

View binding

View binding

Data binding permite definir variables en los layouts. Así, mediante el uso de expresiones, se puede asignar el valor de dichas variables en los atributos de las views.

A través de data binding se simplifica la programación de nuestras activities puesto que, por ejemplo, no será necesario la utilización de `findViewById()`. Data binding vincula directamente el *layout* con los datos de la aplicación.

La utilización de LiveData como variables, implicaría la permanente actualización de la UI

View binding

Activación

Module: build.gradle

```
: : :  
android {  
    : : :  
  
    buildFeatures {  
        viewBinding true  
    }  
}
```

View binding

MainViewModel

```
public class MainViewModel extends ViewModel
{
    public LiveData<String> id;
    public LiveData<String> name;
    : : :
    public void ok() {...}
}
```

```
public class MainActivity extends AppCompatActivity
{
    private ActivityMainBinding viewBinding;
```

Se genera una clase cuyo nombre se deriva del identificador del *layout* (camel case sin subrayados más el sufijo **Binding**)

R.layout.activity_main \Rightarrow **ActivityMainBinding**

```
}
```

R.layout.activity_main

```
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/
    : : :>

    <TextView
        android:id="@+id/tvId"
        : : :/>

    <EditText
        android:id="@+id/etName"
        : : : />

    <RadioGroup
        android:id="@+id/rgOpciones"
        : : :>
        <RadioButton
            android:id="@+id/rbOpcionA"
            : : :/>
        <RadioButton
            android:id="@+id/rbOpcionB"
            : : :/>
        <RadioButton
            android:id="@+id/rbOpcionC"
            : : :/>
    </RadioGroup>

    <Button
        android:id="@+id/btOK"
        : : :/>

</androidx.constraintlayout.widget.ConstraintLayout>
```

View binding

MainViewModel

```
public class MainViewModel extends ViewModel
{
    public LiveData<String> id;
    public LiveData<String> name;
    : : :
    public void ok() {...}
}
```

```
public class MainActivity extends AppCompatActivity
{
    private ActivityMainBinding viewBinding;

    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        this.viewBinding = ActivityMainBinding.inflate( getLayoutInflater() );
    }
}
```

Se obtiene el objeto **viewBinding** «inflando» el *layout*. Para ello se invoca al método **inflate()** de la clase *view binding* pasándole como parámetro el objeto **LayoutInflater** que nos devuelve el método **getLayoutInflater()** de la *activity*

R.layout.activity_main

```
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/
    : : :>

    <TextView
        android:id="@+id/tvId"
        : : :/>

    <EditText
        android:id="@+id/etName"
        : : : />

    <RadioGroup
        android:id="@+id/rgOpciones"
        : : :>
        <RadioButton
            android:id="@+id/rbOpcionA"
            : : :/>
        <RadioButton
            android:id="@+id/rbOpcionB"
            : : :/>
        <RadioButton
            android:id="@+id/rbOpcionC"
            : : :/>
    </RadioGroup>

    <Button
        android:id="@+id/btOK"
        : : :/>

</androidx.constraintlayout.widget.ConstraintLayout>
```

View binding

MainViewModel

```
public class MainViewModel extends ViewModel
{
    public LiveData<String> id;
    public LiveData<String> name;
    : : :
    public void ok() {...}
}
```

```
public class MainActivity extends AppCompatActivity
{
    private ActivityMainBinding viewBinding;

    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        this.viewBinding = ActivityMainBinding.inflate( getLayoutInflater() );
    }
}
```

El objeto *viewBinding* representa el *layout*, de forma que contendrá atributos `public` correspondientes a cada widget incluido en el *layout* para el que se haya definido un **`android:id`**. El nombre del atributo coincide con el identificador asignado al *widget*, el tipo del atributo coincide con el tipo del *widget*.

