February 29th
Thursday March 1st
June 2012 June 19th
CARRIER
CODENAME ONE LIVE!
Builds
Settings
LIVE!
Shai
Almog
Company
shai@codenameone.com
Save
Kitchen Sink
Effects
Layouts
Dialogs
Contacts
Mail
Themes
Web
Components
Video
BlackBerry
ePassword
Self Destruct
after 5 failed login attem...
Master Password
all Passwords
ore from Cloud
up to Cloud
denameone.com
Settings
About
at&t
postcode here

API Designer Simulator Plugin

Build Server Service Cloud

Features

- JavaSE 5 Subset with Java 8 Syntax support (Lambdas, Try-with etc…)
- Write Once Deploy Everywhere (iOS, Android, WinPhone, BlackBerry, J2ME, HTML5, Windows, Mac, Linux, …)
- Drag & Drop Development
- Easy resources for multi-DPI
- Too many features to list…

Many other APIs

Social

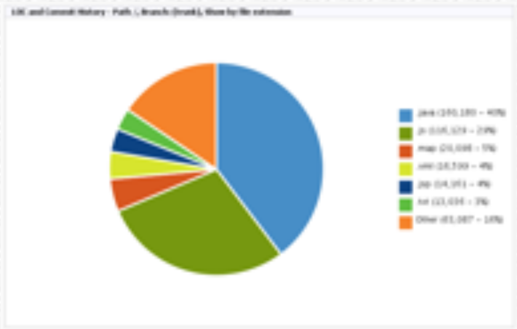
AdMob by Google

vserv.mobi

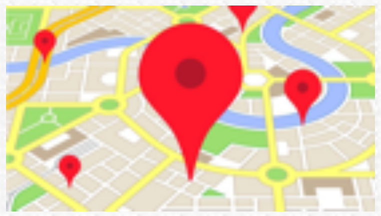
Ads

START/APP

FLURRY



Charts



Maps

Analytics

Push

Websockets

Parse
The Cloud Application Platform

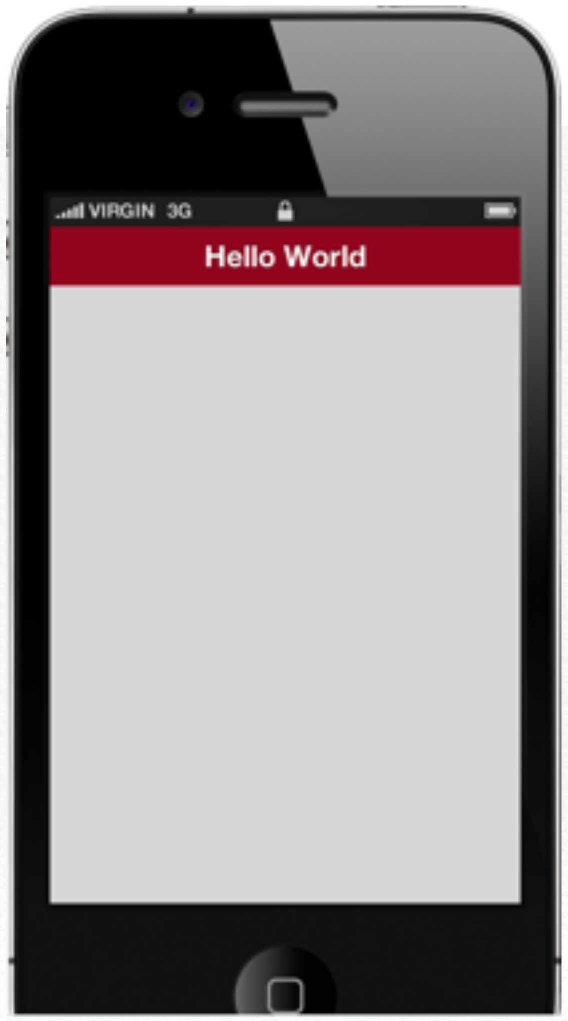
PubNub

SQLite

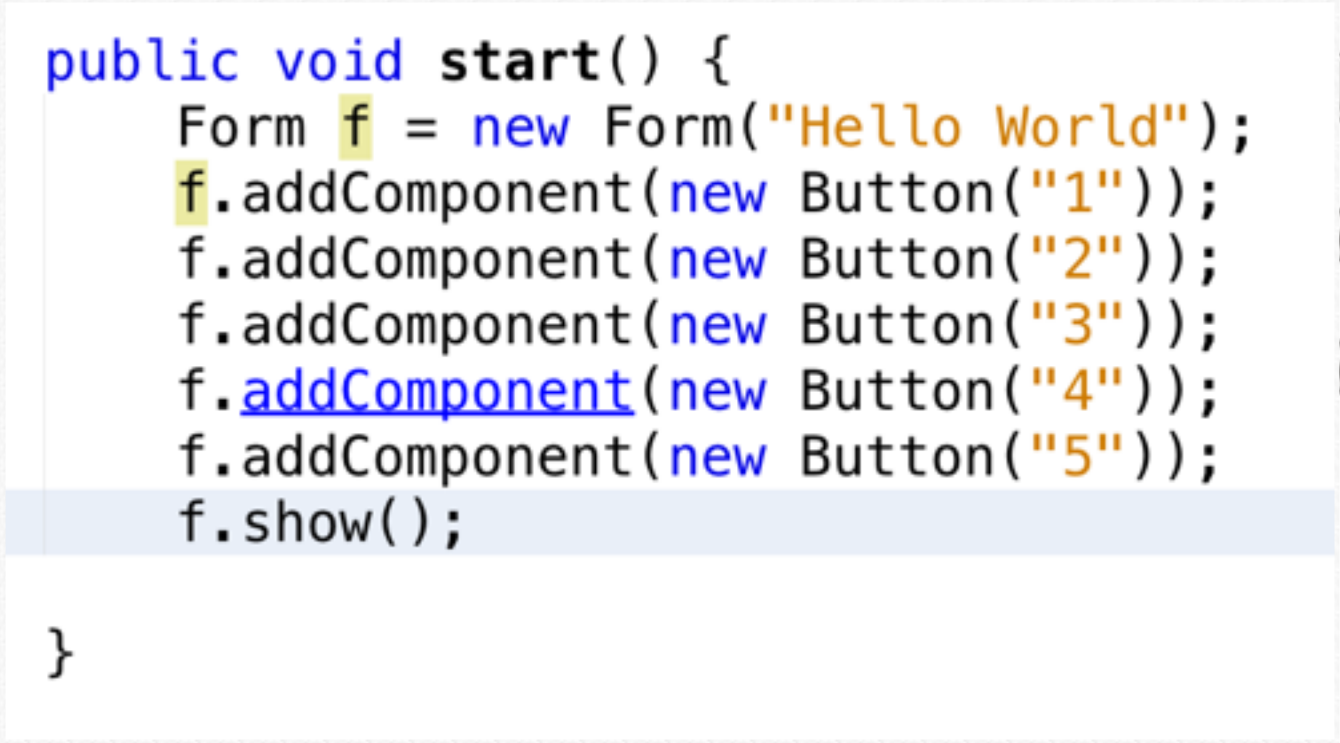
CODENAME ONE

Simplest App

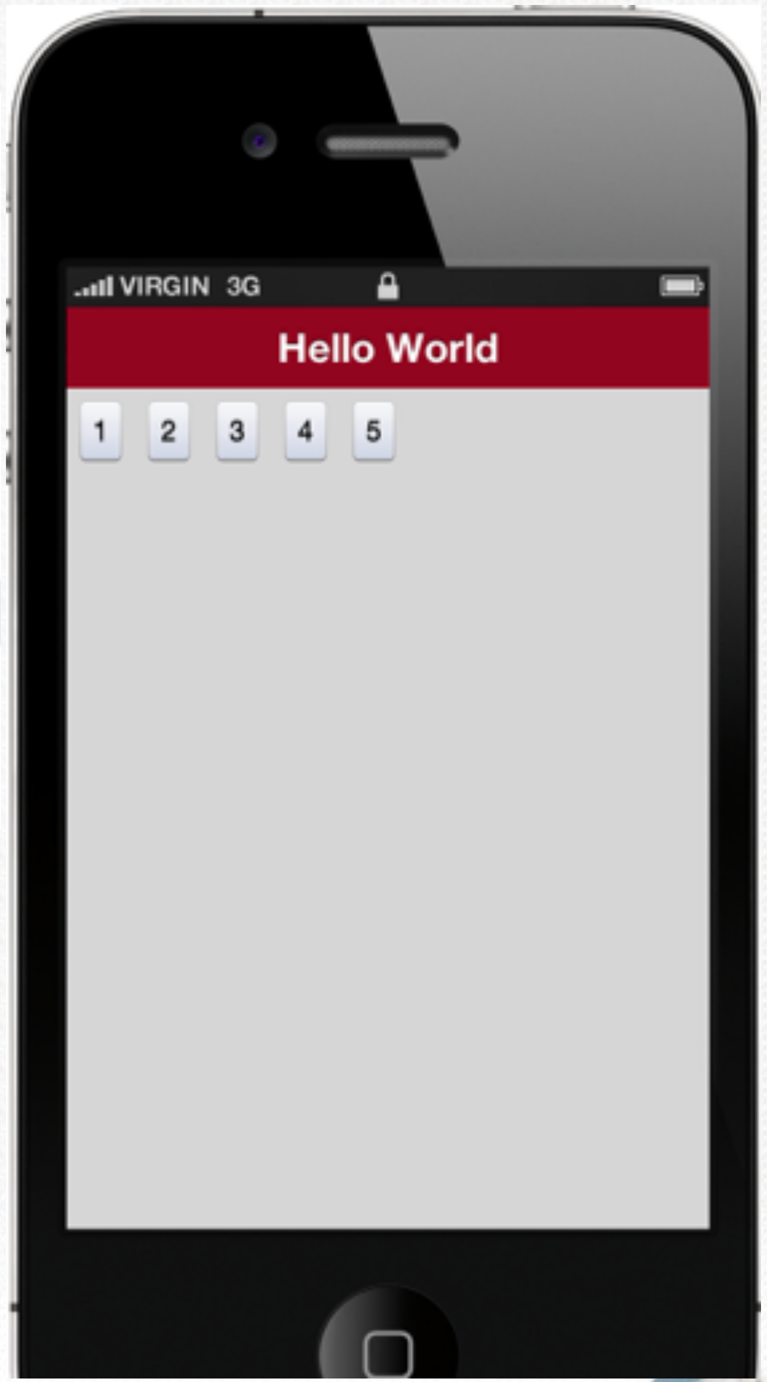
```
public void start() {
    new Form("Hello World").show();
}
```



Add Some Buttons

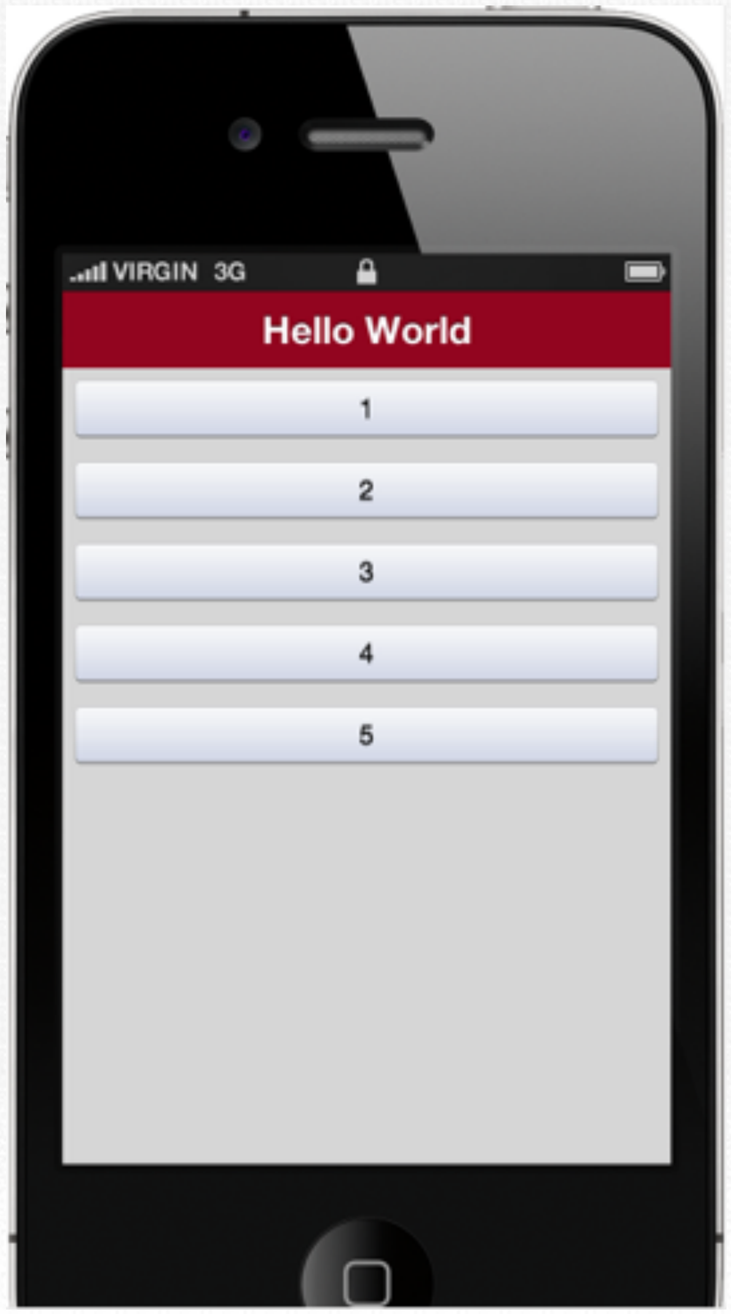


```
public void start() {
    Form f = new Form("Hello World");
    f.addComponent(new Button("1"));
    f.addComponent(new Button("2"));
    f.addComponent(new Button("3"));
    f.addComponent(new Button("4"));
    f.addComponent(new Button("5"));
    f.show();

}
```

