



PATRONES GOF

Ingeniería del Software II

Nicolás Rubiano

Angie Peñaloza

Facultad de Ingeniería y Ciencias básicas.
Politécnico Grancolombiano.

Contenido

INTRODUCCION	3
SINGLETON	3
Definición	3
¿Cómo se implementa?.....	3
Única instancia privada y estática	3
Constructor privado.	4
Obtener la instancia del singleton y asignar los datos del servlet.	5
Obtener los datos mediante la instancia	6

INTRODUCCION

La Federación Nacional de Cafeteros con el fin de acercar a los caficultores y de buscar una comunicación y distribución del material audiovisual y didáctico de una forma más eficiente, solicita el desarrollo de una aplicación móvil que realice las actividades anteriormente nombradas. Es por esto que se define el uso de los siguientes patrones los cuales serán implementados en la aplicación móvil.

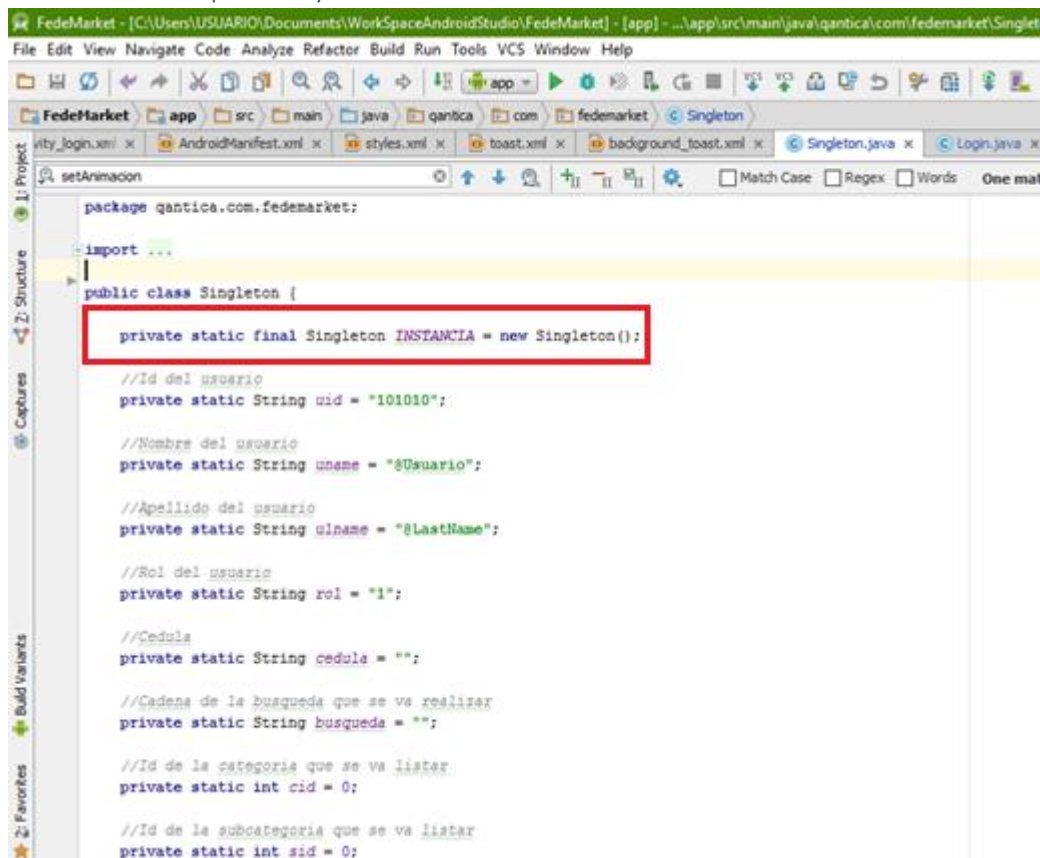
SINGLETON

Definición

Singleton es un patrón de diseño que garantiza que una clase sólo tenga una instancia y proporciona un punto de acceso global a ella.

¿Cómo se implementa?

