

Flutter: Android Studio interfeysi, ilk dasturni yozish va uni emulyatorda ishlatish, komponentalar bilan tanishish

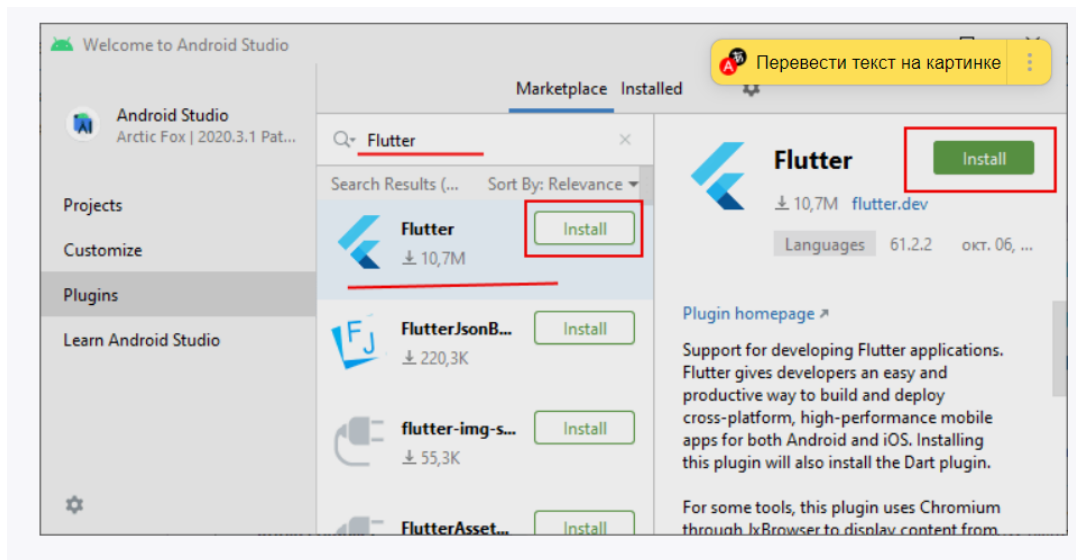
Reja:

1. Flutter: Android Studio interfeysi
2. Android Studio da ilk dasturni yozish va uni emulyatorda ishlatish
3. Android Studio komponentalari bilan tanishish

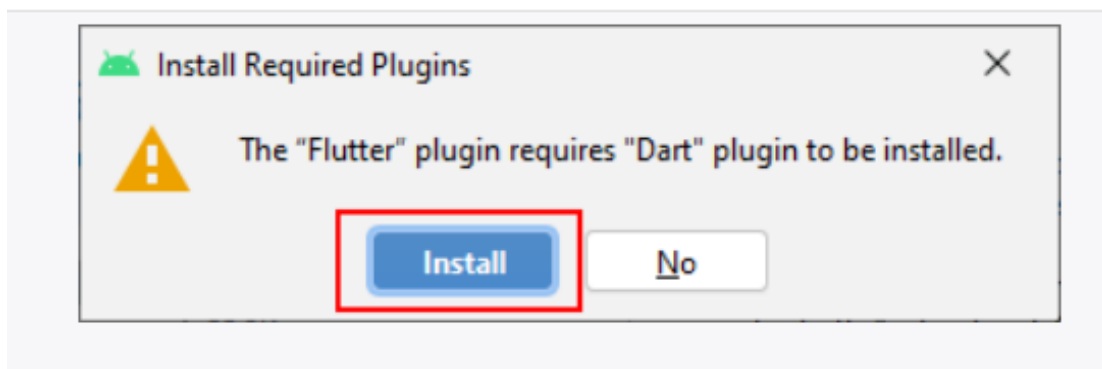
Flutter Android Studio da eng oddiy matn muharriridan kodni kiritishimiz va uni konsolda tuzish imkoniyati mavjud mumkin bo'lsa-da, lekin rivojlanish muhiti dasturni yozish va qurish jarayonini sezilarli darajada kamaytirishga imkon beradi. Bundan tashqari, Android Studio Flutter-da faqat Android uchun emas, balki boshqa qo'llab-quvvatlanadigan platformalar uchun ilovalarni yaratishga imkon beradi.

Android Studio bilan ishlash uchun, avvalo, uni o'rnatish kerak. O'rnatish havolasini yuklab olish kerak <https://developer.android.com/studio>.