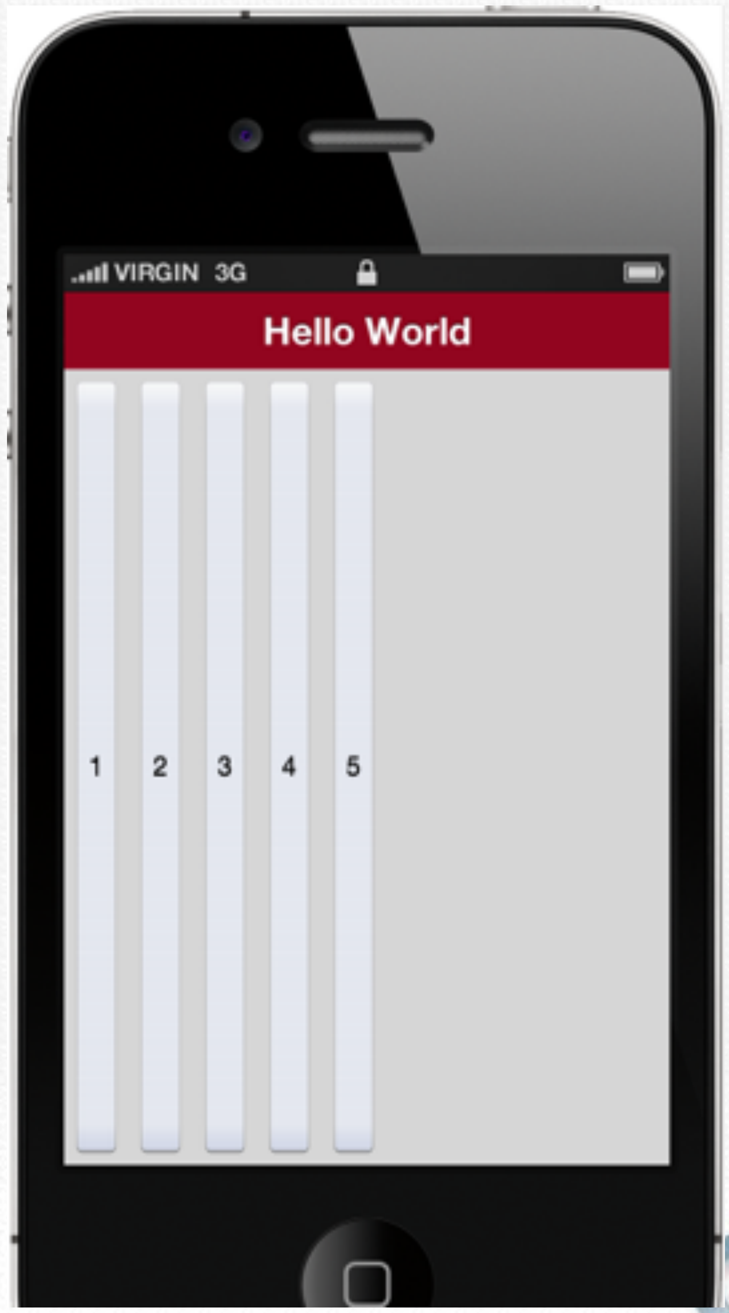


Change the Layout

```
f.setLayout(new BoxLayout(BoxLayout.Y_AXIS));
```



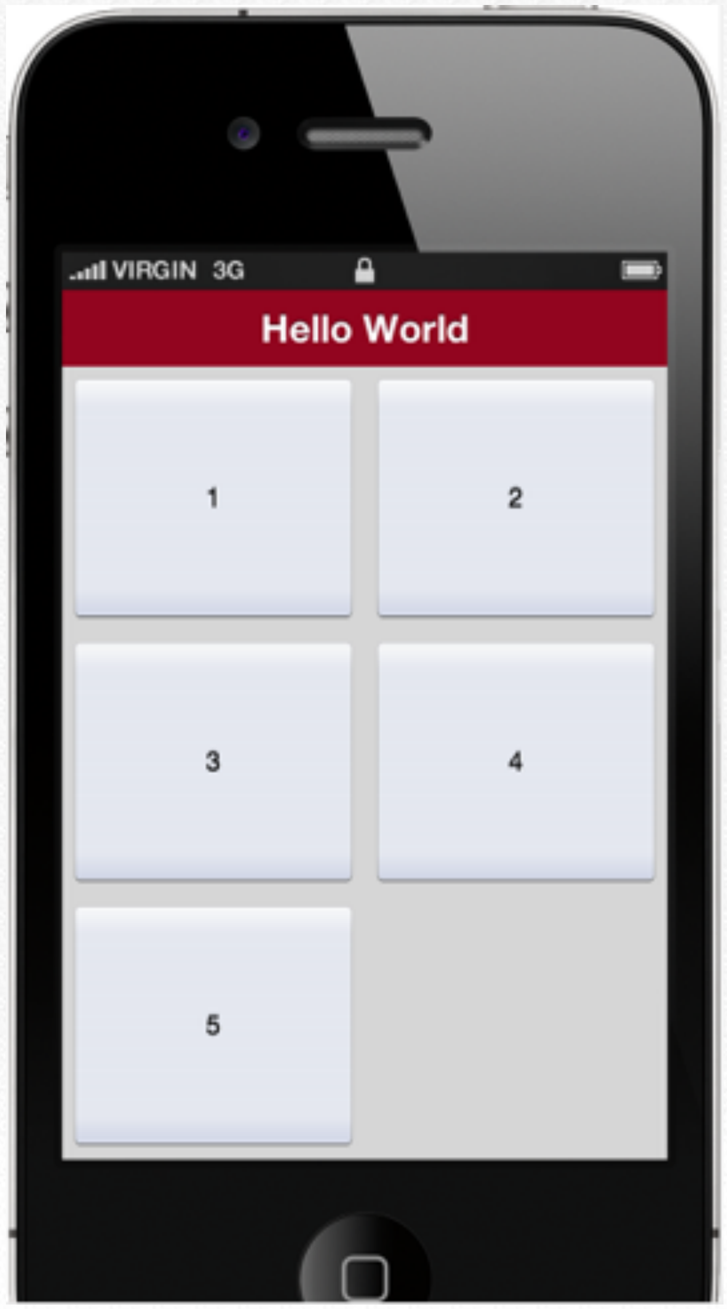
```
f.setLayout(new BoxLayout(BoxLayout.X_AXIS));
```



