

Android Layouts

appSchool.co.il

יש לנו 6 סוגי layout

1. LinearLayout
2. RelativeLayout
3. TableLayout
4. FrameLayout

Examples for LinearLayout

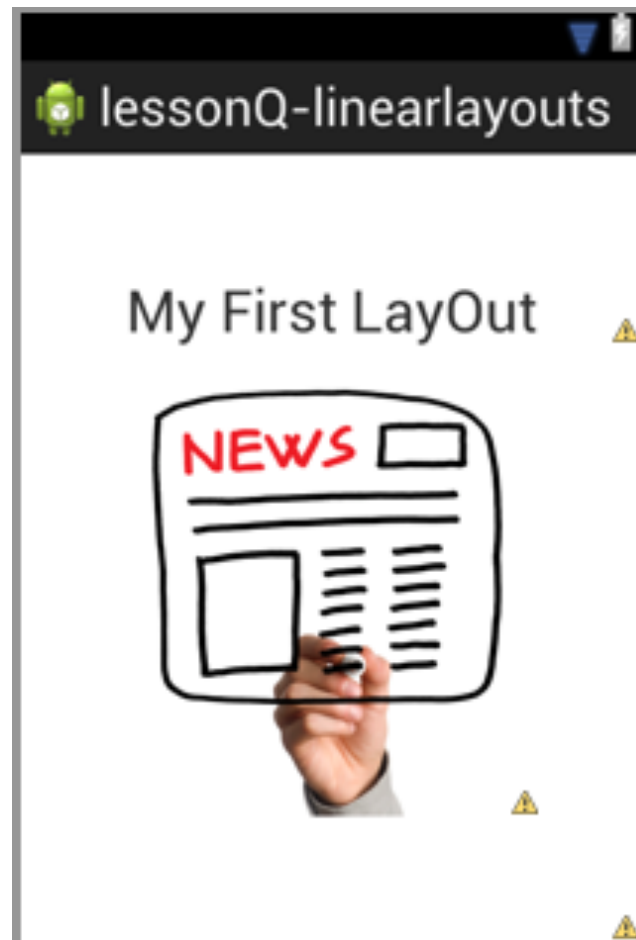
example 1

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:gravity="center">
    <TextView
        android:layout_width="fill_parent"
        android:id="@+id/TitleLabel"
        android:layout_height="wrap_content"
        android:text="My First LayOut"
        android:gravity="center_horizontal"
        android:textSize="30dp" />
    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/news"
        android:maxHeight="250dp"
        android:maxWidth="250dp"
        android:id="@+id/Photo" />
</LinearLayout>
```

הגדרת 1 layout דרך הקוד

```
public class MainActivity extends Activity {  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        //setContentView(R.layout.activity_main);  
  
        super.onCreate(savedInstanceState);  
        // setContentView(R.layout.main);  
        TextView label = new TextView(this);  
        label.setText("My first layout");  
        label.setTextSize(20);  
        label.setGravity(Gravity.CENTER_HORIZONTAL);  
        ImageView pic = new ImageView(this);  
        pic.setImageResource(R.drawable.news);  
        pic.setLayoutParams(new LayoutParams(LayoutParams.WRAP_CONTENT, LayoutParams.WRAP_CONTENT));  
        pic.setAdjustViewBounds(true);  
        pic.setScaleType(ScaleType.FIT_XY);  
        pic.setMaxHeight(250);  
        pic.setMaxWidth(250);  
        LinearLayout ll = new LinearLayout(this);  
        ll.setOrientation(LinearLayout.VERTICAL);  
        ll.setLayoutParams(new LayoutParams(LayoutParams.FILL_PARENT, LayoutParams.FILL_PARENT));  
        ll.setGravity(Gravity.CENTER);  
        ll.addView(label);  
        ll.addView(pic);  
        setContentView(ll);  
    }  
}
```

example 1



example 2

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent" android:layout_height="fill_parent"
    android:orientation="vertical">
    <TextView android:text="YELLOW" android:id="@+id/TV2"
        android:layout_height="wrap_content" android:layout_width="fill_parent"
        android:layout_weight=".2" android:background="#F2F202"
        android:gravity="center" android:textColor="#000"></TextView>
    <TextView android:text="GREEN" android:id="@+id/TV3"
        android:layout_height="wrap_content" android:layout_width="fill_parent"
        android:layout_weight=".2" android:background="#1FF235" android:gravity="center"
        android:textColor="#000"></TextView>
    <TextView android:text="BLUE" android:id="@+id/TV4"
        android:layout_height="wrap_content" android:layout_width="fill_parent"
        android:layout_weight=".2" android:background="#2D0FF2" android:gravity="center"
        android:textColor="#fff"></TextView>
        <TextView android:text="RED" android:id="@+id/TV5"
            android:layout_height="wrap_content" android:background="#F22707"
            android:layout_width="fill_parent" android:layout_weight=".1"
            android:gravity="center" android:textColor="#000"></TextView>

    <TextView android:text="VIOLET" android:id="@+id/TextView07"
        android:layout_height="wrap_content" android:layout_width="fill_parent"
        android:layout_weight=".1" android:background="#F252DD"
        android:gravity="center" android:textColor="#000"></TextView>
</LinearLayout>
```

