**NOTIFICHE**

**Quali sono i principali modi per fornire notifiche all’utente? Si fornisca una breve descrizione.**

* ***Toast***: Un Toast è una piccola notifica popup che viene utilizzata per visualizzare informazioni sull'operazione che abbiamo eseguito nella nostra app. Toast mostrerà il messaggio per un breve periodo di tempo e scomparirà automaticamente dopo un timeout. Generalmente, la notifica Toast in Android verrà visualizzata con un testo semplice.
* ***Dialog***: una piccola finestra che richiede all'utente di prendere una decisione o inserire informazioni aggiuntive. Una finestra di dialogo non riempie lo schermo e viene normalmente utilizzata per eventi modali che richiedono agli utenti di eseguire un'azione prima di poter procedere.
* ***Notification*** ***Area*** (***Status*** ***Bar***): è un messaggio che Android visualizza all'esterno dell'interfaccia utente dell'app per fornire all'utente promemoria, comunicazioni di altre persone o altre informazioni tempestive dall'app. Gli utenti possono toccare la notifica per aprire l'app o eseguire un'azione direttamente dalla notifica. Le notifiche possono apparire brevemente in una finestra mobile denominata notifica heads-up. Questo comportamento è in genere per le notifiche importanti che l'utente dovrebbe conoscere immediatamente e appare solo se il dispositivo è sbloccato.

**Le notifiche in un app android sono brevi messaggi per l’utente visualizzati in vari modi. Si elenchino le tipologie di notifiche conosciute, possibilmente con dei frammenti di codice che mostrano il loro utilizzo.**

* ***Toast***: *Toast.makeText(getApplicationContext(), "Toast!", Toast.LENGTH\_LONG).show();*
* ***Dialog***:

*DialogInterface.OnClickListener dialogClickListener = new DialogInterface.OnClickListener() {*

*public void onClick(DialogInterface dialog, int which) {*

*switch (which) {*

*case DialogInterface.BUTTON\_POSITIVE:*

*// to do*

*case DialogInterface.BUTTON\_NEGATIVE:*

*//to do*

*}*

*}*

*};*

*AlertDialog.Builder builder = new AlertDialog.Builder(this);*

*builder.setMessage("Stai per ripartire da capo. Sei sicuro?")*

*.setPositiveButton("Si", dialogClickListener)*

*.setNegativeButton("No", dialogClickListener).show();*

* ***Notifica***:

*Notification.Builder notificationBuilder = new Notification.Builder( getApplicationContext()).setTicker(“Esempio”)*

*NotificationManager mNotificationManager = (NotificationManager)getSystemService(Context.NOTIFICATION\_SERVICE);*

*mNotificationManager.notify(1,notificationBuilder.build());*

**Che cosa è un Toast customizzato? Si spieghi come implementare un Toast customizzato.**

Un toast è una piccola finestra che occupa una porzione dell’UI usata per fornire informazioni all’utente durante l’interazione con l’applicazione; spesso viene utilizzato per informare l’utente circa l’esito o il progresso di una computazione. In Android, un toast di default è bianco, semi-trasparente ed ha dimensioni e layout del testo ben definiti. Mentre viene mostrato un toast, l’activity corrente resta visibile ed interattiva: essi spariscono automaticamente dopo alcuni secondi.

Android dà la possibilità di creare un toast customizzato. Per fare ciò, dapprima bisogna fornire il layout del toast in un file XML. In seguito, programmaticamente:

- creiamo un’istanza di Toast;

- impostiamo vari attributi del toast (gravity, duration, …) e ne impostiamo il layout con il metodo setView(), a cui passiamo l’inflate di tale layout;

- lo mostriamo con il metodo show().

|  |
| --- |
| *public void showCustomToast(View v) {*  *Toast toast = new Toast(getApplicationContext());*  *toast.setGravity(Gravity.CENTER\_VERTICAL, 0, 0);*  *toast.setDuration(Toast.LENGTH\_LONG);*  *toast.setView(getLayoutInflater().inflate(R.layout.custom\_toast,null));*  *toast.show();*  *}* |