Odatda Android Studio Flutterni qo'llab-quvvatlamaydi, shuning uchun tegishli plaginni o'rnatishimiz kerak. Buni amalga oshirish uchun boshlang'ich ekranda Android Studio-da Plugins-ni tanlang (yoki file - > Settings> menyusiga o'ting va keyin ochilgan oynada Plugins-ni tanlang). Va pluginlar panelida Flutter pluginini topamiz:

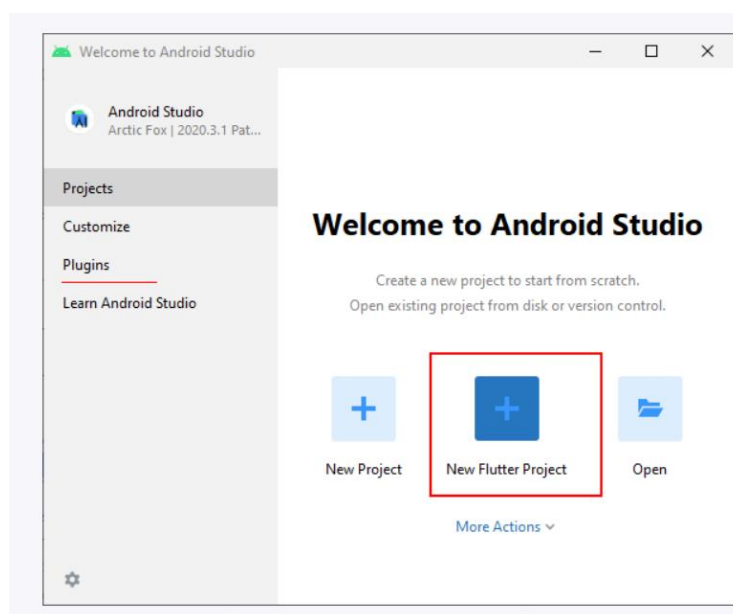


Kerakli plaginni qidirishni soddalashtirish uchun qidiruv tizimiga "Flutter" so'zini kiritishimiz mumkin va birinchi natija faqat o'rnatilishi kerak. Plaginni o'rnatishda dart uchun plaginni o'rnatish taklifi bilan oyna ham ko'rsatiladi. Bundan tashqari, uni o'rnatish uchun OK ni bosing:



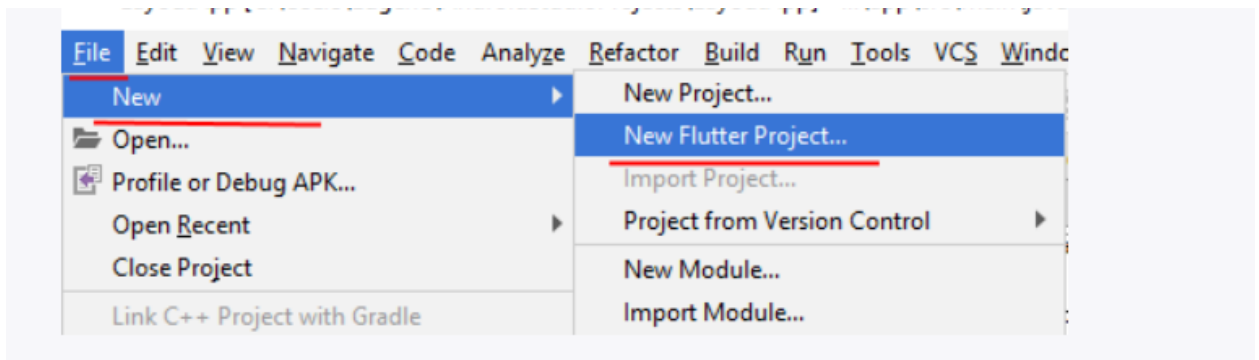
Plaginni o'rnatganingizdan so'ng Android studiyasini qayta ishga tushirishingiz kerak bo'ladi.

Android Studio-da boshlang'ich ekranda qayta ishga tushirilgandan so'ng, biz yangi Flutter loyihasi tugmachasini ko'rishimiz mumkin:

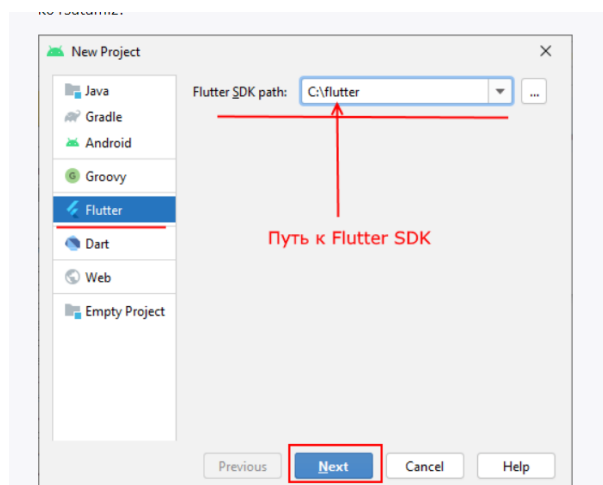


Flutter ostida loyiha yaratish uchun ushbu tugmani bosing.

