



Móviles

Tema: Pestañas con las Clases `TabHost`, `TabSpec` y `TabWidget`

CONCEPTOS

Tabs (pestañas)

Las pestañas permiten crear una interfaz de usuario con pestañas que se seleccionan al digitar sobre las pestañas que se muestran en la parte superior:

<code>TabHost</code>	Contenedor para una vista de ventana con pestañas.
<code>TabHost.TabSpec</code>	Una pestaña tiene un índice de tabulación, el contenido y una etiqueta que se utiliza para realizar un seguimiento de la misma.
<code>TabWidget</code>	Muestra una lista de etiquetas de las pestañas que representan a cada página en la colección de la pestaña del padre.

Cuando las pestañas se utilizan con `Fragment`s, se necesita utilizar las clases `FragmentTabHost` y `TabWidget`. `FragmentTabHost` debe ser el nodo raíz que contendrá, tanto el `TabWidget` para la mostrar las pestañas, como un `FrameLayout` que muestra el contenido. Ver la figura 1.

La estructura básica de las pestañas con fragmentos es la siguiente:

```
<android.support.v4.app.FragmentTabHost
    android:id="@android:id/tabhost" .../>
<LinearLayout ...>
    <TabWidget
        android:id="@android:id/tabs" .../>
    <FrameLayout
        android:id="@android:id/tabcontent" ...>
</LinearLayout>
</TabHost>
```



Figura 1. Dos pestañas.

DESARROLLO

EJEMPLO 1.

Paso 1. En este primer ejemplo no utilizan fragmentos, como se muestra en el siguiente código XML.



Crear un nuevo proyecto **Pestañas1**. En la carpeta **res/layout**, abrir el archivo predeterminado **activity_main.xml** para modificarlo con el siguiente código:

```
<LinearLayout ... >
    <TabHost android:id="@android:id/tabhost">
        <LinearLayout ... >
            <TabWidget ... />
            <FrameLayout
                android:id="@android:id/tabcontent" >
                <!-- Aquí se agregan los componentes de las pestañas -->
            </FrameLayout>
        </LinearLayout>
    </TabHost>
</LinearLayout>
```

Paso 2. Insertar los componentes, resaltados con letras negritas de cada pestaña:

```
<LinearLayout ... >
    <TabHost android:id="@android:id/tabhost">
        <LinearLayout ... >
            <TabWidget ... />
            <FrameLayout android:id="@android:id/tabcontent" >
                <LinearLayout android:id="@+id/xtab1" >
                <TextView android:id="@+id/xtv1" />
                <Button ... />
                <Button ... />
                </LinearLayout>
                <LinearLayout android:id="@+id/xtab2" >
                <TextView android:id="@+id/xtv2" />
                <EditText ... />
                <Button ... />
                </LinearLayout>
            </FrameLayout>
        </LinearLayout>
    </TabHost>
</LinearLayout>
```

Paso 3. Una vez agregados los componentes, se definen sus propiedades. El código final del archivo **activity_main.xml**, es el siguiente:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_height="match_parent"
    android:layout_width="match_parent">
    <TabHost
        android:id="@android:id/tabhost"
        android:layout_width="match_parent"
        android:layout_height="match_parent">
        <LinearLayout
            android:orientation="vertical"
            android:layout_width="match_parent"
            android:layout_height="match_parent" >
            <TabWidget
                android:id="@android:id/tabs"
                android:layout_width="match_parent"
                android:layout_height="wrap_content" />
            <FrameLayout
```



```
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@android:id/tabcontent" >
        <LinearLayout
            android:id="@+id/xtab1"
            android:orientation="vertical"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:background="@color/colortab1" >
            <TextView
                android:id="@+id/xtv1"
                android:text="\n Contenido de Tab 1\n"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:background="@color/colortxt" />
            <Button
                android:text="Opcion A"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content" />
            <Button
                android:text="Opcion B"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content" />
        </LinearLayout>
        <LinearLayout
            android:id="@+id/xtab2"
            android:orientation="vertical"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:background="@color/colortab2">
            <TextView
                android:id="@+id/xtv2"
                android:text="\n Contenido de Tab 2\n"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:background="@color/colortxt" />
            <EditText
                android:text="Escribir aquí..."
                android:layout_width="wrap_content"
                android:layout_height="wrap_content" />
            <Button
                android:text="Digitar"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content" />
        </LinearLayout>
    </FrameLayout>
</LinearLayout>
</TabHost>
</LinearLayout>
```

Paso 4. En la carpeta `java/com.example.mipaquete`, abrir el archivo `MainActivity.java` y modificarlo con el siguiente código:

```
import android.os.*;
import android.app.*;
import android.content.res.*;
import android.view.*;
import android.widget.*;
```



