

[ESCRIBIR EL NOMBRE DE LA COMPAÑÍA]

Creación de un AsyncTask en Android

Juan Antonio Japón de la Torre

Juan

06/09/2016

Índice

Creación de un AsyncTask en Android	3
Implementando métodos necesarios: doInBackground y onPostExecute	3
Como ejecutar el AsyncTask desde una vista	4
Implementando el método doInBackground	5
Implementando el método onPostExecute	5
Código Final de la Actividad Principal	6

Creación de un AsyncTask en Android

Un async task también conocido como hilo es una operación que se realiza en segundo plano en nuestra aplicación con el objetivo de no colapsar la memoria de la app.

En Android los hilos se utilizan cuando tenemos que realizar operaciones con un retardo mayor a 1 Segundo ya que por encima de dicho tiempo la aplicación se cierra.

Para comenzar a crear un async task tenemos que crear dentro de nuestro Activity una clase interna que extienda de AsyncTask:

```
public class tareaInsertEnDB extends AsyncTask<Parametros, Progreso,
Resultado> {
}
}
```

Como se puede comprobar entre llaves el primer parámetro corresponde a:

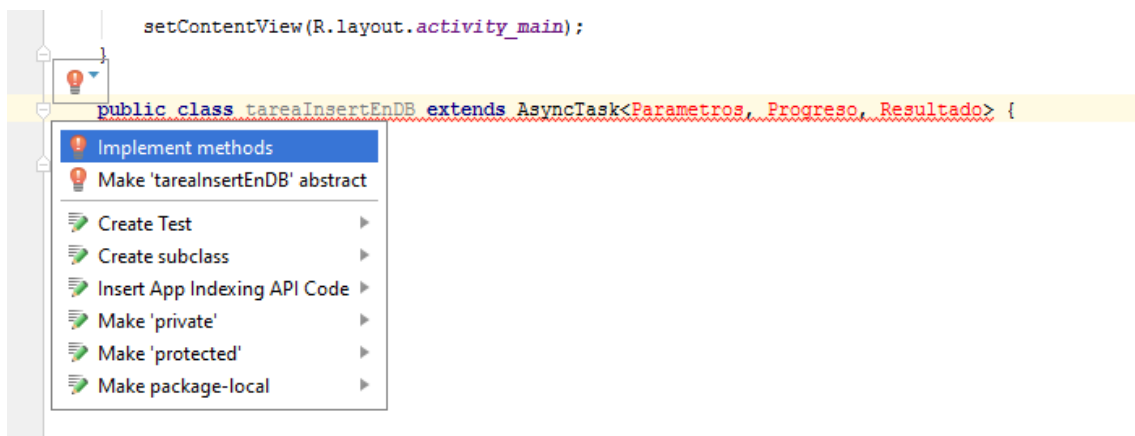
- los parámetros: que le pasaremos nosotros al async task para trabajar con ellos dentro de la tarea
- Progreso: por defecto será Void ya que en principio no devuelve nada, este segundo parámetro se utiliza para el caso de tener spinners u otro componente que tenga una barra de progreso
- Resultado: es el valor que devolveremos en el método **doInBackground**. Dicho resultado se utilizará en el método **OnPostExecute** para mostrarle al usuario el resultado de la operación que realizó

Implementando métodos necesarios: doInBackground y OnPostExecute

Dentro de los AsyncTask tenemos 3 métodos importantes de los cuales trabajaremos con solo 2 en principio que son:

- **doInBackground**: Es la parte más importante donde tenemos que realizar la tarea propiamente dicha
- **OnPostExecute**: Este método se usa para mostrar en el interfaz de usuario el resultado de la tarea

Para implementarlos podemos importarlos o bien escribiéndolos o bien pulsando en la ayuda que aparece abajo.



Nos quedaría de la manera que se muestra abajo de forma inicial:

```
public class tareaInsertEnDB extends AsyncTask<Parametros, Progreso,
Resultado> {
    @Override
    protected Resultado doInBackground(Parametros... params) {
        return null;
    }
    @Override
    protected void onPostExecute(Resultado resultado) {
        super.onPostExecute(resultado);
    }
}
```

Donde pone Parámetros, Progreso y Resultado debemos substituirlo por el tipo de datos que esperamos que devuelva las operaciones. Por ejemplo si hago un insert en base de datos a través de un AsyncTask los tipos que usaríamos sería:

