

## Flutter: Android Studioda asosiy komponentalar bilan ishlash: Layout, Table, ListView, Grid, List va boshqalar

### Reja:

1. Flutter: Android Studioda asosiy komponentalar bilan ishlash
2. Layout, Table, ListView, Grid, List lar haqida umumiy ma'lumot

Palette panelida foydalanuvchi interfeysi komponentlari mavjud.

### *LinearLayout*

Layout da ikkita variant **mavjud: Horizontal** va **Vertical**.

LinearLayout tartibi barcha ob'ektlar bir yo'nalishda — vertikal yoki gorizontal ravishda moslashtiradi.

Yo'nalish **Android yo'nalishi xususiyati bilan belgilanadi: orientation:**

android:orientation="horizontal"

android:orientation="vertical"

### **Misol:**

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<LinearLayout
```

```
xmlns:android="http://schemas.android.com/apk/res/android"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent"
```

```
    android:orientation="vertical">
```

```
<LinearLayout
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent"
```

```
    android:layout_weight="1"
```

```
    android:orientation="horizontal">
```

```
<TextView
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="fill_parent"
```

```
    android:layout_weight="1"
```

```
    android:background="#aa0000"
    android:gravity="center_horizontal"
    android:text="red"/>
```

```
<TextView
```

```
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:background="#00aa00"
    android:gravity="center_horizontal"
    android:text="green"/>
```

```
<TextView
```

```
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:background="#0000aa"
    android:gravity="center_horizontal"
    android:text="blue"/>
```

```
<TextView
```

```
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:background="#aaaa00"
    android:gravity="center_horizontal"
    android:text="yellow"/>
```

```
</LinearLayout>
```

```
<LinearLayout
```

```
    android:layout_width="match_parent"
    android:layout_height="fill_parent"
    android:layout_weight="1"
```

```
        android:orientation="vertical">
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="row one"
            android:textSize="15dp"/>
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="row two"
            android:textSize="15dp"/>
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="row three"
            android:textSize="15dp"/>
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="row four"
            android:textSize="15dp"/>
    </LinearLayout>
</LinearLayout>
```

**Natija:**



Ushbu misolda TextView elementlarini gorizontal va vertikal ravishda joylashtirishning ikkita usuli qo'llaniladi.

LinearLayout markirovkasi android-ning qiziqarli xususiyatiga ega: layout\_weight, bu ichki elementning egallaydigan joyini bildiradi. Ushbu xususiyat taqdimotning "ahamiyati" ni belgilaydi va bu elementni qolgan bo'sh joyni to'ldirish uchun kengaytirishga imkon beradi. Standart holda 0 ga teng bo'ladi.

Misol uchun, agar uchta matn maydoni mavjud bo'lsa va ularning ikkitasi 1 qiymatida bo'lsa, ikkinchisiga hech qanday qiymat (0) berilmagan bo'lsa, u holda uchinchi matn maydoni kengaytirilmaydi va bu maydon tomonidan ko'rsatilgan matn hajmi bilan belgilanadigan maydonni egallaydi. Boshqa ikkitasi uchinchi maydon bilan band bo'lmagan bo'sh joyni to'ldirish uchun bir xil darajada kengayadi. Agar uchinchi maydon 2 (0 o'rniga) hajmda belgilansa, bu maydon boshqa ikki kishiga qaraganda "muhimroq" deb e'lon qilinadi, shuning uchun uchinchi maydon umumiy maydonning 50 foizini oladi, dastlabki ikkitasi umumiy maydonning 25 foizini oladi.

Bundan tashqari, Android xususiyatini ham belgilashingiz mumkin: weightSum. Agar atribut 100 qiymatini belgilasa, unda biz ichki matnlarning oladigan joy hajmini foiz sifatida qulay shaklda ko'rsatishimiz mumkin. Ushbu usul veb-ustalar tomonidan keng qo'llaniladi.

### Misol:

```
<?xml version="1.0" encoding="UTF-8"?><LinearLayout  
xmlns:android="http://schemas.android.com/apk/res/android"
```

```
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="#0000FF"
android:orientation="vertical"
android:weightSum="100">
```

```
<LinearLayout
```

```
    android:layout_width="fill_parent"
    android:layout_height="0dp"
    android:layout_weight="50"
    android:background="#FFFF00"
    android:gravity="center">
```

```
<TextView
```

```
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:text="Salom
    android:textColor="@color/black"
    android:textSize="45sp"
```

dunyo"

/>

```
</LinearLayout></LinearLayout>
```

**Natija:**



## TabLayout/TabItem

TableLayout, masalan, Sudoku, Tic-toe va boshqalar kabi mantiqiy o'yinlar yaratishda foydalanish uchun qulay.

TableLayout uchun ba'zi qoidalar mavjud. Birinchidan, har bir ustunning kengligi ustundagi eng keng tarkib bilan belgilanadi. Ichki elementlari atributlarda **match\_parent** **qiymatini** **ishlatadi**. Layout\_height uchun **TableRow xususiyati** har doim *wrap\_content*.

Agar **Android xususiyati**: "\*" " qiymatini belgilash uchun TableLayout komponentining stretchColumns bo'lsa, har bir TableRow komponentining tarkibi tartibning to'liq kengligiga cho'zilishi mumkin.

### Misol

#### *Ob-havo prognozi jadvalini yaratish*

Misol uchun, ob-havo prognozi jadvalini yaratish uchun quyidagi dizayndan foydalanish mumkin:

1. Birinchi qatorda jadval sarlavhasini ko'rsatish
2. Ikkinchi qatorda taqvim turiga ko'ra sanalar ko'rsatiladi
3. Uchinchi qatorda eng yuqori haroratni ko'rsating
4. To'rtinchi qatorda eng past haroratni ko'rsating
5. Beshinchi qatorda grafiklarni ko'rsatish (yomg'ir, qor, bulut, quyosh)



### ListView

#### **ListView-elementlarning aylanadigan ro'yxati**

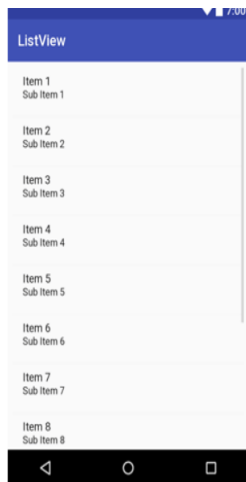
Biroq, agar XML kodini ko'rib chiqsak, u erda hech narsa ko'rilmaydi.