```
import android.widget.TabHost.*;
public class MainActivity extends Activity {
    Resources r;
    TabHost th;
    TabSpec ts;
    public void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_main);
        r = getResources();
        th = (TabHost) findViewById(android.R.id.tabhost);
        th.setup();
        ts = th.newTabSpec("mitab1");
        ts.setContent(R.id.xtab1);
        ts.setIndicator("TAB1", r.getDrawable(android.R.drawable.ic_btn_speak_now));
        th.addTab(ts);
        ts = th.newTabSpec("mitab2");
        ts.setContent(R.id.xtab2);
        ts.setIndicator("TAB2", r.getDrawable(android.R.drawable.ic_dialog_map));
        th.addTab(ts);
        th.setCurrentTab(0);
        th.setOnTabChangeListener(new OnTabChangeListener() {
            public void onTabChanged(String tabId) {
                Toast.makeText(getApplicationContext(), "Pestaña seleccionada: " +
tabId, Toast.LENGTH_SHORT).show();
            }
        });
    }
}
```

Paso 5. En la carpeta **res/values**, abrir y modificar el archivo **strings.xml**, con el siguiente código:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#3F51B5</color>
    <color name="colorPrimaryDark">#303F9F</color>
    <color name="colorAccent">#FF4081</color>
    <color name="colortab1">#303F9F</color>
    <color name="colortab2">#FF4081</color>
    <color name="colortxt">#FFAAAA</color>
</resources>
```

Paso 6. Por último, al ejecutar la aplicación se debe mostrar una imagen similar a la siguiente con el código terminado. Al digitar la pestaña TAB1 o TAB2, se muestra su contenido correspondiente. Ver la figura 2.



Figura 2. Prueba de la selección de pestañas.



EJEMPLO 2.

Paso 1. En este segundo ejemplo sí se utilizan fragmentos.

Crear un nuevo proyecto **Pestañas2**. En la carpeta `java/com.example.mipaquete`, abrir el archivo predeterminado `MainActivity.java` para modificarlo con el siguiente código:

```
import android.os.Bundle;
import android.support.v4.app.FragmentActivity;
import android.support.v4.app.FragmentManager;
import android.support.v4.app.FragmentTabHost;
public class MainActivity extends FragmentActivity {
    private FragmentTabHost tabHost;
    @Override
    protected void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_main);
        tabHost= (FragmentTabHost) findViewById(android.R.id.tabhost);
        tabHost.setup(this, getSupportFragmentManager(),android.R.id.tabcontent);
        tabHost.addTab(tabHost.newTabSpec("tab1").setIndicator("Pestaña 1"), Tab1.class, null);
        tabHost.addTab(tabHost.newTabSpec("tab2").setIndicator("Pestaña 2"), Tab2.class, null);
        tabHost.addTab(tabHost.newTabSpec("tab3").setIndicator("Pestaña 3"), Tab3.class, null);
    }
}
```

Ahora, se crean los fragmentos `Tab1.java`, `Tab2.java`, `Tab3.java`.

Paso 2. En la carpeta `java/com.example.mipaquete`, crear y abrir el archivo predeterminado `Tab1.java` para agregar el siguiente código:

```
import android.os.*;
import android.support.v4.app.Fragment;
import android.view.*;
public class Tab1 extends Fragment {
    @Override
    public void onCreate(Bundle b) {
        super.onCreate(b);
    }
    @Override
    public View onCreateView(LayoutInflater li, ViewGroup vg, Bundle bn) {
        return li.inflate(R.layout.tab1, vg, false);
    }
}
```

Paso 3. En la carpeta `java/com.example.mipaquete`, crear y abrir el archivo predeterminado `Tab2.java` para agregar el siguiente código:

```
import android.os.*;
import android.support.v4.app.Fragment;
import android.view.*;
public class Tab2 extends Fragment {
    @Override
    public void onCreate(Bundle b) {
        super.onCreate(b);
    }
    @Override
    public View onCreateView(LayoutInflater li, ViewGroup vg, Bundle bn) {
        return li.inflate(R.layout.tab2, vg, false);
    }
}
```

Paso 4. En la carpeta `java/com.example.mipaquete`, crear y abrir el archivo predeterminado `Tab3.java` para agregar el siguiente código:

```
import android.os.*;
import android.support.v4.app.Fragment;
```



```
import android.view.*;
public class Tab3 extends Fragment {
    @Override
    public void onCreate(Bundle b) {
        super.onCreate(b);
    }
    @Override
    public View onCreateView(LayoutInflater li, ViewGroup vg, Bundle bn) {
        return li.inflate(R.layout.tab3, vg, false);
    }
}
```

Paso 5. En la carpeta res/layout, abrir el archivo predeterminado activity_main.xml para modificarlo con el siguiente código:

```
<android.support.v4.app.FragmentTabHost
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@android:id/tabhost"
    android:layout_width="match_parent"
    android:layout_height="match_parent" >
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical" >
        <TabWidget
            android:id="@android:id/tabs"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="0"
            android:orientation="horizontal" />
        <FrameLayout
            android:id="@android:id/tabcontent"
            android:layout_width="match_parent"
            android:layout_height="0dp"
            android:layout_weight="1" />
    </LinearLayout>
</android.support.v4.app.FragmentTabHost>
```

Ahora, se crean los fragmentos tab1.xml, tab2.xml, tab3.xml.