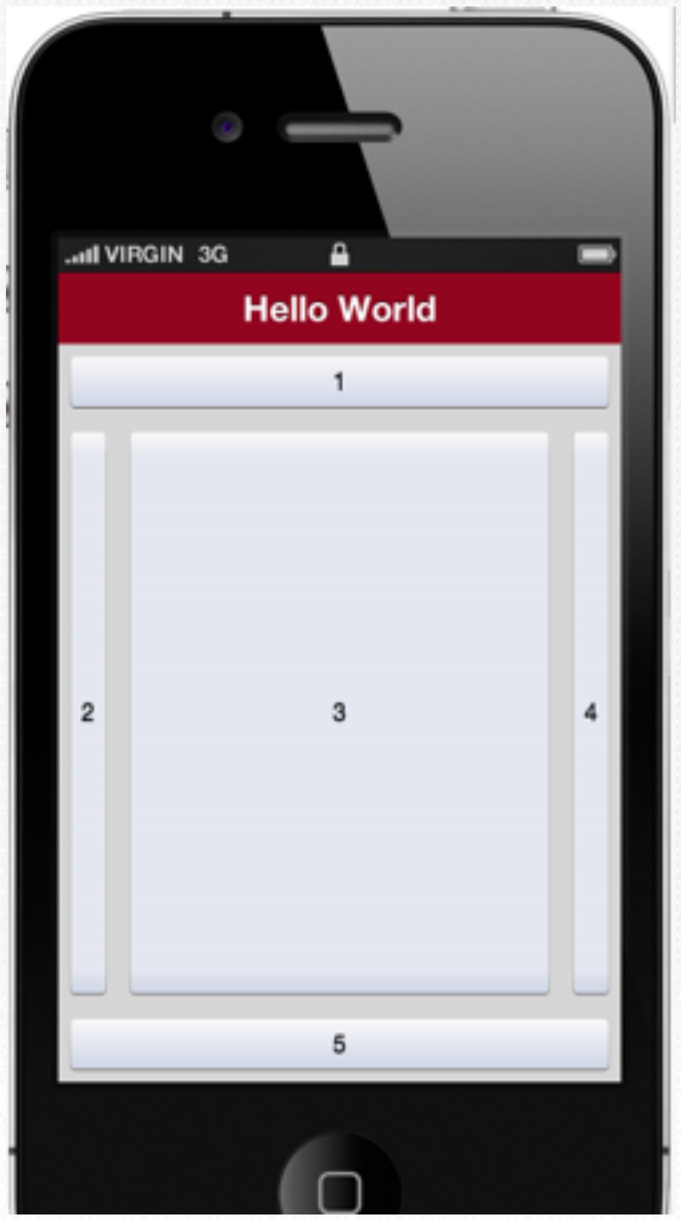
CODENAME ONE

More Layouts

```java
f.setLayout(new GridLayout(3,2));
```



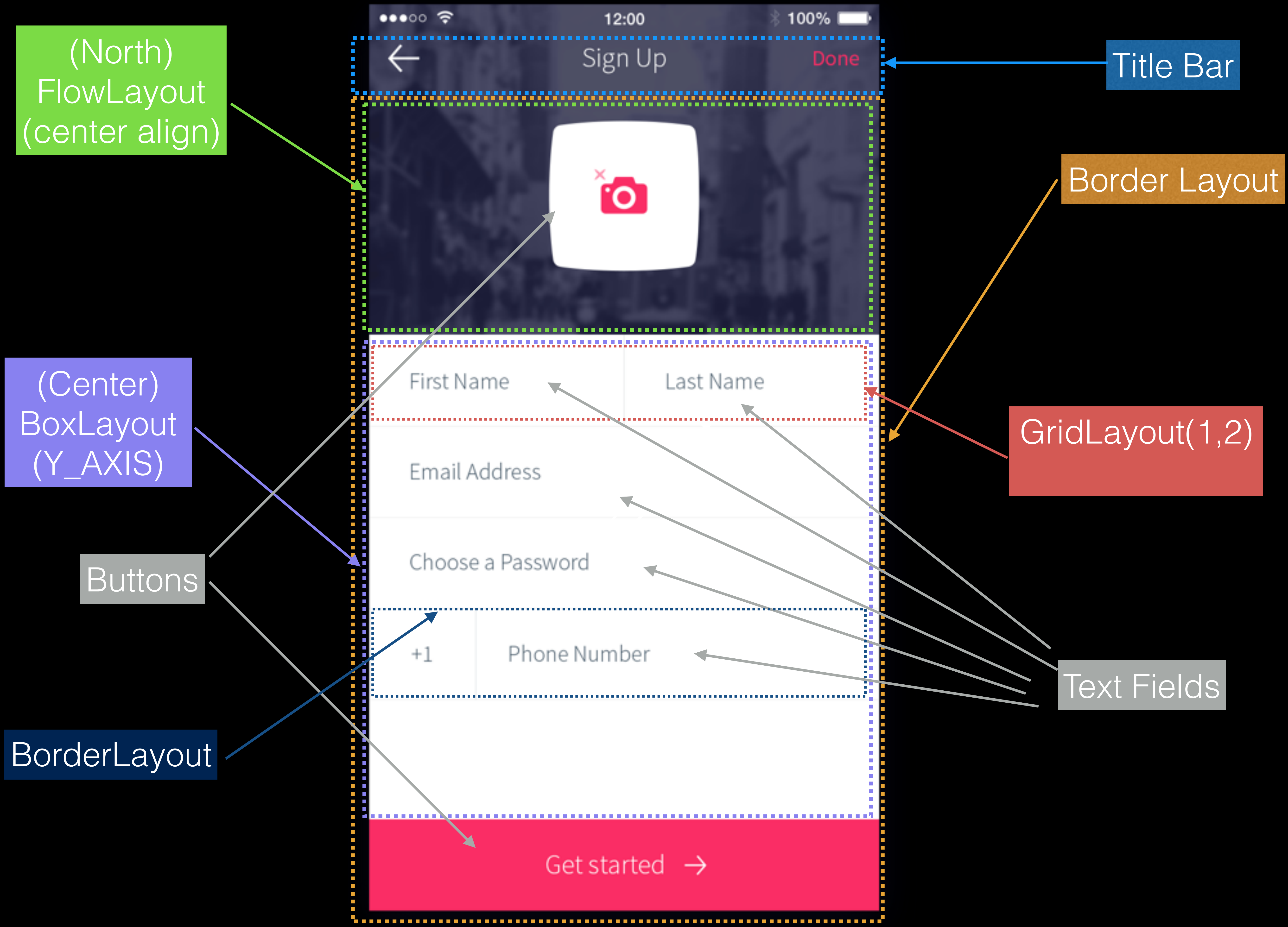
```java
f.setLayout(new BorderLayout());
f.addComponent(BorderLayout.NORTH, new Button("1"));
f.addComponent(BorderLayout.WEST, new Button("2"));
f.addComponent(BorderLayout.CENTER, new Button("3"));
f.addComponent(BorderLayout.EAST, new Button("4"));
f.addComponent(BorderLayout.SOUTH, new Button("5"));
```



Complex Layouts

Use Container class to create a nested hierarchy of components.