Única instancia privada y estática



```
package qantica.com.fedemarket;

import ...

public class Singleton {

    private static final Singleton INSTANCIA = new Singleton();

    //Id del usuario
    private static String uid = "101010";

    //Nombre del usuario
    private static String uname = "8Usuario";

    //Apellido del usuario
    private static String ulname = "@LastName";

    //Rol del usuario
    private static String rol = "1";

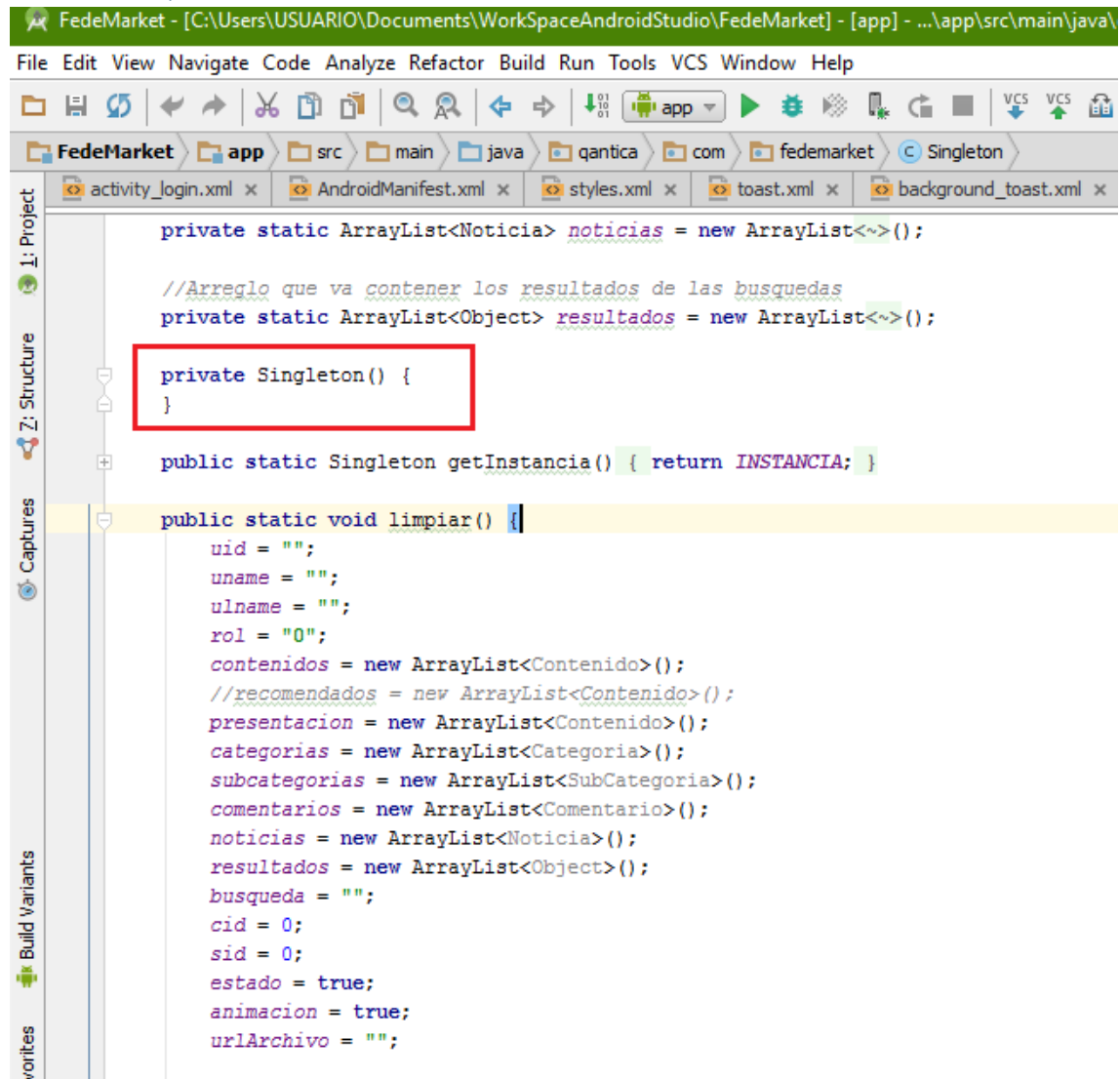
    //Cedula
    private static String cedula = "";

    //Cadena de la busqueda que se va realizar
    private static String busqueda = "";

    //Id de la categoria que se va listar
    private static int cid = 0;

    //Id de la subcategoria que se va listar
    private static int sid = 0;
}
```