Paso 6. En la carpeta res/layout, crear y abrir el archivo tab1.xml para ingresar el siguiente código:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
    <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">
        <LinearLayout
            android:orientation="vertical"
            android:layout_width="match_parent"
            android:layout_height="match_parent">
            <TextView
                android:id="@+id/xtv1"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:textAppearance="?android:attr/textAppearanceMedium"
                android:text="\nPestaña 1" />
            <EditText
                android:layout_width="match_parent"
```



```

        android:layout_height="wrap_content"
        android:gravity="center_vertical|center_horizontal"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:text="Escribir aquí..." />
    </LinearLayout>
</FrameLayout>
</LinearLayout>

```

Paso 7. En la carpeta res/layout, crear y abrir el archivo tab2.xml para ingresar el siguiente código:

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
    <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">
        <LinearLayout
            android:orientation="vertical"
            android:layout_width="match_parent"
            android:layout_height="match_parent">
            <TextView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:textAppearance="?android:attr/textAppearanceMedium"
                android:text="\nPestaña 2" />
            <EditText
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:gravity="center_vertical|center_horizontal"
                android:textAppearance="?android:attr/textAppearanceMedium"
                android:text="Escribir aquí..." />
            <Button
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:gravity="center_vertical|center_horizontal"
                android:textAppearance="?android:attr/textAppearanceMedium"
                android:text="Opcion 1" />
        </LinearLayout>
    </FrameLayout>
</LinearLayout>

```

Paso 8. En la carpeta res/layout, crear y abrir el archivo tab3.xml para ingresar el siguiente código:

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
    <FrameLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">
        <LinearLayout
            android:orientation="vertical"
            android:layout_width="match_parent"
            android:layout_height="match_parent">
            <TextView
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:textAppearance="?android:attr/textAppearanceMedium"
                android:text="\nPestaña 3" />

```




```

<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center_vertical|center_horizontal"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:text="Opcion 1" />
<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center_vertical|center_horizontal"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:text="Opcion 2" />
</LinearLayout>
</FrameLayout>
</LinearLayout>

```

Paso 9. Por último, ejecutar la aplicación. La imagen debe ser similar a la siguiente. Digitar en cada pestaña para mostrar su correspondiente contenido:



NOTA. Generar un reporte con las imágenes obtenidas al ejecutar las aplicaciones. Guardar el documento con la sintaxis AlumnoPestañasGrupo.pdf y enviarlo al sitio indicado por el profesor.

EJERCICIO.

Ejecutar el siguiente código para mostrarlo dentro de una pestaña en uno de los ejemplos anteriores.

```

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class K extends JPanel{
    JFrame jf;
    JLabel jl;
    float x, y, dir;
    int level = 1;
    public K(){

```



```
jf = new JFrame("K");
jl = new JLabel("Digitar varias veces sobre la imagen..... 'soy un copo de nieve'
.....");
jf.add("North", jl);
jf.setSize(400, 400);
addMouseListener(new MouseAdapter(){
    public void mousePressed(MouseEvent evt){
        level++;
        repaint();
    }
});
jf.add(this);
jf.setVisible(true);
}

public void paintComponent(Graphics g){
    int length = getWidth()/2;
    y = (float)(getHeight()/2 + Math.sin(Math.toRadians(30))*length/2);
    x = this.getWidth()/4;
    g.setColor(Color.white);
    g.fillRect(0, 0, getWidth(), this.getHeight());
    g.setColor(Color.black);
    dir = 0;
    dK(g, length, level);
    dir -= 120;
    dK(g, length, level);
    dir -= 120;
    dK(g, length, level);
}

public void dK(Graphics g, double len, int n){
    if (n==0){
        double dirRad = Math.toRadians(dir);
        double xInc = len * Math.cos(dirRad); // x increment
        double yInc = len * Math.sin(dirRad); // y increment
        float x1 = x + (float)xInc;
        float y1 = y + (float)yInc;
        g.drawLine((int)x, (int)y, (int)x1, (int)y1);
        x = x1;
        y = y1;
    }else{
        dK(g, len/=3, --n);
        dir += 60;
        dK(g, len, n);
        dir -= 120;
        dK(g, len, n);
        dir += 60;
        dK(g, len, n);
    }
}

public static void main(String [] args){
    new K();
}
}
```