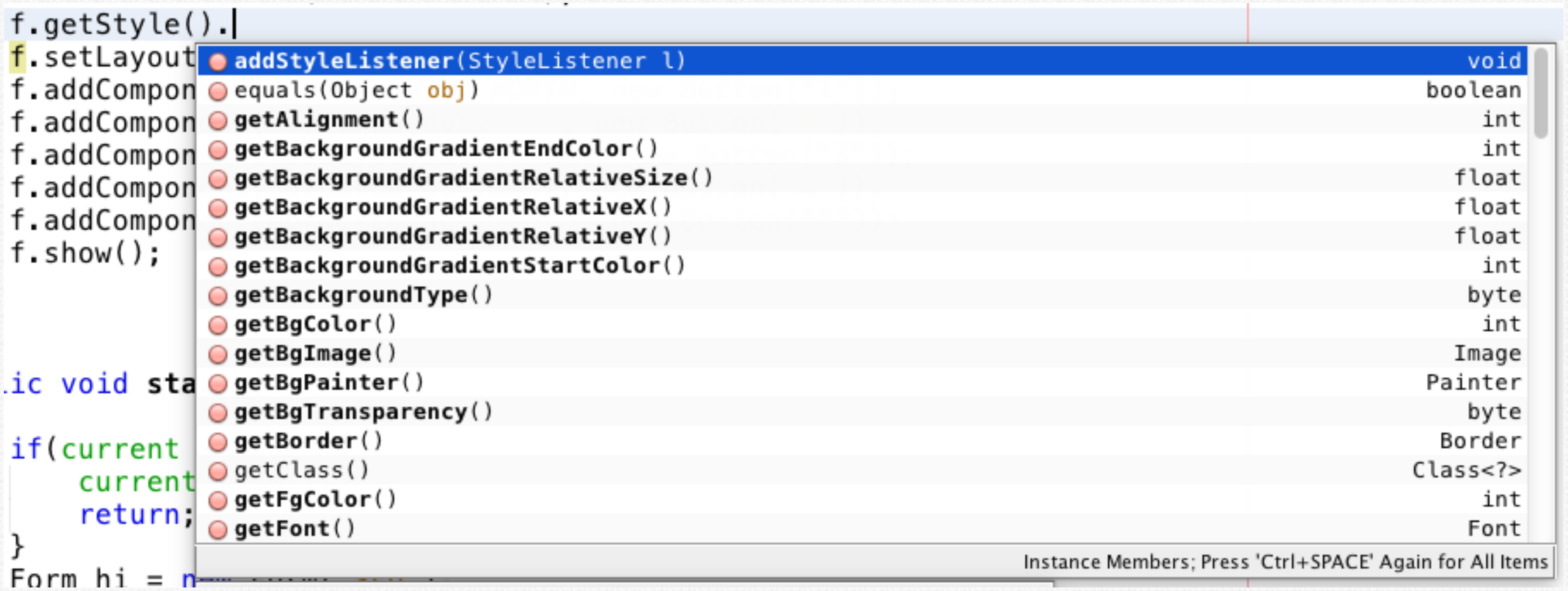
Each Container can have its own layout.


(North) FlowLayout (center align)
12:00
100%
Sign Up
Done
Title Bar
Border Layout
(Center) BoxLayout (Y_AXIS)
First Name
Last Name
GridLayout(1,2)
Email Address
Choose a Password
Buttons
+1
Phone Number
Text Fields
BorderLayout
Get started

Styles

- Use Layouts for positioning
- Use Styles and UIIDs for “styling”.
  - Colors, fonts, padding, margin, background images, borders.

Style via Java API



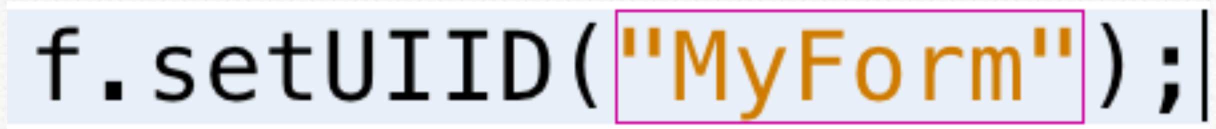
```
f.getStyle().
f.setLayout
f.addCompon
f.addCompon
f.addCompon
f.addCompon
f.addCompon
f.show();


ic void sta

if(current
    current
    return;
}
Form hi = n
```

```
addStyleListener(StyleListener l)                    void
equals(Object obj)                                   boolean
getAlignment()                                       int
getBackgroundGradientEndColor()                      int
getBackgroundGradientRelativeSize()                  float
getBackgroundGradientRelativeX()                     float
getBackgroundGradientRelativeY()                     float
getBackgroundGradientStartColor()                    int
getBackgroundType()                                  byte
getBgColor()                                         int
getBgImage()                                         Image
getBgPainter()                                       Painter
getBgTransparency()                                  byte
getBorder()                                          Border
getClass()                                           Class<?>
getFgColor()                                         int
getFont()                                            Font
```

Instance Members; Press 'Ctrl+SPACE' Again for All Items

Style via UIID

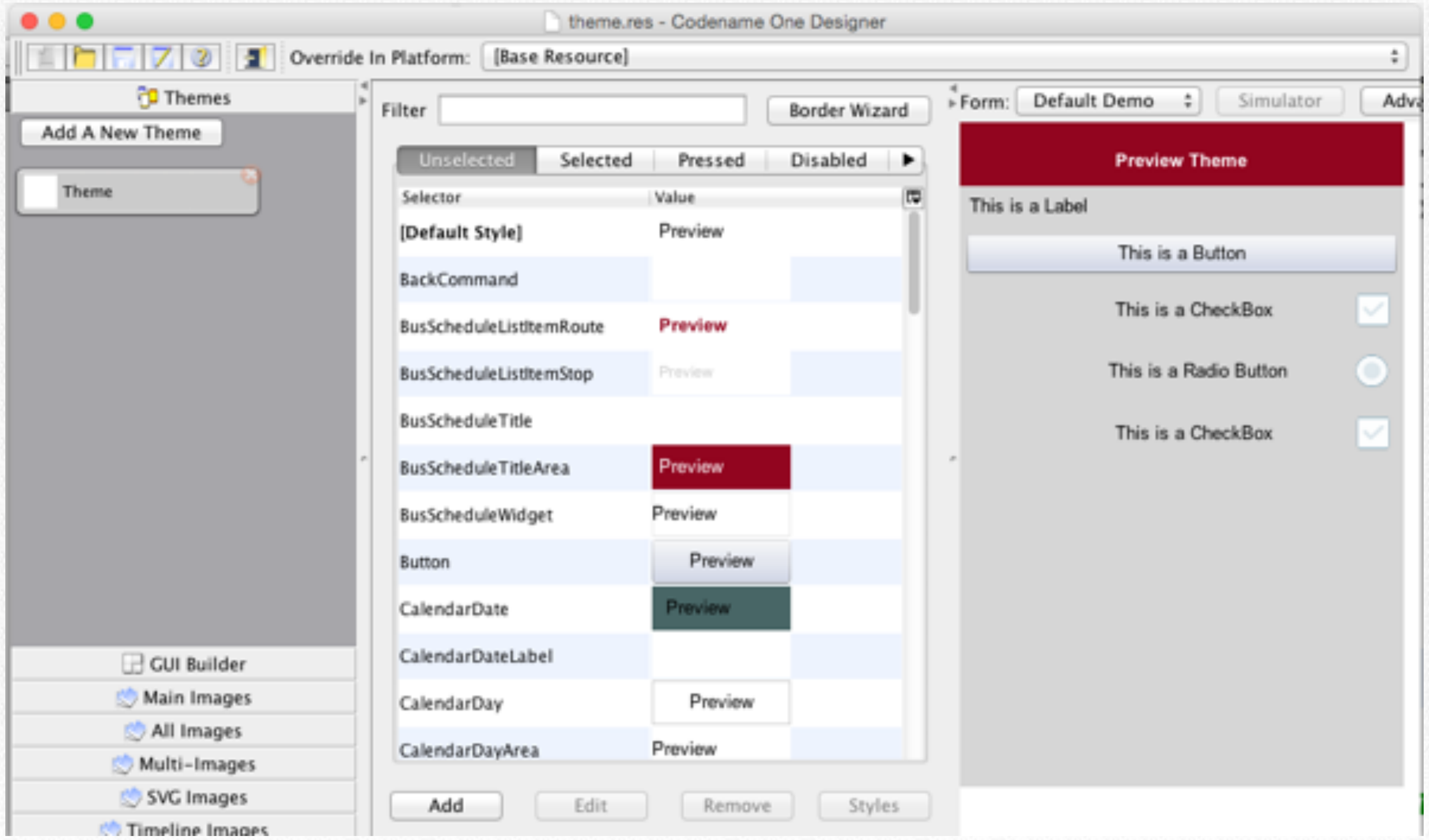
- UIIDs like CSS classes. Register a component with a UIID, and it will retrieve styles from current theme in resource file.



```
f.setUIID("MyForm");
```