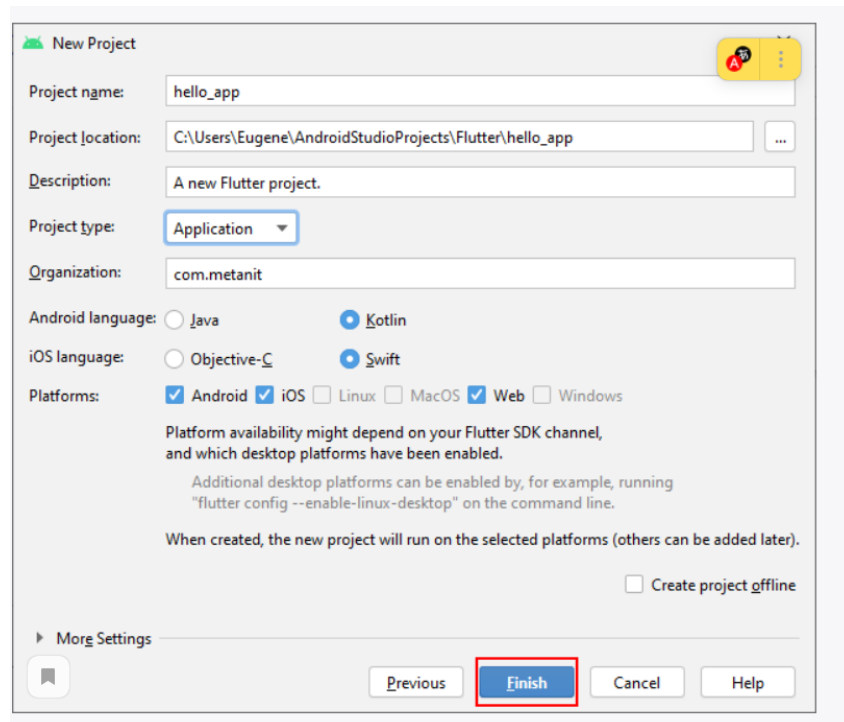
Shu bilan bir qatorda, studiyada loyiha yaratish uchun menyuga o'ting fayl - > yangi - > > yangi Flutter loyihasi > >:



Keyin biz yangi loyiha yaratish oynasini ochamiz. Chap tomonda **Flutter elementini tanlang**va flutter SDK yo'lidagi markazda **flutter SDK yo'lini** ko'rsatamiz:



Keyingi oynada bir qator loyiha parametrlarini ko'rsatamiz:



Loyiha nomi sohasida loyiha nomini beramiz. hello_app Agar taklif qilingan standart manzil mos kelmasa, loyiha joylashgan joyida loyiha manzilini o'zgartirish imkoniyati mavjud

Description maydonida loyihaning tavsifi ko'rsatilishi mumkin

Loyiha turi sohasida loyiha turi ko'rsatiladi. Odatiy bo'lib, u dastur qiymatiga ega (ya'ni, loyiha dasturni yaratish uchun mo'ljallangan). Ushbu standartni qoldiring.

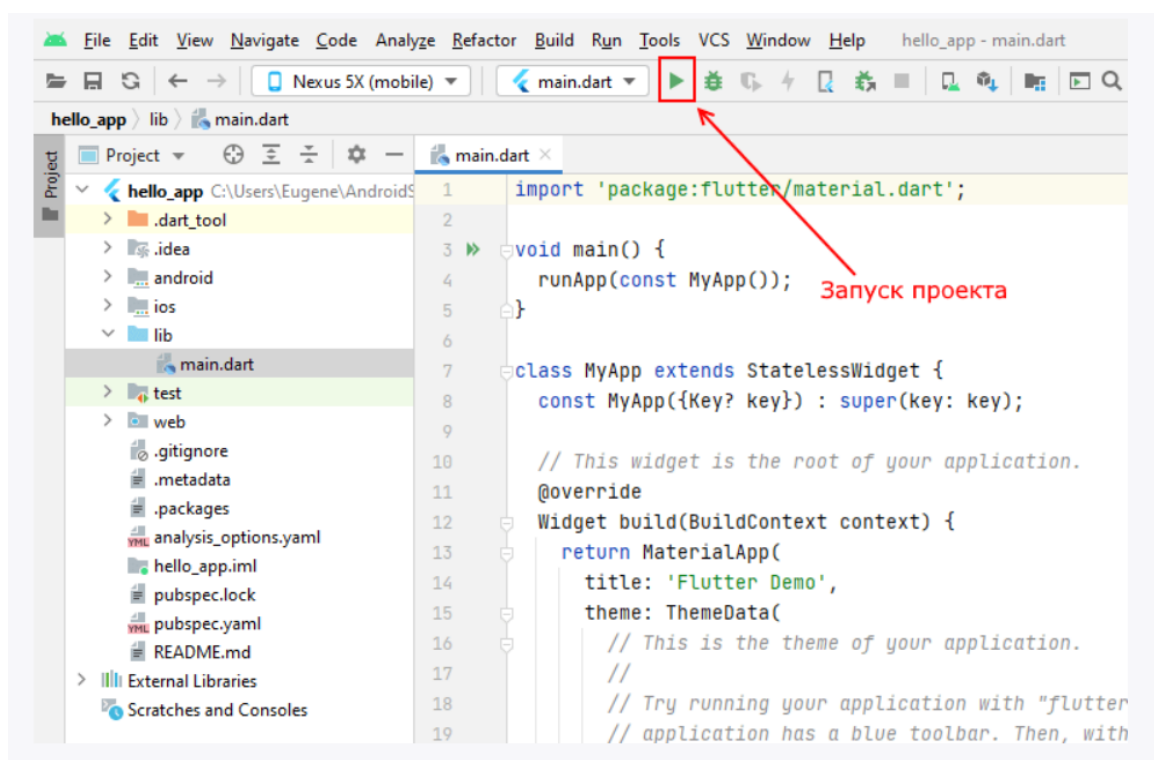
Android language maydoni Android uchun tilni ko'rsatadi. Asl qiymati qoldirish mumkin - Kotlin.