הגדרת 2 layout דרך הקוד

```
public class MainActivity extends Activity {  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        //setContentView(R.layout.activity_main);  
        super.onCreate(savedInstanceState);  
        // setContentView(R.layout.main);  
        TextView tv1 = new TextView(this);  
        tv1.setText("HOW DO U FEEL?");  
        tv1.setTextSize(20);  
        tv1.setGravity(Gravity.CENTER);  
        TextView tv2 = new TextView(this);  
        tv2.setTextSize(20);  
        tv2.setGravity(Gravity.CENTER);  
        tv2.setText("FIND");
```

```
        TextView tv3 = new TextView(this);  
        tv3.setTextSize(20);  
        tv3.setGravity(Gravity.CENTER);  
        tv3.setText("LOVE YOU");  
        LinearLayout ll = new LinearLayout(this);  
        ll.setOrientation(LinearLayout.VERTICAL);  
        ll.setLayoutParams(new LayoutParams  
(LayoutParams.FILL_PARENT, LayoutParams.FILL_PARENT));  
        ll.setGravity(Gravity.CENTER);  
        ll.addView(tv1);  
        ll.addView(tv2);  
        ll.addView(tv3);  
        setContentView(ll);  
    }  
}
```


Example 2



example 3

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:orientation="vertical"
    android:weightSum="100" >
    <EditText
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:hint="to"
        android:layout_weight="7"
    />
```