```
<ListView android:id="@+id/listView"
```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content" >
</ListView>

```



Mainactivity oynasiga o'tib quyidagilarni yozamiz

// kod onCreate usulida yozilgan()// ListView elementining namunasini oling

```

ListView listView = findViewById(R.id.listView);
// string qator aniqlashfinal
String[] cat Names = new String[]
{"Ryjik", "Barsik", "Murzik", "Murka", "Vaska","Thomasina",
"Kristina"," Qurol", "Smoke", "Kuzya","Kiti", "Masyanya", "Simba"};
// ma'lumot adapteridan foydalaningArrayAdapter<String> adapter = new
ArrayAdapter<>(this,android.R.layout.simple_list_item_1,
catNames);listView.setAdapter(adapter);

```

## GridLayout

Android 4.0-da GridLayout (**Asboblari panelidagi Layouts bo'limi**) nomi ostida yangi tartib paydo bo'ldi. Bir qarashda TableLayout kabi ko'rinishi mumkin. Lekin, aslida, bu juda ham qulay va funktsionaldir. Va yangi platforma uchun ishlab chiqilgan yangi loyihalarda o'rganish va undan foydalanish tavsiya etiladi.

Android kutubxonasiga **Android** klassi **qo'shildi**. **support.v7.widget.GridLayout**, bu komponentni eski qurilmalar uchun ishlatishga imkon beradi.

Belgilar Android sinfiga tegishli **widget.GridLayout** va **TableLayout** kabi ustunlar, qatorlar, hujayralar **mavjud**, ammo elementlar moslashuvchan tarzda sozlanishi mumkin.

Ustunlar va qatorlar soni **Android** atributlari tomonidan ishlatiladi: **columnCount="number"** va **android: rowCount="number"**.

**Misol:**

```
<?xml version="1.0" encoding="utf-8"?>
<GridLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:columnCount="4"
    android:orientation="horizontal" >
    <Button android:text="1" />
    <Button android:text="2" />
    <Button android:text="3" />
    <Button android:text="4" />
    <Button android:text="5" />
    <Button android:text="6" />
    <Button android:text="7" />
    <Button android:text="8" />
    <Button android:text="9" />
    <Button android:text="10" />
    <Button android:text="11" />
    <Button android:text="12" />
    <Button android:text="13" />
```

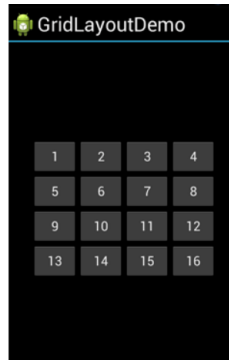


```
<Button android:text="14" />
```

```
<Button android:text="15" />
```

```
<Button android:text="16" />
```

**Natija:**



## GridView

GridView komponenti tekis jadval hisoblanadi. GridView uchun ArrayAdapter yoki BaseAdapter sinfidan olingan sinfni yaratish va uning getView usulini () qayta tuzish orqali ma'lumotlar elementlarini ko'rsatish uchun o'z maydonlarini ishlatish mumkin.

GridView uchun ustunlar soni tez-tez statik tarzda o'rnatiladi. Elementdagi qatorlar soni adapter taqdim etadigan elementlarning soniga qarab dinamik ravishda aniqlanadi.

### Xususiyatlari

**android** — numColumns-ustunlar sonini aniqlaydi. Agar qiymat *berilgan* bo'lsa auto\_fit, tizim mavjud maydonga asoslangan ustunlar sonini hisoblab chiqadi

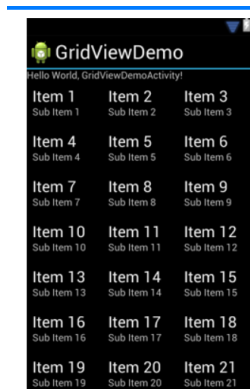
**android** — verticalSpacing-jadval hujayralari orasidagi bo'sh joy hajmini belgilaydi

**android:columnWidth**-ustunlar kengligini o'rnatadi

**android** — stretchMode-bo'sh joy qoldig'ini belgilangan Android qiymati bilan jadval uchun ajratilganligini ko'rsatadi: numColumns="auto\_fit". Ustunlar xujayralari orasidagi bo'sh joy

qoldig'ini kengaytirish uchun yoki *spacingWidth* — hujayralar orasidagi bo'shliqni kengaytirish uchun *columnWidth* qiymatlarini qabul qiladi

### Misol:



### Nazorat savollari

1. Flutter: Android Studioda asosiy komponentalari haqida ma'lumot bering?
2. Layout xususiyatlari haqida ma'lumot bering?
3. Table va ListView xususiyatlari haqida ma'lumot bering?
4. Grid va List lar haqida umumiy ma'lumot bering?