IOS language sohasida platforma iOS uchun til ko'rsatiladi. Asl qiymati qoldirish mumkin - Swift.

Platformalar sohasida siz loyihani yaratadigan platformalarni belgilashingiz mumkin.

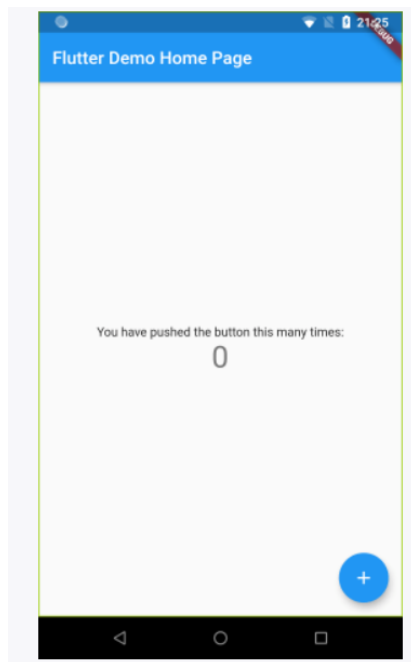
Odatda Android va iOS elementlari belgilanadi, ammo boshqa mavjud platformalarni tanlash mumkin.

Va keyin barcha sozlamalarni o'rnatganingizdan so'ng, loyihani to'g'ridan-to'g'ri yaratish uchun "tugatish" tugmasini bosing. Yaratilganidan so'ng, Android Studio yaratilgan loyihani ochadi:

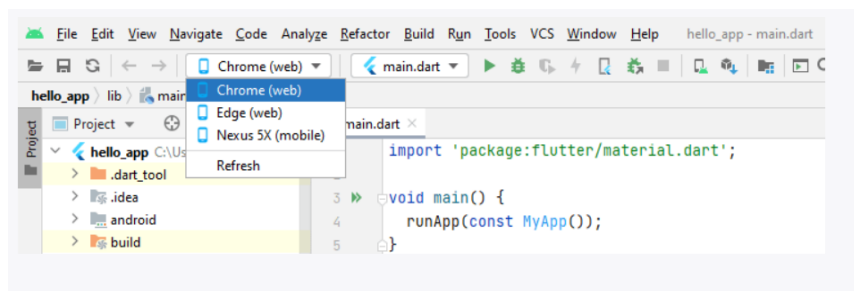


Yaratilgan loyiha jamoa yordamida konsolda o'tgan mavzuda yaratilgan bir xil tuzilishga ega bo'ladi flutter create. Studiya markazida asosiy fayl ochiladi.to'g'ri dastur kodini o'z ichiga olgan dart.

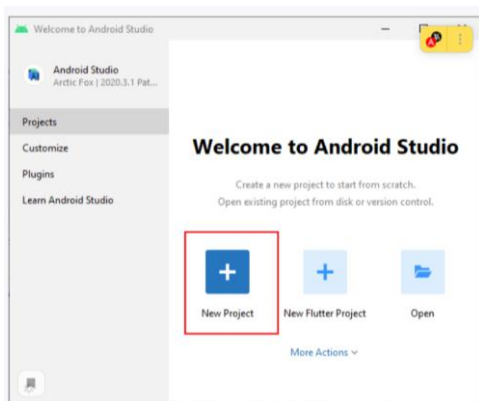
Android qurilmasini kompyuterga ulang (yoki emulyatorlardan foydalaning) va Android Studio panelida dasturni ishga tushirish uchun yashil tugmani bosing.