Editing Theme

- Codename One plugin comes with a resource editor to edit theme styles:

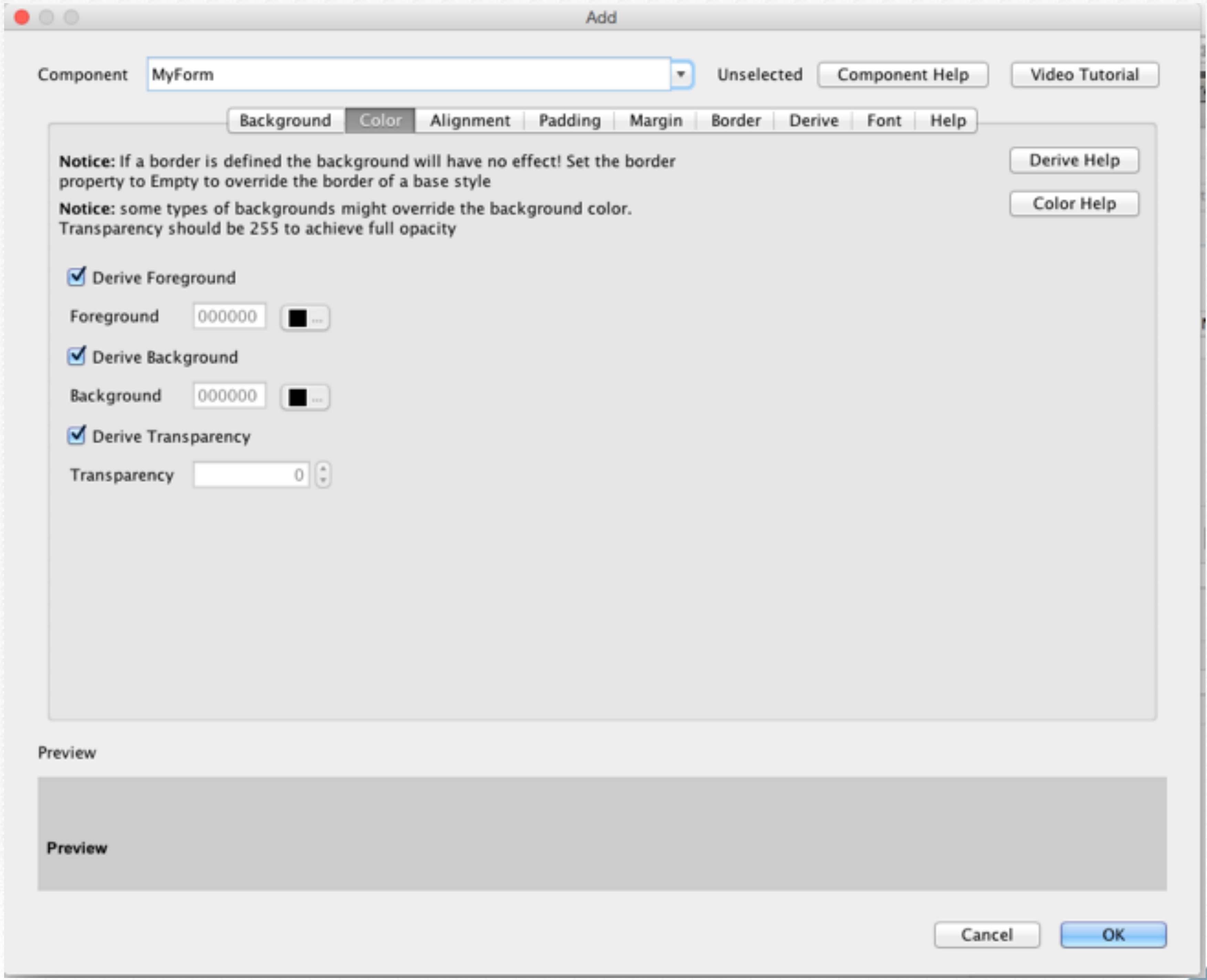


Override In Platform: [Base Resource]

Form: Default Demo

Selector	Value
[Default Style]	Preview
BackCommand	
BusScheduleListItemRoute	Preview
BusScheduleListItemStop	Preview
BusScheduleTitle	
BusScheduleTitleArea	Preview
BusScheduleWidget	Preview
Button	Preview
CalendarDate	Preview
CalendarDateLabel	
CalendarDay	Preview
CalendarDayArea	Preview

Editing UIID Style



Component: MyForm

Unselected
Component Help
Video Tutorial
Background | Color | Alignment | Padding | Margin | Border | Derive | Font | Help
Derive Help
Color Help

Notice: If a border is defined the background will have no effect! Set the border property to Empty to override the border of a base style

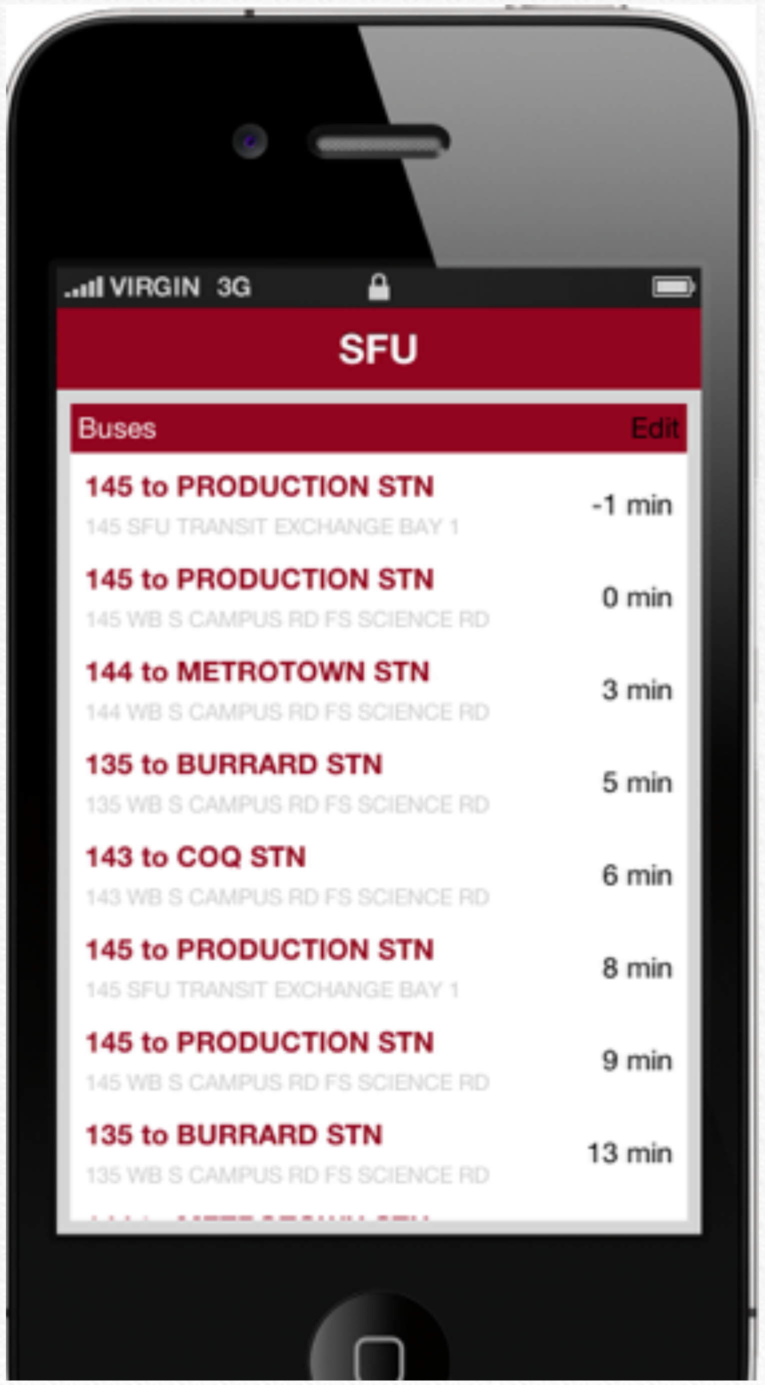
Notice: some types of backgrounds might override the background color. Transparency should be 255 to achieve full opacity