Constructor privado.



```
FedeMarket - [C:\Users\USUARIO\Documents\WorkSpaceAndroidStudio\FedeMarket] - [app] - ...\app\src\main\java\
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

FedeMarket app src main java qantica com fedemarket Singleton
activity_login.xml x AndroidManifest.xml x styles.xml x toast.xml x background_toast.xml x

private static ArrayList<Noticia> noticias = new ArrayList<>();

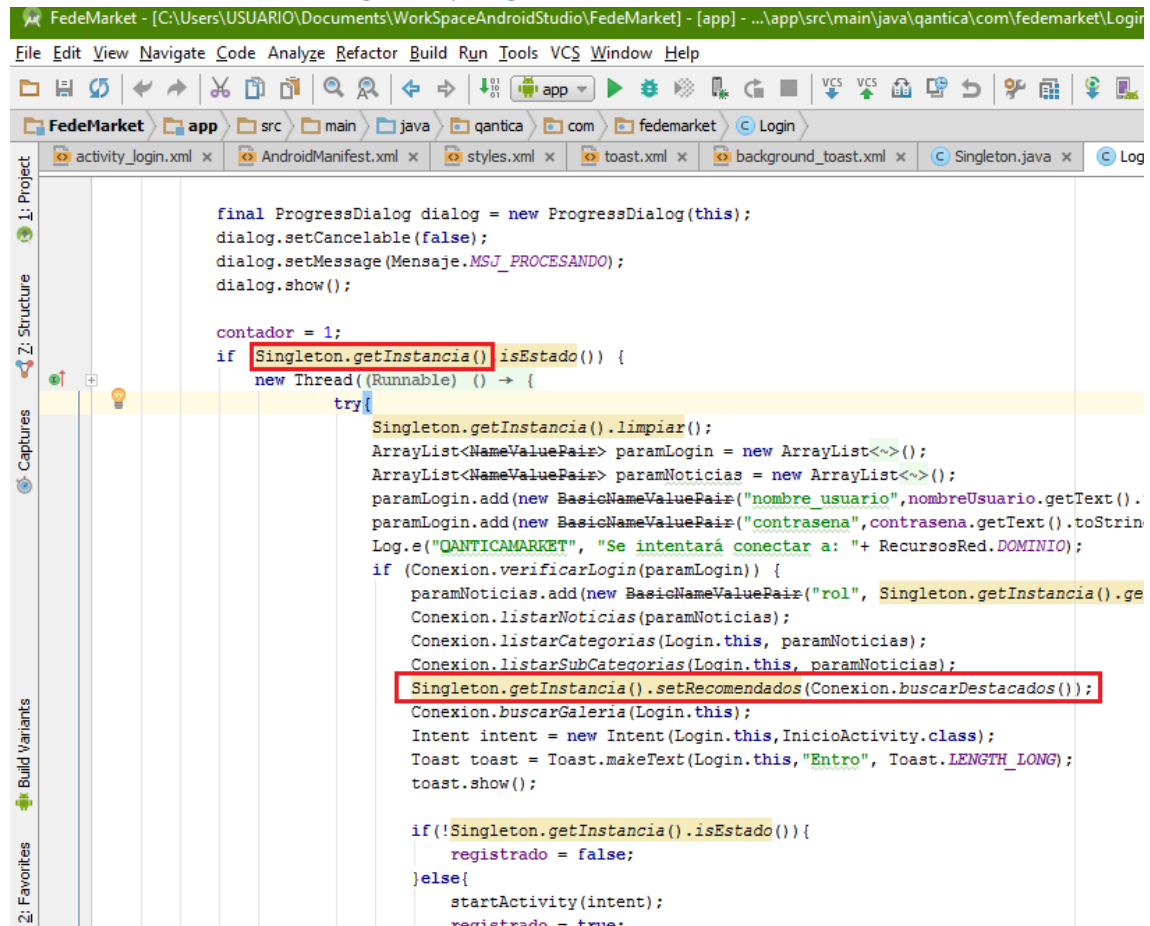
//Arreglo que va contener los resultados de las busquedas
private static ArrayList<Object> resultados = new ArrayList<>();

private Singleton() {
}

public static Singleton getInstance() { return INSTANCIA; }

public static void limpiar() {
    uid = "";
    uname = "";
    ulname = "";
    rol = "0";
    contenidos = new ArrayList<Contenido>();
    //recomendados = new ArrayList<Contenido>();
    presentacion = new ArrayList<Contenido>();
    categorias = new ArrayList<Categoria>();
    subcategorias = new ArrayList<SubCategoria>();
    comentarios = new ArrayList<Comentario>();
    noticias = new ArrayList<Noticia>();
    resultados = new ArrayList<Object>();
    busqueda = "";
    cid = 0;
    sid = 0;
    estado = true;
    animacion = true;
    urlArchivo = "";
}
```

Obtener la instancia del singleton y asignar los datos del servlet.