- **Parametros (String):** ya que a la tarea le pasaríamos variables tipo string como dato
- **Progreso(Void);** ya que como no estamos implementando el progreso no devuelve nada
- **Resultado(String);** En este caso el servicio web que utilizaremos para enviar y crear por ejemplo un usuario nos devolverá una respuesta en forma de cadena

En base a todo esto nos quedaría el código del AsyncTask de la siguiente manera:

```
public class tareaInsertEnDB extends AsyncTask<String, Void,
String> {
    @Override
    protected String doInBackground(String... params) {
        //Eventos de la tarea
        return null;
    }
    @Override
    protected void onPostExecute(String resultado) {
        super.onPostExecute(resultado);
        //Resultado a mostrar al usuario
    }
}
```

Como ejecutar el AsyncTask desde una vista

Para llamar a la tarea creada con anterioridad debemos:

New **NombreDelAsynctaskCreado().execute(parametro1,parametro2...);**

Ejemplo: `new TareaInsertEnDB().execute(u,p,e);`
`new TareaInsertEnDB().execute("jajapon", "1234", "jajapon@gmail.com");`

Siendo u,p,e los parametros que vayamos a pasarle (usuario, contraseña, email) en este caso.

Como aun no hemos desarrollado los métodos **doInBackground** y **onPostExecute** no ejecutamos la aplicación

Implementando el método **doInBackground**

Como le pasamos 3 parametros anteriormente a la hora de cargar el AsyncTask para recogerlos debemos

```
@Override
protected String doInBackground(String... params) {
    //Recuperamos los datos que pasamos anteriormente al asyncTask
    String usu = params[0]; // jajapon
    String con = params[1]; // 1234
    String ema = params[2]; // jajapon@gmail.com

    //Realizamos la operación
    String urlConsulta =
    "http://10.0.2.2:8080/WebServices/alta_usuario.php?username="+usu+"&userpass="+con+"&email="+ema;
    String respuesta = Util.getResultadoUrl(urlConsulta);

    //devolvemos la respuesta (mensaje)
    return respuesta;
}
```

Recursos Necesarios para trabajar **(Importante):**

- Clase **Util**:
<https://github.com/jajapon/Android/blob/master/Ejercicios%20de%20Practica/inserteNDB/app/src/main/java/com/insertendb/android/insertendb/Util.java>
- WebService **de Alta de Usuario**
https://github.com/jajapon/Android/blob/master/Ejercicios%20de%20Practica/WebServices/alta_usuario.php

Implementando el método **onPostExecute**

Así quedaría el método. En el estamos mostrándole al usuario un mensaje en caso de que no se Cree el usuario y en caso de que se cree

```
@Override
protected void onPostExecute(String resultado) {
    super.onPostExecute(resultado);
    //Resultado a mostrar al usuario
    super.onPostExecute(resultado);
    if(resultado == null){
        //mensaje en caso de error
        Toast.makeText(getApplicationContext(), "Usuario no
insertado", Toast.LENGTH_SHORT).show();
    }else{
        //mensaje en caso de realizarse satisfactoriamente
        Toast.makeText(getApplicationContext(), "Usuario insertado",
        Toast.LENGTH_SHORT).show();
    }
}
```

Finalmente ejecutamos la aplicación

Código Final de la Actividad Principal

```

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        //Datos a enviar a la tarea
        String u = "jajapon";
        String p = "1234";
        String e = "jajapon@gmail.com";

        //Ejecucion de la tarea
        new TareaInsertEnDB().execute(u,p,e);
    }

    public class TareaInsertEnDB extends AsyncTask<String, Void,
String> {
        @Override
        protected String doInBackground(String... params) {
            //Recuperamos los datos que pasamos anteriormente al
            AsyncTask
            String usu = params[0];
            String con = params[1];
            String ema = params[2];
            //Realizamos la operación
            String urlConsulta =
"http://10.0.2.2:8080/WebServices/alta_usuario.php?username="+usu+"&us
erpass="+con+"&email="+ema;
            String respuesta = Util.getResultadoUrl(urlConsulta);
            //devolvemos la respuesta
            return respuesta;
        }
        @Override
        protected void onPostExecute(String resultado) {
            super.onPostExecute(resultado);
            //Resultado a mostrar al usuario
            super.onPostExecute(resultado);
            if(resultado == null){
                //mensaje en caso de error
                Toast.makeText(getApplicationContext(), "Usuario no
insertado", Toast.LENGTH_SHORT).show();
            }else{
                //mensaje en caso de realizarse satisfactoriamente
                Toast.makeText(getApplicationContext(), "Usuario
insertado", Toast.LENGTH_SHORT).show();
            }
        }
    }
}

```