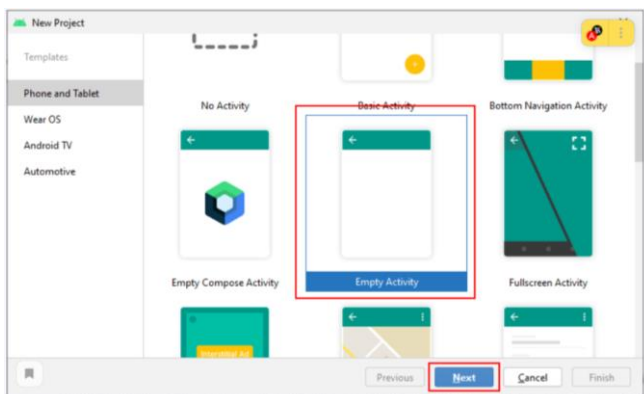
Xuddi shunday, Android studiyasida siz boshqa "qurilmalar" uchun, masalan, veb-sayt ostida loyihani ishga tushirishimiz mumkin. Buning uchun faqat asboblar panelidagi tegishli qurilmani tanlashingiz kerak:



Android operations tizimi uchun Android Studio muhitida birinchi ilovani yaratamiz. Android studiyasini oching va boshlang'ich ekranda **yangi loyihani tanlang**:

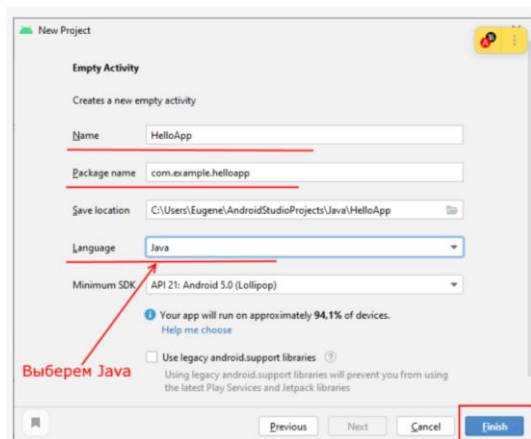


Android Studio loyihasini yaratishda dastlab loyiha shablonini tanlashni taklif qilamiz:



Android Studio turli xil vaziyatlar uchun bir qator shablonlarni taqdim etadi. Activity shablonini tanlang va keyingi tugmani bosing.

Shundan so'ng, yangi loyiha sozlamalari oynasi ko'rsatiladi:



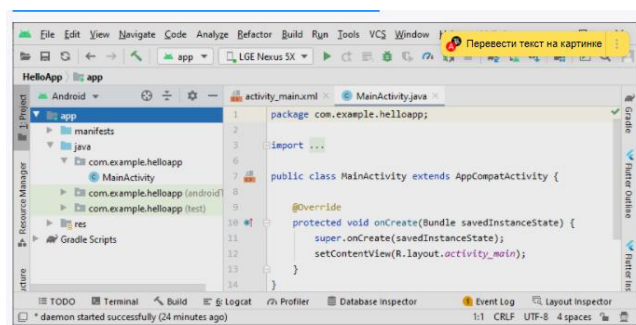
Yangi loyihani yaratish oynasida uning dastlabki sozlamalarini o'rnatishimiz mumkin:

- Ism maydonida ilovaning nomi kiritiladi. HelloApp nomini ko'rsatamiz

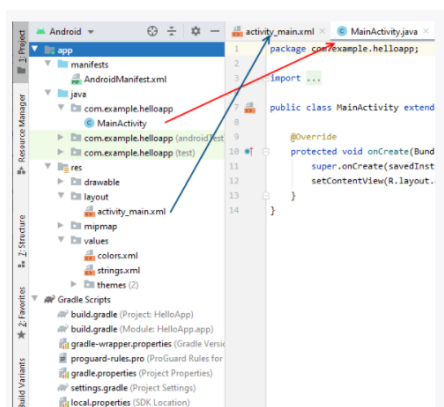
- Paket nomi maydonida ilovaning asosiy klassi joylashgan paketning nomi ko'rsatiladi. Bunday holda, test loyihalari uchun bu qiymat katta ahamiyatga ega emas, shuning uchun biz maqomni o'rnatamiz.example.helloapp.
- Save JoyXarita maydonida loyiha fayllarining joylashuvi qattiq diskda o'rnatiladi. Standart qiymatni qoldirish mumkin.
- Til sohasida dasturlash tili sifatida Java-ni ko'rsatamiz .
- Minimal SDK maydonida SDKNING eng kam qo'llab-quvvatlanadigan versiyasi ko'rsatiladi. Standart API 21: Android 5.0 (Lollipop) ni qoldiring, ya'ni bizning ilovamiz android 5.0 dan boshlab ishga tushirilishi mumkin, ya'ni qurilmalarning 94%. Qadimgi qurilmalarda ishga tushirish mumkin bo'lmaydi.

Shuni esda tutish kerakki, SDK versiyasi qanchalik baland bo'lsa, qo'llab-quvvatlanadigan qurilmalar soni kamroq.

Keyin Finish tugmasini bosamiz va Android Studio yangi loyihani yaratadi:



Birinchidan, loyihaning tuzilishini qisqacha ko'rib chiqamiz.