Adicionalmente, el objeto *viewBinding* contiene un método **`getRoot()`** que devuelve la *view* correspondiente a la vista raíz del *layout* (en el ejemplo `ConstraintLayout`)

R.layout.activity_main

```
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/
    : : :>

    <TextView
        android:id="@+id/tvId"
        : : :/>

    <EditText
        android:id="@+id/etName"
        : : :/>

    <RadioGroup
        android:id="@+id/rgOpciones"
        : : :>
        <RadioButton
            android:id="@+id/rbOpcionA"
            : : :/>
        <RadioButton
            android:id="@+id/rbOpcionB"
            : : :/>
        <RadioButton
            android:id="@+id/rbOpcionC"
            : : :/>
    </RadioGroup>

    <Button
        android:id="@+id/btOK"
        : : :/>

</androidx.constraintlayout.widget.ConstraintLayout>
```

View binding

MainViewModel

```
public class MainViewModel extends ViewModel
{
    public LiveData<String> id;
    public LiveData<String> name;
    : : :
    public void ok() {...}
}
```

```
public class MainActivity extends AppCompatActivity
{
    private ActivityMainBinding viewBinding;

    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        this.viewBinding = ActivityMainBinding( this.getLayoutInflater() );
        setContentView( this.viewBinding.getRoot() );
    }
}
```

Se fija cuál será vista de la activity (el ConstraintLayout de nuestro layout)

R.layout.activity_main

```
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/
    : : :>

    <TextView
        android:id="@+id/tvId"
        : : :/>

    <EditText
        android:id="@+id/etName"
        : : : />

    <RadioGroup
        android:id="@+id/rgOpciones"
        : : :>
        <RadioButton
            android:id="@+id/rbOpcionA"
            : : :/>
        <RadioButton
            android:id="@+id/rbOpcionB"
            : : :/>
        <RadioButton
            android:id="@+id/rbOpcionC"
            : : :/>
    </RadioGroup>

    <Button
        android:id="@+id/btOK"
        : : :/>

</androidx.constraintlayout.widget.ConstraintLayout>
```

View binding

MainViewModel

```
public class MainViewModel extends ViewModel
{
    public LiveData<String> id;
    public LiveData<String> name;
    : : :
    public void ok() {...}
}
```

```
public class MainActivity extends AppCompatActivity
{
    private ActivityMainBinding viewBinding;
    private MainViewModel viewModel;

    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        this.viewBinding = ActivityMainBinding( this.getLayoutInflater() );
        setContentView( this.viewBinding.getRoot() );
        this.viewModel = new ViewModelProvider(this).get(MainViewModel.class);
    }
}
```

Se obtiene el *viewModel* de la *activity*, que contendrá los datos a mostrar en el UI

R.layout.activity_main

```
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/
    : : :>

    <TextView
        android:id="@+id/tvId"
        : : :/>

    <EditText
        android:id="@+id/etName"
        : : : />

    <RadioGroup
        android:id="@+id/rgOpciones"
        : : :>
        <RadioButton
            android:id="@+id/rbOpcionA"
            : : :/>
        <RadioButton
            android:id="@+id/rbOpcionB"
            : : :/>
        <RadioButton
            android:id="@+id/rbOpcionC"
            : : :/>
    </RadioGroup>

    <Button
        android:id="@+id/btOK"
        : : :/>

</androidx.constraintlayout.widget.ConstraintLayout>
```


View binding

MainViewModel

```
public class MainViewModel extends ViewModel
{
    public LiveData<String> id;
    public LiveData<String> name;
    : : :
    public void ok() {...}
}
```

```
public class MainActivity extends AppCompatActivity
{
    private ActivityMainBinding viewBinding;
    private MainViewModel viewModel;

    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        this.viewBinding = ActivityMainBinding( this.getLayoutInflater() );
        setContentView( this.viewBinding.getRoot() );
        this.viewModel = new ViewModelProvider(this).get(MainViewModel.class);
        this.viewModel.id.observe(this, (v) -> viewBinding.tvId.setText(v) );
    }
}
```

