



Android

תכנות דינאמי

אלון חימוביץ

צורך

- ▶ יצירת ה LAYOUT ב XML קבועה
- ▶ לעיתים צריך ליצור לוח גדול או תבנית קבועה בעיצוב המסך
- ▶ פעמים רבות אנו נדרשים לייצר מסך שכמות הרכיבים משתנה
- ▶ בהתאם לקלט של משתמש או לרמת משתמש
- ▶ ניתן ב JAVA לא רק לפנות לרכיבים קיימים אלא ליצור רכיבים חדשים
- ▶ רכיבים אלו יקושרו לLayout אב שמכיל אותם

גישה לרכיבים קיימים ועדכון בתכנה

- ▶ עד כה ניגשנו בתכנה לרכיבים קיימים ב XML
- ▶ הוגדרה הפנייה לרכיב קיים findViewById
- ▶ באמצעות ההפנייה ניתן לבצע פעולות על כל שדה שקיים ברכיב כגון
 - ▶ קריאה וכתיבה
 - ▶ שינוי מיקום (באמצעות חיישנים)
 - ▶ הצגה והסתרה של רכיב
 - ▶ שינוי גודל
 - ▶ שינוי צבע

גישה לרכיבים קיימים ועדכון בתכנה- דוגמה

הגדרת LinearLayout בXML

הפנייה לרכיב ב JAVA

שינוי ה Orientation

XML:

<LinearLayout

```
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/llDynamic"
    android:orientation="horizontal"
    tools:context=".DynamicProgramming"
>
```

JAVA:

```
LinearLayout llMainDynamic=findViewById(R.id.LlDynamic);
llMainDynamic.setOrientation(LinearLayout.VERTICAL);
```

יצירת Views ב JAVA

כאשר הרכיב מוגדר ב XML הוא הופך לאובייקט בתהליך ה `inflate` או `setContentView` ►

בעת יצירת רכיב ב Java נדרש זימון הבנאי של הרכיב ►

```
// add a text view
// create the element
TextView tvFirst = new TextView(this);
// add the text to be displayed
tvFirst.setText("First tv created");
```

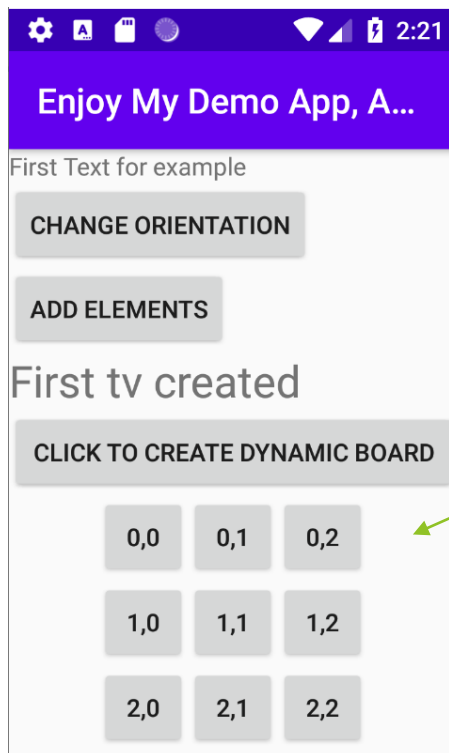
הגדרת ה `layout parameters` ►

`width,height` ►

```
// We add layout params= (height,width) to each view we create
LinearLayout.LayoutParams params=new LinearLayout.LayoutParams(LinearLayout.LayoutParams.MATCH_PARENT,
LinearLayout.LayoutParams.WRAP_CONTENT);

// set width and height with layout params created
tvFirst.setLayoutParams(params);
```

יצירת לוח תוך שימוש בתכנות דינאמי



- ▶ כאשר גודל הלוח מתקבל כקלט מהמשתמש
- ▶ בדוגמה שלנו נבנה לוח 3X3 של כפתורים השלבים:
- ▶ נדרש vertical linear layout
- ▶ מכיל שלושה horizontal linear layout
- ▶ לטובת עיצוב בהתאם למסך - נקבל מידות מסך של המכשיר

```
DisplayMetrics metrics = getResources().getDisplayMetrics();  
int width = metrics.widthPixels;
```

בניית ה vertical layout

► בדוגמה נחלק את הלוח באופן הבא $1/5$ רווח מימין, $3/5$ לוח משחק ו $1/5$ רווח משמאל

```
// we will create the board in the following way:  
// 1/5 margin 3/5 board 1/5 margin  
LinearLayout linearLayoutBoard = new LinearLayout(this);  
linearLayoutBoard.setOrientation(LinearLayout.VERTICAL);  
linearLayoutBoard.setLayoutParams(new LinearLayout.LayoutParams(  
LinearLayout.LayoutParams.MATCH_PARENT, LinearLayout.LayoutParams.WRAP_CONTENT));
```

► כל שורה ב Layout היא Horizontal Layout:

```
// define layoutparams for each row  
LinearLayout.LayoutParams rowLayout= new LinearLayout.LayoutParams  
(LinearLayout.LayoutParams.MATCH_PARENT, LinearLayout.LayoutParams.WRAP_CONTENT);  
rowLayout.setMargins(width/5,1,width/5,1);
```

בניית השורות

יצירת Layout Params לכל שורה ולכל רכיב בטבלה ►

נחשב רוחב כל כפתור בהתאם ►

```
// create a new row horizontal linear layout
// define lp for each element in the board
LinearLayout.LayoutParams elementLayout= new LinearLayout.LayoutParams
    (width/5, LinearLayout.LayoutParams.WRAP_CONTENT);
```

בכל שורה יוצרים רכיב חדש ►

בסיום בניית השורה מקשרים ל Layout שמייצג את הלוח ►

```
LinearLayout rowInBoard;
for(int i=0;i<rows;i++)
{
    rowInBoard = new LinearLayout(this);
    rowInBoard.setLayoutParams(rowLayout);
    rowInBoard.setOrientation(LinearLayout.HORIZONTAL);
    for (int j=0;j<cols;j++)
    {
        // create element in the row
        // add to row - next Slide
    }
    //add the row to the layout
    linearLayoutBoard.addView(rowInBoard);
}
llMainDynamic.addView(linearLayoutBoard);
}
```


בניית כל שורה - לולאה פנימית

- ▶ יצירת כפתור
- ▶ קביעת גודל הכפתור בהתאם
- ▶ עדכון ה Tag/ID בהתאם למיקום (שורה/עמודה)
- ▶ הוספת הכפתור לשורה

```
for (int j=0;j<cols;j++)  
{  
    Button b = new Button(this);  
    b.setLayoutParams(elementLayout);  
    b.setTag(i + "" + j);  
    b.setText(i + "," + j);  
    rowInBoard.addView(b);  
}
```

0,0	0,1	0,2
1,0	1,1	1,2
2,0	2,1	2,2