```
<EditText
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:hint="subject"
    android:layout_weight="7"
/>
<EditText
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_weight="79"
    android:gravity="top"
    android:hint="message" />
<Button
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:layout_gravity="right"
    android:layout_weight="7"
    android:text="send" />
</LinearLayout>
```

example 3

The image shows a screenshot of an Android application window titled "lessonQ-linearlayouts". The app has a dark header bar with an Android icon and the title. Below the header, there are three text input fields stacked vertically, labeled "to", "subject", and "message". A blue rectangular frame with square handles at its corners and midpoints is drawn around the "subject" and "message" fields, indicating a linear layout constraint. At the bottom right of the form is a grey button labeled "send". The status bar at the top right shows a blue triangle and a battery icon.

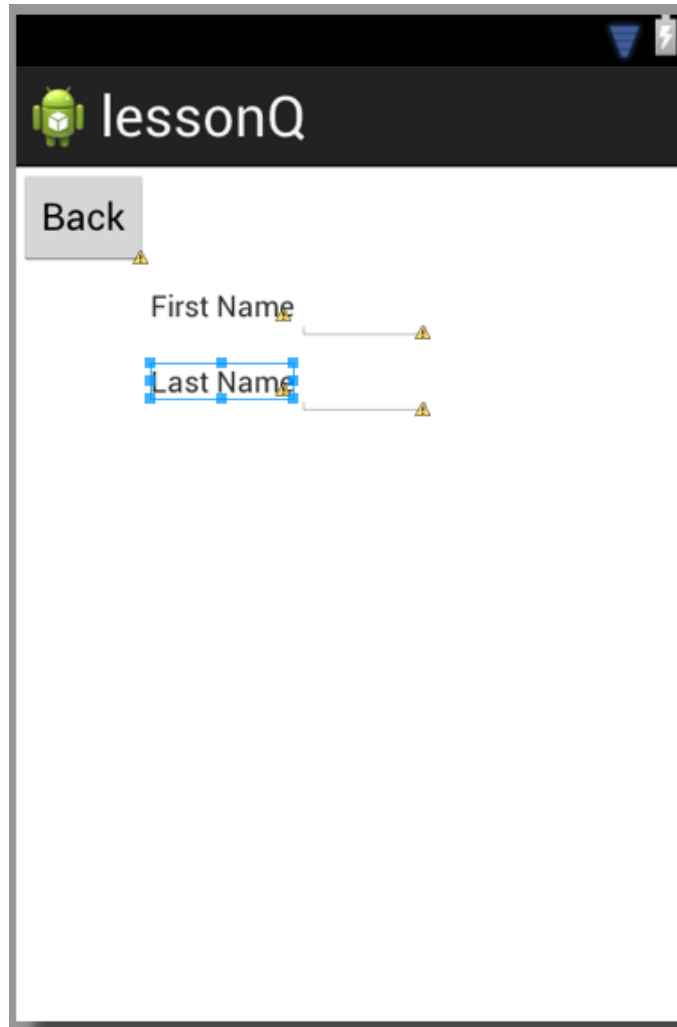
Table LayOut

TableLayout-example 1

```
<TableLayout
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    xmlns:android="http://schemas.android.com/apk/res/android">
    <TableRow>
        <Button
            android:id="@+id/backbutton"
            android:text="Back"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </TableRow>
    <TableRow>
        <TextView
            android:text="First Name"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_column="2" />
        <EditText
            android:width="100px"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </TableRow>
```

```
<TableRow>
    <TextView
        android:text="Last Name"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_column="2" />
    <EditText
        android:width="100px"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</TableRow>
</TableLayout>
```

example 1 - TableLayout



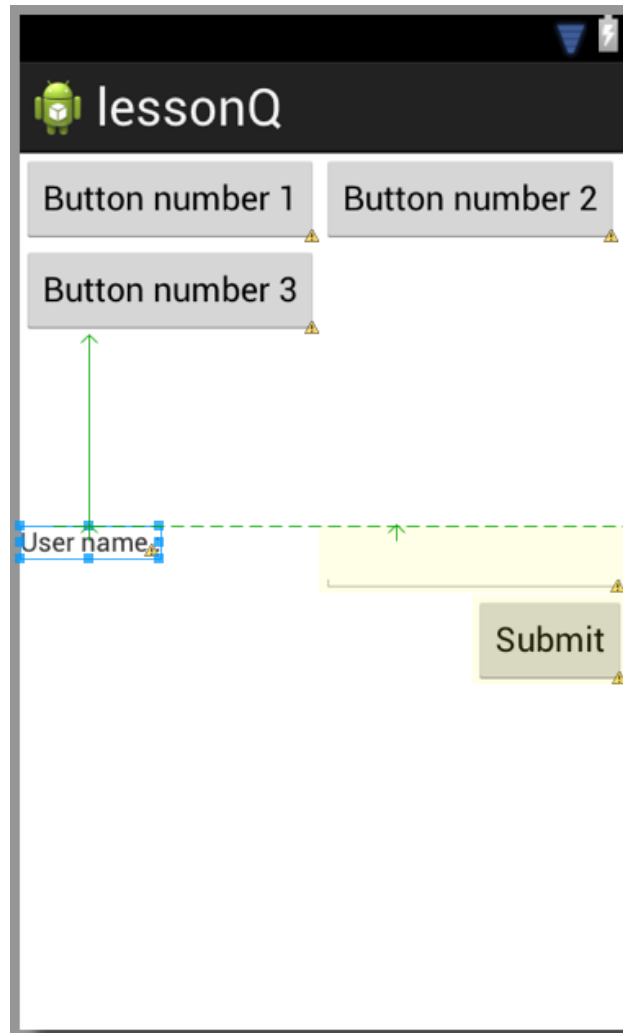
RelativeLayout

RelativeLayout-example 1

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.
com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >
    <Button
        android:id="@+id/btn1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Button number 1"/>
    <Button
        android:id="@+id/btn2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Button number 2"
        android:layout_toRightOf="@+id/btn1"/>
    <Button
        android:id="@+id/btn3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Button number 3"
        android:layout_below="@+id/btn1"/>
```

```
<TextView
    android:id="@+id/tv1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/btn3"
    android:layout_marginTop="100dp"
    android:text="User name : "
/>
<EditText
    android:id="@+id/et1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentRight="true"
    android:layout_alignTop="@+id/tv1"
    android:layout_toRightOf="@+id/btn3" />
<Button
    android:id="@+id/btnSubmit"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentRight="true"
    android:layout_below="@+id/et1"
    android:text="Submit" />
</RelativeLayout>
```