Para el LiveData **id** del *viewModel* se crea un *observador* para que fije su valor en el TextView "**@+id/tvId**" del *layout* (accesible a través del atributo **viewBinding.tvId**)

R.layout.activity_main

```
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/
    : : :>

    <TextView
        android:id="@+id/tvId"
        : : :/>

    <EditText
        android:id="@+id/etName"
        : : :/>

    <RadioGroup
        android:id="@+id/rgOpciones"
        : : :>
        <RadioButton
            android:id="@+id/rbOpcionA"
            : : :/>
        <RadioButton
            android:id="@+id/rbOpcionB"
            : : :/>
        <RadioButton
            android:id="@+id/rbOpcionC"
            : : :/>
    </RadioGroup>

    <Button
        android:id="@+id/btOK"
        : : :/>

</androidx.constraintlayout.widget.ConstraintLayout>
```

View binding

MainViewModel

```
public class MainViewModel extends ViewModel
{
    public LiveData<String> id;
    public LiveData<String> name;
    : : :
    public void ok() {...}
}
```

```
public class MainActivity extends AppCompatActivity
{
    private ActivityMainBinding viewBinding;
    private MainViewModel viewModel;

    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        this.viewBinding = ActivityMainBinding( this.getLayoutInflater() );
        setContentView( this.viewBinding.getRoot() );
        this.viewModel = new ViewModelProvider(this).get(MainViewModel.class);
        this.viewModel.id.observe(this, (v) -> viewBinding.tvId.setText(v) );
        this.viewModel.name.observe(this, viewBinding.etName::setText);
    }
}
```

Se realiza la misma operación pero con el LiveData name.
Ahora se utiliza otra sintaxis más breve

R.layout.activity_main

```
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/
    : : :>

    <TextView
        android:id="@+id/tvId"
        : : :/>

    <EditText
        android:id="@+id/etName"
        : : : />

    <RadioGroup
        android:id="@+id/rgOpciones"
        : : :>
        <RadioButton
            android:id="@+id/rbOpcionA"
            : : :/>
        <RadioButton
            android:id="@+id/rbOpcionB"
            : : :/>
        <RadioButton
            android:id="@+id/rbOpcionC"
            : : :/>
    </RadioGroup>

    <Button
        android:id="@+id/btOK"
        : : :/>

</androidx.constraintlayout.widget.ConstraintLayout>
```

View binding

MainViewModel

```
public class MainViewModel extends ViewModel
{
    public LiveData<String> id;
    public LiveData<String> name;
    : : :
    public void ok() {...}
}
```

```
public class MainActivity extends AppCompatActivity
{
    private ActivityMainBinding viewBinding;
    private MainViewModel viewModel;

    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        this.viewBinding = ActivityMainBinding( this.getLayoutInflater() );
        setContentView( this.viewBinding.getRoot() );
        this.viewModel = new ViewModelProvider(this).get(MainViewModel.class);
        this.viewModel.id.observe(this, (v) -> viewBinding.tvId.setText(v) );
        this.viewModel.name.observe(this, viewBinding.etName::setText);
        viewBinding.btOK.setOnClickListener((v)-> viewModel.ok());
    }
```

Igualmente, se puede fijar el *listener on click* del Button "@+id/btOK" del *layout* (accesible a través del atributo `viewBinding.btOK`)

R.layout.activity_main

```
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/
    : : :>

    <TextView
        android:id="@+id/tvId"
        : : :/>

    <EditText
        android:id="@+id/etName"
        : : : />

    <RadioGroup
        android:id="@+id/rgOpciones"
        : : :>
        <RadioButton
            android:id="@+id/rbOpcionA"
            : : :/>
        <RadioButton
            android:id="@+id/rbOpcionB"
            : : :/>
        <RadioButton
            android:id="@+id/rbOpcionC"
            : : :/>
    </RadioGroup>

    <Button
        android:id="@+id/btOK"
        : : :/>

</androidx.constraintlayout.widget.ConstraintLayout>
```