



Mobile Programming

Prof. De Prisco

Prova scritta del esempio 5

NOME: _____

COGNOME: _____

MATRICOLA: _____

Domande	Punti
1	/10
2	/10
3	/10
4	/10
5	/10
6	/10
7	/10
8	/10
9	/10
10	/10
TOTALE	/100

Per ognuno dei seguenti casi scrivere il corrispondente spezzone di codice XML da inserire nel file di layout.

1. Il testo "Ciao", non editabile, largo quanto il contenitore e alto quanto basta a visualizzare il testo con il testo allineato al centro.

2. Il testo "Android" editabile, largo e alto quanto basta a visualizzarlo, che si posiziona sulla destra del suo contenitore.

3. Un pulsante, largo 300dp e alto 60dp, con identificativo "pulsante3", con testo "Premi" allineato al centro del pulsante, ed avente come listener la funzione "pulsantePremuto()"

4. Una progressBar alta 10dp e larga quanto il suo contenitore con identificativo "pb1", inizialmente invisibile.

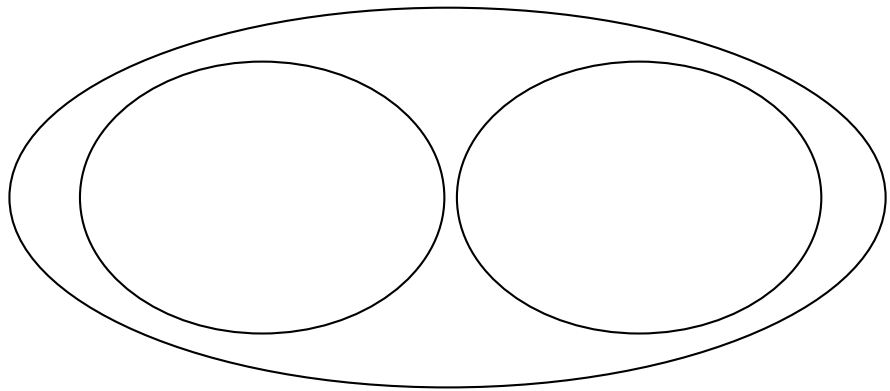
5. Un gridView largo 400dp e alto 500dp in cui gli elementi sono organizzati su 4 colonne

Android permette di specificare le dimensioni degli elementi dell'interfaccia grafica utilizzando varie unità di misura:

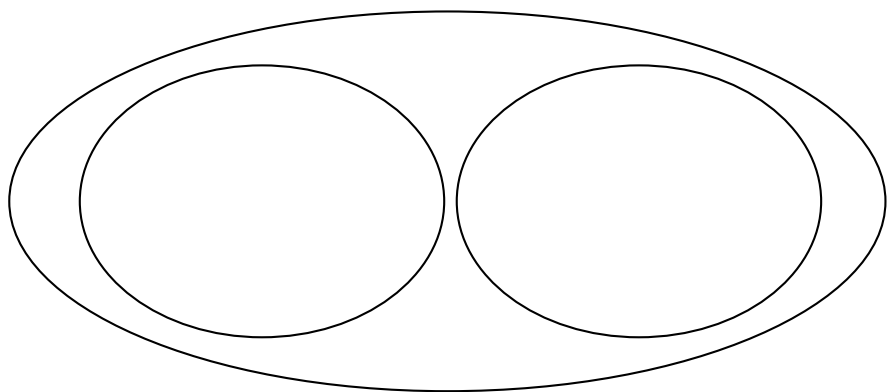
1. px, pixels
2. dp, density-independent pixels
3. sp, scale-independent pixels
4. in, pollici (inches)
5. mm, millimetri
6. pt, punti

Partiziona questo insieme di misure in due sottoinsiemi prima in funzione del fatto che la dimensione fisica sullo schermo è indipendente dalla densità del display e poi in funzione del fatto che la dimensione fisica sullo schermo dipende dalle preferenze dell'utente sulla grandezza dei font.

Partizionamento rispetto all'indipendenza dalla densità dello schermo:



Partizionamento rispetto all'indipendenza dalle preferenze dell'utente sui font:



Quesito 3

Prof. DE PRISCO

Disegna il ciclo di vita delle activity e spiega cosa succede a (1) un'activity, che è in esecuzione in foreground e (2) a un'activity che è in esecuzione in background, quando viene ruotato lo schermo.