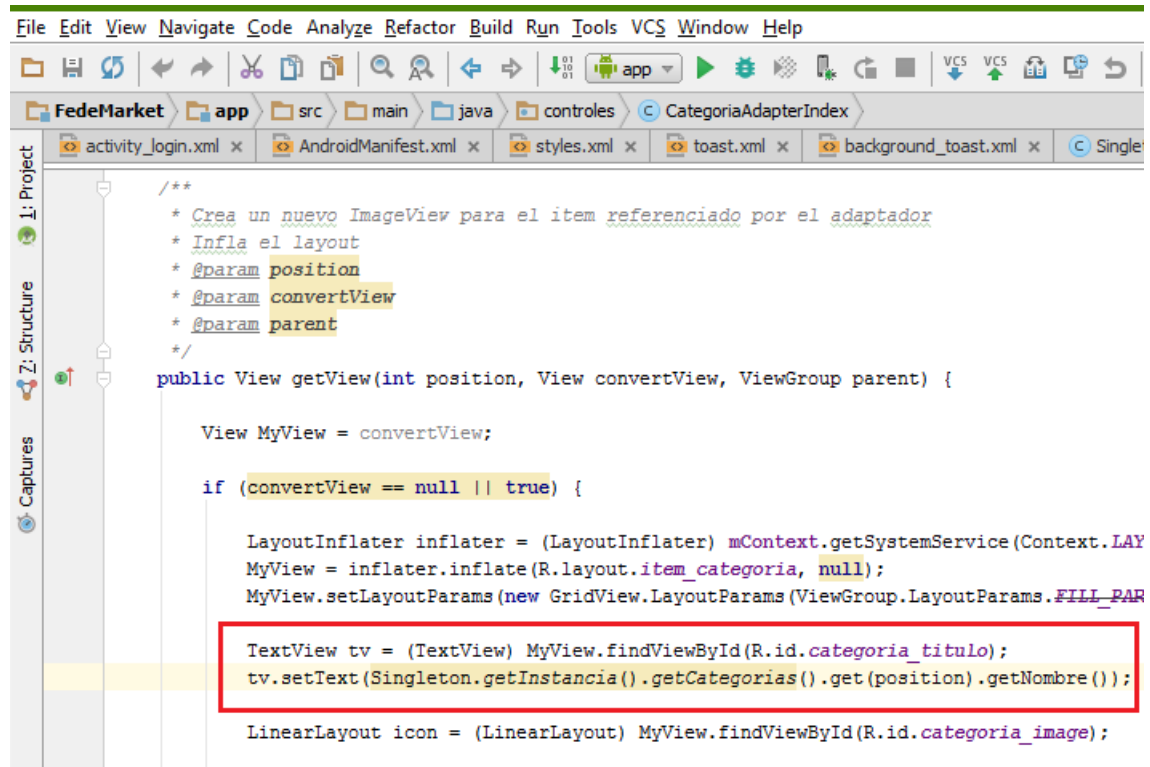


```
final ProgressDialog dialog = new ProgressDialog(this);
dialog.setCancelable(false);
dialog.setMessage(Mensaje.MSJ_PROCESANDO);
dialog.show();

contador = 1;
if (Singleton.getInstance().isEstado()) {
    new Thread((Runnable) () -> {
        try {
            Singleton.getInstance().limpiar();
            ArrayList<NameValuePair> paramLogin = new ArrayList<>();
            ArrayList<NameValuePair> paramNoticias = new ArrayList<>();
            paramLogin.add(new BasicNameValuePair("nombre_usuario", nombreUsuario.getText().toString()));
            paramLogin.add(new BasicNameValuePair("contrasena", contrasena.getText().toString()));
            Log.e("QANTICAMARKET", "Se intentará conectar a: " + RecursosRed.DOMINIO);
            if (Conexion.verificarLogin(paramLogin)) {
                paramNoticias.add(new BasicNameValuePair("rol", Singleton.getInstance().getRol()));
                Conexion.listarNoticias(paramNoticias);
                Conexion.listarCategorias(Login.this, paramNoticias);
                Conexion.listarSubCategorias(Login.this, paramNoticias);
                Singleton.getInstance().setRecomendados(Conexion.buscarDestacados());
                Conexion.buscarGaleria(Login.this);
                Intent intent = new Intent(Login.this, InicioActivity.class);
                Toast toast = Toast.makeText(Login.this, "Entro", Toast.LENGTH_LONG);
                toast.show();

                if (!Singleton.getInstance().isEstado()) {
                    registrado = false;
                } else {
                    startActivity(intent);
                    registrado = true;
                }
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }).start();
}
```

Obtener los datos mediante la instancia



```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
FedeMarket app src main java controles CategoriaAdapterIndex
activity_login.xml x AndroidManifest.xml x styles.xml x toast.xml x background_toast.xml x Singleton

/**
 * Crea un nuevo ImageView para el item referenciado por el adaptador
 * Infla el layout
 * @param position
 * @param convertView
 * @param parent
 */
public View getView(int position, View convertView, ViewGroup parent) {

    View MyView = convertView;

    if (convertView == null || true) {

        LayoutInflater inflater = (LayoutInflater) mContext.getSystemService(Context.LAY
        MyView = inflater.inflate(R.layout.item_categoria, null);
        MyView.setLayoutParams(new GridView.LayoutParams(ViewGroup.LayoutParams.FILL_PAR

        TextView tv = (TextView) MyView.findViewById(R.id.categoria_titulo);
        tv.setText(Singleton.getInstancia().getCategorias().get(position).getNombre());

        LinearLayout icon = (LinearLayout) MyView.findViewById(R.id.categoria_image);
```