Android loyihasi turli modullardan iborat bo'lishi mumkin. Odatda, biz loyihani yaratganimizda, bitta modul-app yaratiladi. Modul uchta pastki papkaga ega:

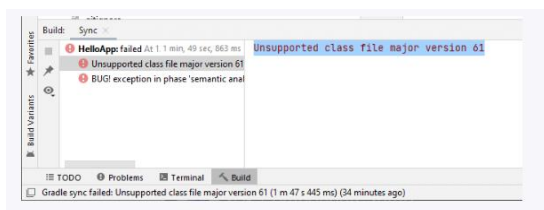
- manifestlar: `AndroidManifest` manifest faylini saqlaydi.xml, dastur konfiguratsiyasini tavsiflaydi va ushbu dasturning har bir tarkibiy qismini aniqlaydi.
- java: java-da kod fayllarini saqlaydi, ular alohida paketlar bo'yicha tuzilgan. Shunday qilib, papkada `com.example.helloapp`(uning nomi loyihani yaratish bosqichida ko'rsatilgan) dasturda `MainActivity` fayli mavjud. Java kodi Java tilida, ya'ni dastur boshlanganda dastur ishga tushirilgan `MainActivity` sinfini ifodalaydi
- res: ilovada ishlatiladigan resurslarni o'z ichiga oladi. Barcha resurslar pastki papkalarga bo'linadi.
 - chizilgan papka ilovada ishlatiladigan tasvirlarni saqlash uchun mo'ljallangan
 - layout papkani grafik interfeysni aniqlaydigan fayllarni saqlash uchun mo'ljallangan. Odatda, bu erda `activity_main` fayli mavjud.xml, asosiy sinf uchun interfeysni xml sifatida belgilaydi
 - `mipmap` papkalari turli xil ekran ruxsati bilan ilova belgilarini yaratish uchun mo'ljallangan rasm fayllarini o'z ichiga oladi.
 - `values` papkasida resurslar to'plamlarini o'z ichiga olgan turli xil xml fayllari saqlanadi-ilovada qo'llaniladigan turli ma'lumotlar. Odatda, bu erda ikkita fayl va bitta papka mavjud:
 - `res` / `drawable` xml ilovada ishlatiladigan ranglarning tavsifini saqlaydi, faylga yozish mumkin emas.
 - `res` / `string` xml ilovada ishlatiladigan string resurslarini o'z ichiga oladi, faylga yozish mumkin emas.
 - mavzular papkalari ikkita dastur mavzusini - yorug'lik (kunduzi)va qorong'i (tungi)

Gradle Scriptsning alohida elementi dasturni qurishda ishlatiladigan bir qator skriptlarni o'z ichiga oladi.

Ushbu tuzilishda MainActivity faylini tanlashimiz kerak bo'ladi. Android Studioda ochilgan va ilovaning mantig'ini o'z ichiga olgan java va aslida ilovaning bajarilishi boshlanadi.

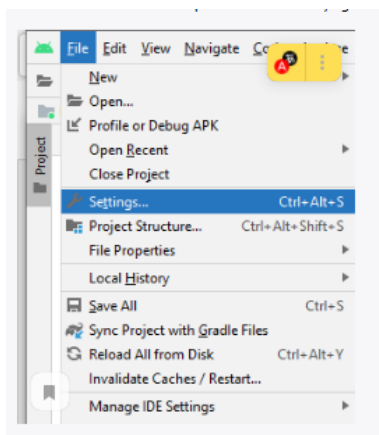
Paydo bo'lishi mumkin bo'lgan muammolar

Java dasturini yaratish uchun ishlatiladi. Va dasturni yaratish uchun Gradle infratuzilmasi qo'llaniladi. Biroq, Gradle ning joriy ishlatilgan versiyasi standart JDK versiyasiga mos kelmasligi mumkin. Va bu holda, Android Studio, masalan, unsupported class file xatosi major version 61 kabi xatolarni ko'rsatishi mumkin:



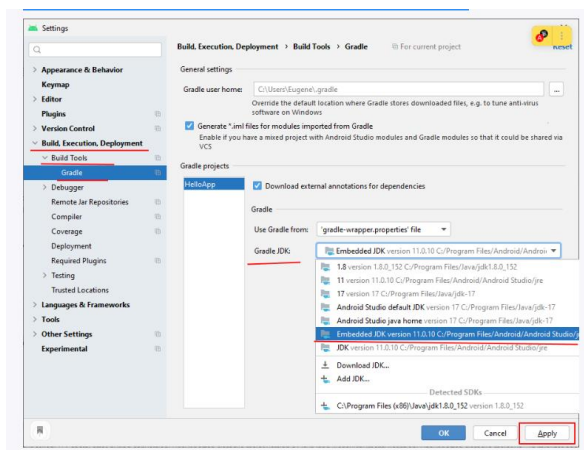
Ushbu xato JDK 17 versiyasi joriy Gradle versiyasiga mos kelmasligini aytadi. Va biz kichikroq versiyani ishlatishimiz kerak.