Le due seguenti funzioni fanno parte del codice dell'ActivityA che lancia l'ActivityB per ottenere come risultato un valore booleano e un valore intero. Si completi il codice del metodo onActivityResult() in modo tale che alla fine dell'esecuzione dell'ActivityB il valore intero sia memorizzato nella variabile **numero** e il valore booleano nella variabile **flag**. L'activityB restituisce i valori come dati "extra" associati alle stringhe "VALORE_BOOLEANO" e "VALORE_INTERO". Si facciano gli opportuni controlli per assicurarsi che tutto sia andato per il verso giusto.

```
private boolean flag;  
private int numero = 0;
```

```
public void lanciaActivityB() {
    Intent i = new Intent();
    i.setClass(getApplicationContext(), ActivityB.class);
    startActivityForResult(i, 77);
}
```

```
@Override
```

```
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    if (requestCode != ...) return;
    if (
        != Activity.RESULT_OK) return;
}
```

}

Quesito 5

Prof. DE PRISCO

Si consideri la classe `AsyncTask<Integer, Integer, Bitmap>`. A cosa servono i tre parametri che in questo particolare esempio sono istanziati con i tipi `Integer`, `Integer`, `Bitmap`? Di quale metodi si deve fare l'override? Per ognuno di tali metodi si spieghi quali sono le operazioni solitamente svolte in esso.

La seguente classe `Pentagramma` implementa un widget customizzato che permette di inserire delle note in un pentagramma. Le note sono memorizzate con degli oggetti `Nota` che specificano la posizione nel pentagramma tramite delle coordinate cartesiane. Completare la classe `Pentagramma` per permettere di inserire nuove note e cancellare quelle esistenti (il costruttore, la funzione `onLayout` e `onDraw` già ci sono anche se non sono riportati i dettagli, quindi non serve aggiungere niente per queste funzioni). Cosa si deve fare, all'esterno di questa classe, dopo avere inserito o cancellato una nota affinché la modifica venga visualizzata?

Una nuova nota può essere inserita in una qualsiasi posizione dell'array `notesList`.

Per cancellare una nota occorre passare l'indice dell'array `notesList` in cui la nota è memorizzata.

```
public class Pentagramma extends View {
    ArrayList<Nota> notesList = new ArrayList<Nota>();

    public Pentagramma(Context c, int screen_w, int screen_h) {
        super(c);
        //Costruttore, inizializza l'oggetto
    }

    @Override
    protected void onLayout(boolean b, int x1, int y1, int x2, int y2) {
        //Questa funzione specifica le dimensioni
    }

    @Override
    protected void onDraw(Canvas canvas) {
        //Questa funzione disegna il pentagramma e le note in notesList
    }
}
```

Descrivi brevemente gli approcci che un'app può utilizzare per la memorizzazione di informazioni persistenti. Quale di questi metodi conviene utilizzare e in quali casi?

Quesito 8

Prof. DE PRISCO

Stai sviluppando un'app che permette di visualizzare una serie di scacchiere. Assumi di avere un oggetto `Tris`, che estende la classe `Fragment`, e che implementa tutte le funzionalità della singola scacchiera. Il numero di scacchiere da visualizzare, che indicheremo con `N`, lo specifica l'utente, quindi non lo conosciamo a priori, ma sappiamo che sarà al massimo 9. Supponendo di avere un file di layout che preveda 9 `FrameLayout`, i cui identificatori sono `frame1`, `frame2`, `frame3`, ..., `frame9`, scrivi uno spezzone di codice che permetta di inserire in modo programmatico `N` frammenti, ognuno dei quali è un oggetto `Tris`, in `N` contenitori `FrameLayout`, a partire dal primo (`frame1`).

Si spieghi come funziona il meccanismo del Multitouch, spiegando cosa sono i MotionEvent, i Pointer e gli Action_Codes.

Fai un esempio di `ContentProvider`, un esempio di `BroadcastReceiver` e un esempio di `Service`. Fornisci qualche dettaglio per ognuno degli esempi.