RelativeLayout example 1



FrameLayout

FrameLayout example 1

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">
    <ImageView
        android:id="@+id/img1"
        android:layout_height="300dp"
        android:layout_width="300dp"
        android:src="@drawable/news">
    </ImageView>
    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:textColor="#000"
        android:textSize="20dp"
        android:text="Title of picture" />
```

```
<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="info of picture"
    android:layout_gravity="bottom"
    android:gravity="right"
    android:textColor="#000"
    android:textSize="50sp" />
</FrameLayout>
```

Frame layout code example 1

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    super.onCreate(savedInstanceState);
    TextView tv1 = new TextView(this);
    tv1.setText("Title of picture");
    tv1.setTextSize(50);
    tv1.setTextColor(Color.BLACK);
    TextView tv2 = new TextView(this);
    tv2.setLayoutParams(new LayoutParams(LayoutParams.FILL_PARENT, LayoutParams.
WRAP_CONTENT, Gravity.BOTTOM));
    tv2.setTextSize(40);
    tv2.setGravity(Gravity.RIGHT);
    tv2.setText("info of picture");
    tv2.setTextColor(Color.WHITE);
    ImageView iv1 = new ImageView(this);
    iv1.setImageResource(R.drawable.news);
    iv1.setLayoutParams(new LayoutParams(LayoutParams.FILL_PARENT, LayoutParams.
FILL_PARENT));
    FrameLayout f = new FrameLayout(this);
    f.setLayoutParams(new LayoutParams(LayoutParams.FILL_PARENT, LayoutParams.
FILL_PARENT));
    f.addView(iv1);
    f.addView(tv1);
    f.addView(tv2);
    setContentView(f);
}
```

framelayout example 1



שיעורי בית

1. צור מסך LinearLayout הכולל טופס הרשמה.
 - a. firstname + תיבת text
 - b. lastname + תיבת text
 - c. address + תיבת text
 - d. email + תיבת text
 - e. כפתור submit לאישור.
 - f. הוסף תמונת רקע למסך.
2. בנה את אותו המסך באמצעות TableLayout
3. סדר 9 תמונות ב-3 שורות ו-3 עמודות בגודל 30*30 הוסף למסך כותרת הגלריה שלי.
בנה תרגיל זה באמצעות tablelayout ו-linearlayout
4. סדר 9 תמונות הכוללות - תמונה + textview המסביר על התמונה + textview המציג את שם מצלם התמונה. הוסף כותרת למסך.
5. צור מערך של 9 תמונות. מקם 3 כפתורים בראש המסך אחד ליד השני - לחיצה על
 - a. הכפתור הראשון תציג רק את השורה הראשונה של התמונות
 - b. לחיצה על הכפתור השני תציג 2 שורות של תמונות
 - c. לחיצה על הכפתור השלישי תציג 3 תמונות על המסך. .

יום ללא תכנות הוא יום מבוזבז!