Ushbu muammoni hal qilish uchun studiyaga File ->Settings > (MacOS-da Android Studio -> Preferences>)



Keyin ochilgan Sozlamalar oynasida Menyu **elementiga o'ting Build, ijro etish, Deployment - > Build Tools - > > Gradle> >** va keyin JDK versiyasini o'zgartiradigan Gradle JDK maydonini toping. 11 va undan yuqori versiyalari bo'lishi kerak. Odatda, Android Studio bilan birga JDK-ning o'rnatilgan va

qo'llab-quvvatlanadigan versiyasi-hozirgi vaqtda jdk 11. Va u JDK ro'yxatida tanlanishi mumkin:



Android Studio bilan birga keladigan JDK versiyalarini tanlash uchun eng maqbul nuqta "embedded JDK version" deb ataladi.... Ekran tasvirida ko'rinib turganidek, bu versiya 11, lekin keyingi Android Studio yangilanishlari bilan ushbu versiya o'zgarishi mumkin.

O'zgarishlardan so'ng, birinchi OK tugmasini keyin Apply tugmasini bosamiz va loyihani ishga tushirishni takrorlang.

Loyihani ishga tushirish

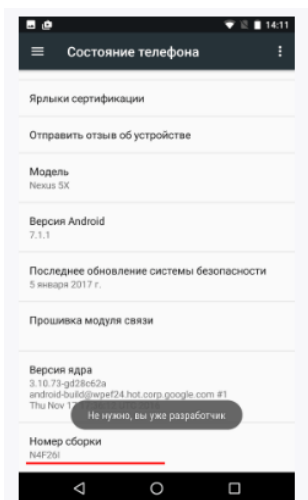
Yuqorida yaratilgan loyiha allaqachon funksionallikni o'z ichiga oladi. To'g'ri, bu funksionallik deyarli hech narsa qilmaydi, faqat " Salom dunyo!". Shunga qaramay, bu aslida biz boshlashimiz mumkin bo'lgan dastur.

Dasturni ishga tushirish va sinovdan o'tkazish uchun emulyatorlarni yoki haqiqiy qurilmalardan foydalanishimiz mumkin. Lekin ideal holda, haqiqiy qurilmalarda sinovdan o'tish yaxshiroqdir. Bundan tashqari, emulyatorlar katta apparat resurslarini talab qiladi va har bir kompyuter emulyatorlarning talablarini tortib ololmaydi. Sinov uchun mobil qurilmadan foydalanish uchun faqat kerakli drayverni o'rnatish kerak bo'lishi mumkin.

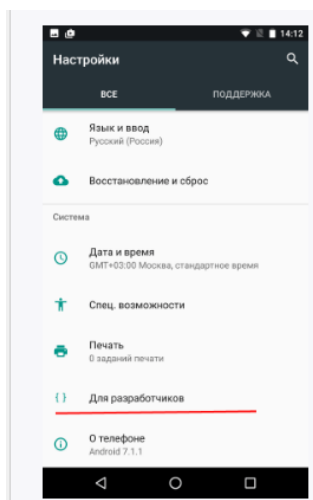
Telefonda ishlab chiquvchi rejimi

Odatda smartfonlardagi ishlab chiquvchi imkoniyatlari yashiringan. Bu imkoniyatni ochish quyidagi ketma-ketlik orqali amalga oshiriladi

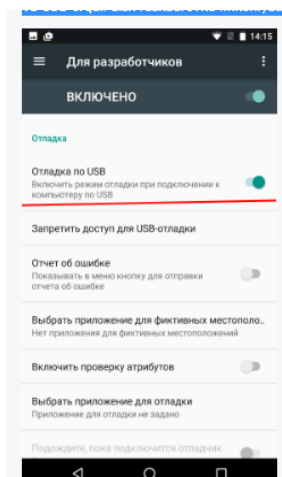
Sozlamalar > telefon Haqida (Settings > telefon Haqida) (Android 8 bo'ladi Sozlamalar > Tizim > telefon Haqida (Sozlamalar > Tizim > telefon Haqida)) va matbuot yetti marta bo'yicha Qurish Soni (qurish Soni).



Endi USB disk raskadrovka usulini yoqishingiz kerak. Buni amalga oshirish uchun Sozlamalar > System > Advanced > > > Developer options > > yoki Sozlamalar > SYSTEM > qo'shimcha> > > ishlab chiquvchilar uchun > > > (Android 8-da Sozlamalar > System > > Developer options > > yoki Sozlamalar> SYSTEM> ishlab chiquvchilar uchun>>).

