UNITAT 07

PREFERÈNCIES, DIÁLOGOS & NOTIFICACIONS

[**1) PreferenceActivity**](#_nhoiknhy536k) **2**

[**2) SharedPreferences**](#_k3xiy2a7ev44) **3**

[**3) PreferenceCategory**](#_suflgxc3eszc) **3**

[**4) Tipus de preferències**](#_806l85gljb5) **4**

[4.1) CheckBoxPreference](#_neprnjoxl6ac) 4

[4.2) EditTextPreference](#_onpt3dn3fm4t) 4

[4.3) ListPreference](#_lnuoes4wp46s) 4

[4.4) MultiSelectListPreference](#_ml84j3pr2ql6) 4

[**5) Toast**](#_e5vnzhynifpc) **5**

[**6) Notificacions**](#_n0ug8mcqvxnc) **6**

# 1) PreferenceActivity

Permeten **personalitzar** l’interacció entre la app i l’usuari.

La classe [PreferenceActivity](https://developer.android.com/reference/android/preference/PreferenceActivity), permet mostrar les preferències definides a un **XML**. Aquests fitxers es troben a **res/xml**.

Per a treballar amb preferències, crearem una classe que **herede** de PreferenceActivity:

public class MyPreferenceActivity extends PreferenceActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

**addPreferencesFromResource(R.xml.preferences);**

}

}

Per a llançar les preferències es crea un intent:

startActivity(new Intent(this, MyPreferenceActivity.class));

# 

# 2) SharedPreferences

Per a **obtindre** les preferències s’utilitza la classe [SharedPreferences](https://developer.android.com/reference/android/content/SharedPreferences), gestionada per [PreferenceManager](https://developer.android.com/reference/android/preference/PreferenceManager?hl=en).

Les preferències que **no** han sigut **definides** per l’usuari, rebran el **valor per defecte** que es troba al XML.

SharedPreferences preferences = PreferenceManager.getDefaultSharedPreferences(this);

# 3) PreferenceCategory

**Etiqueta** XML que permet definir **grups** de preferències.

# 4) Tipus de preferències

**Etiquetes** XML que permeten definir preferències:

## 4.1) CheckBoxPreference

Representa una **casella** que es pot **marcar** i només pot tenir **dos valors**.

## 4.2) EditTextPreference

Representa una **finestra emergent** amb dos botons; acceptar i cancelar, on el usuari pot **introduir** una cadena de text.

## 4.3) ListPreference

Representa una **llista de valors** definits prèviament, on el usuari pot triar-ne **sols un**.

**Element -> Clau**: Cada element es relaciona amb un valor clau, també definit al XML.

## 4.4) MultiSelectListPreference

Representa una **llista de valors** definits prèviament, on el usuari pot triar-ne **un o més**.

**Element -> Clau**: Cada element es relaciona amb un valor clau, també definit al XML.

# 5) Toast

Són **missatges** que es **mostren** en pantalla i **desapareixen** automàticament.

Context context = *getApplicationContext*();

CharSequence text = "Hello toast!";

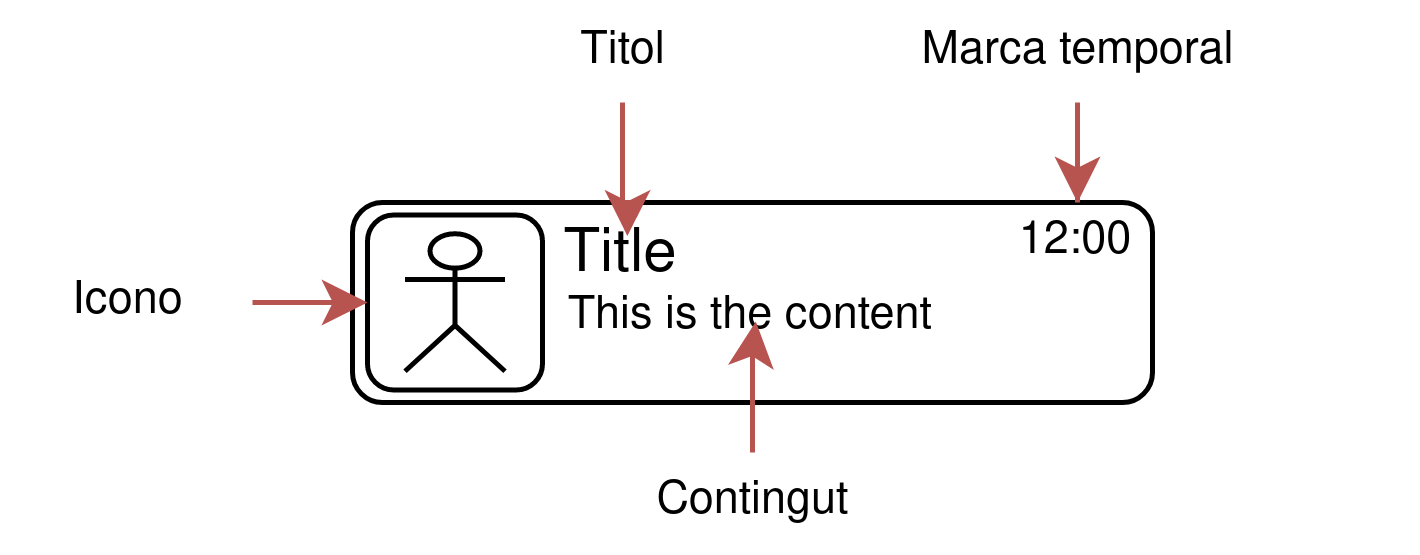
int duration = Toast.*LENGTH\_SHORT*;

Toast toast = Toast.*makeText*(context, text, duration);

toast.show();

# 6) Notificacions

Les notificacions permeten crear **missatges** que es poden **veure** desde **fora** de l'aplicació en pantalla.



/\* Crear el NotificationChannel \*/

private static final String CHANNEL\_ID = "notification\_id";

NotificationChannel channel = new NotificationChannel(CHANNEL\_ID, name, importance);

**NotificationManager notificationManager = getSystemService(NotificationManager.class);**

notificationManager.createNotificationChannel(channel);

NotificationManagerCompat notificationManager = NotificationManagerCompat.from(getApplicationContext());

/\* Acció de la notificació \*/

Intent intent = new Intent(Intent.ACTION\_VIEW);

/\* … \*/

**PendingIntent pendingIntent = PendingIntent.getActivity(getApplicationContext(), 0, intent, PendingIntent.FLAG\_IMMUTABLE);**

/\* Construir la notificació \*/

**NotificationCompat.Builder** builder = new NotificationCompat.Builder(getApplicationContext(), CHANNEL\_ID)

**.setSmallIcon(R.drawable.chimo\_bayo)**

**.setContentTitle("Hu ha!!!")**

**.setContentText("Visita la millor web de la història")**

.setPriority(NotificationCompat.PRIORITY\_DEFAULT)

**.setContentIntent(pendingIntent)**

.setAutoCancel(true);

/\* Mostrar la notificació \*/

**notificationManager.notify(100, builder.build());**