- [✓] Derive Foreground
- Foreground: 000000
- [✓] Derive Background
- Background: 000000
- [✓] Derive Transparency
- Transparency: 0

Exercise 1

- Create an app that displays a list of the next buses to arrive at SFU.
- Hint:

```
client.findNextBuses(null, null);
```



145 to PRODUCTION STN — 145 SFU TRANSIT EXCHANGE BAY 1 — -1 min
145 to PRODUCTION STN — 145 WB S CAMPUS RD FS SCIENCE RD — 0 min
144 to METROTOWN STN — 144 WB S CAMPUS RD FS SCIENCE RD — 3 min
135 to BURRARD STN — 135 WB S CAMPUS RD FS SCIENCE RD — 5 min
143 to COQ STN — 143 WB S CAMPUS RD FS SCIENCE RD — 6 min
145 to PRODUCTION STN — 145 SFU TRANSIT EXCHANGE BAY 1 — 8 min
145 to PRODUCTION STN — 145 WB S CAMPUS RD FS SCIENCE RD — 9 min
135 to BURRARD STN — 135 WB S CAMPUS RD FS SCIENCE RD — 13 min

The EDT

- Codename One is multithreaded.
- All interaction with the UI must occur on single thread: the EDT (Event Dispatch Thread).
- Long running operations should be run off the EDT — **DON'T BLOCK THE EDT.**

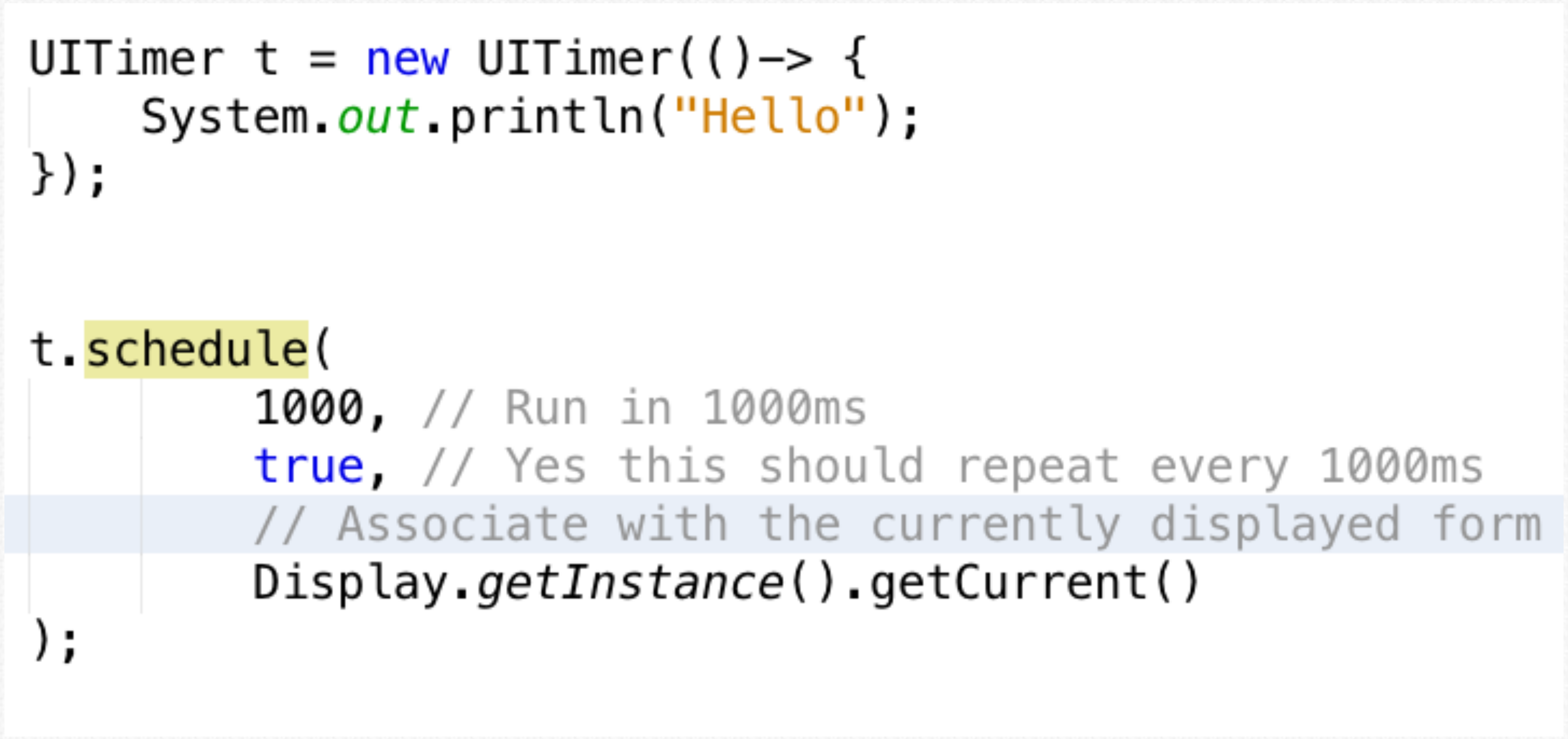
Useful EDT Tools

- Display.isEdt() : Checks if you are on the EDT.
- Display.callSerially() : Runs code on the EDT asynchronously
- Display.invokeAndBlock() : Runs code off the EDT without blocking the EDT.
- Display.callSeriallyAndWait()

Timers

- java.util.Timer runs code periodically in background thread.
- com.codename1.ui.util.UITimer runs code on periodically the EDT.
    - Associated with a form… only runs while form is “active”.

UITimer example



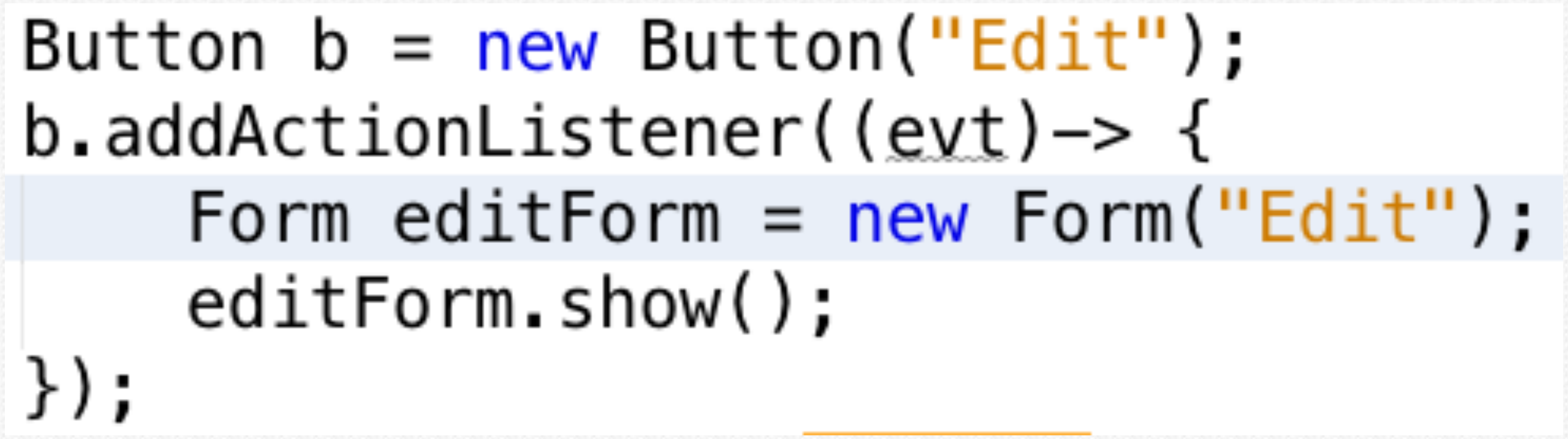
```
UITimer t = new UITimer(()-> {
    System.out.println("Hello");
});


t.schedule(
        1000, // Run in 1000ms
        true, // Yes this should repeat every 1000ms
        // Associate with the currently displayed form
        Display.getInstance().getCurrent()
);
```

Exercise 2

- Add UITimer that refreshes the list of buses whenever the TranslinkRESTClient is modified - or every 30 seconds whichever is sooner.

User Interaction

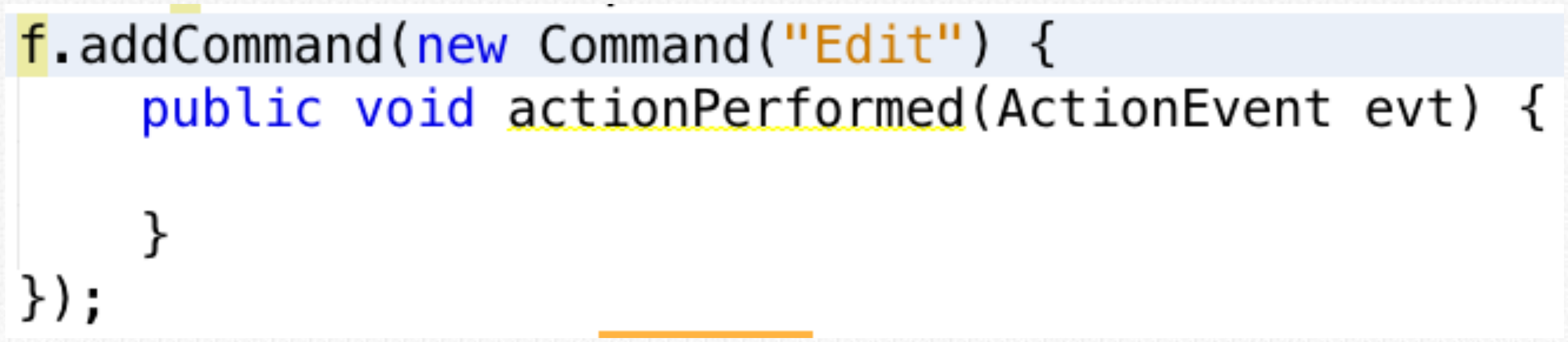
- Many components broadcast “ActionEvent”s when they are clicked. E.g. buttons, lists, checkboxes.



```java
Button b = new Button("Edit");
b.addActionListener((evt)-> {
    Form editForm = new Form("Edit");
    editForm.show();
});
```

Commands

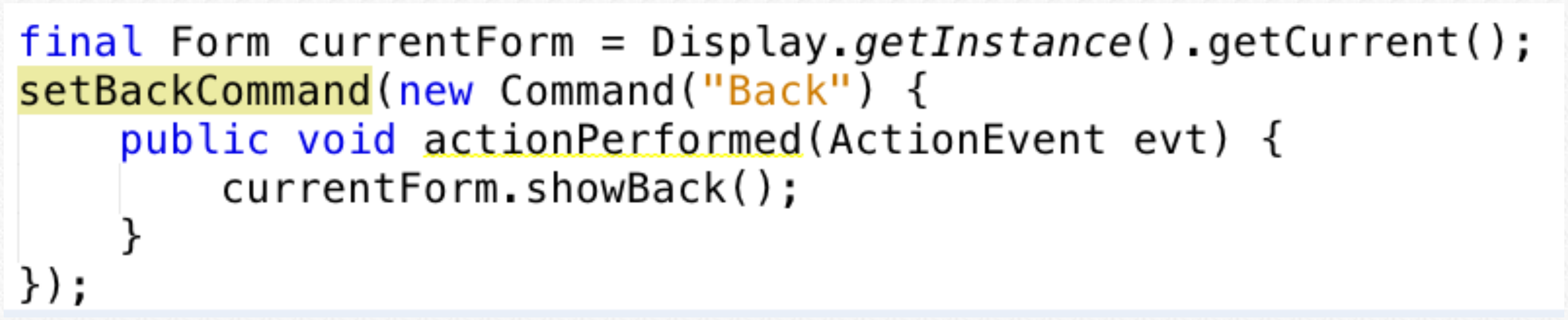
- Forms allow you to register "Command"s.
- Manifested as either buttons, menu options, or soft key handlers depending on your platform's settings.



```java
f.addCommand(new Command("Edit") {
    public void actionPerformed(ActionEvent evt) {

    }
});
```

Back Command

- A special command to handle the "back" action.
- Automatically linked up to "back" button on Android.



```
final Form currentForm = Display.getInstance().getCurrent();
setBackCommand(new Command("Back") {
    public void actionPerformed(ActionEvent evt) {
        currentForm.showBack();
    }
});
```

Exercise 3

- Create a “Preferences” form to specify which stops to include in the bus schedule.

Exercise 4

- Style your app to look more “SFU”-ey