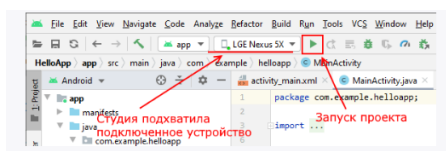


USB orqali disk raskadrovka imkoniyatini yoqing:

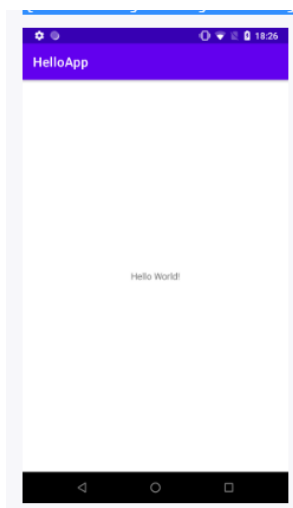


Иловани ishga tushirish

Qurilmani Android OS bilan ulang (agar biz haqiqiy qurilmada sinab ko'rsak) va asboblarni panelidagi yashil tugmani bosish orqali loyihani ishga tushiring.



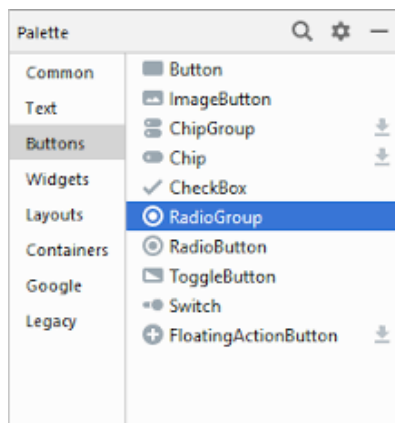
Qurilmani tanlang va OK tugmasini bosing. Va ishga tushirilgandan so'ng, biz ilovani qurilma ekranida quyidagicha ko'ramiz



RadioGroup

Ushbu sinf radio tugmalari to'plami uchun bir nechta istisno doirasini yaratish uchun ishlatiladi. Radio guruhiga tegishli bitta radio tugmachani tekshirish shu guruh ichidagi ilgari tekshirilgan radio tugmachasini tekshirmaydi.

Muayyan radio tugmachasini olib tashlashning iloji bo'lmasa-da, tekshirilgan holatni olib tashlash uchun radio guruhini tozalash mumkin.



RadioButton

Radio tugmasi - bu tekshiriladigan yoki belgilanmagan ikkita holat tugmachasi. Radio tugmasi belgilanmaganida, foydalanuvchi uni bosish yoki bosish orqali tekshirishi mumkin.



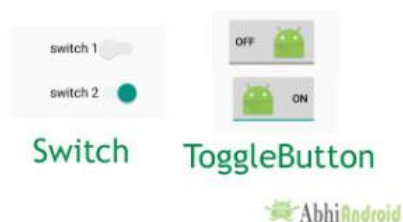
ToggleButton

toggleButton foydalanuvchiga ikkita holat orasidagi sozlamani o'zgartirishga imkon beradi.

[ToggleButton](#) Ob'ekt bilan tartibingizga asosiy o'tish tugmachasini qo'shishingiz mumkin . Slayderni boshqarishni ta'minlaydigan o'chirgich deb nomlangan boshqa bir tugma tugmachasini taqdim etadi va uni [Switch](#) ob'ekt bilan qo'shishingiz mumkin .

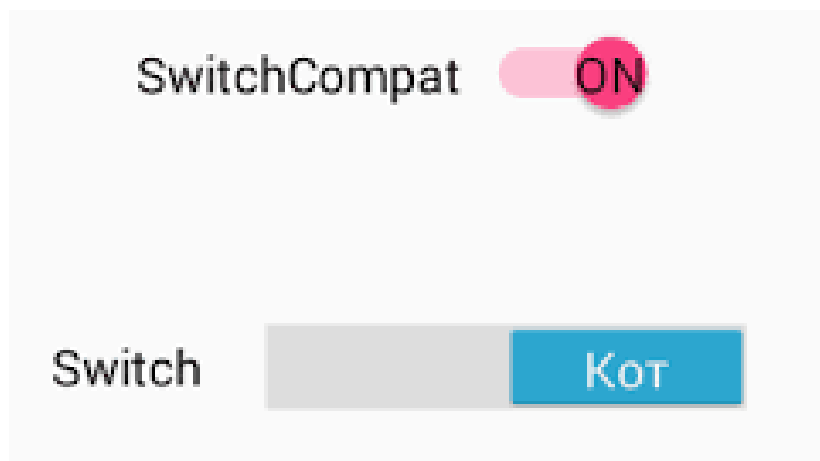
Agar tugmachaning holatini o'zgartirish kerak bo'lsa,

[CompoundButton.setChecked\(\)](#) yoki [CompoundButton.toggle\(\)](#) usulidan foydalanishingiz mumkin .



Switch

Switch - bu ikkita variantni tanlashi mumkin bo'lgan ikkita holatni o'zgartirish tugmachasi vidjeti. Tanlangan variantni tanlash uchun imkoniyatini beradi.



TextView-eng ko'p ishlatiladigan komponentlardan biri. Uning yordami bilan foydalanuvchi dasturda harakat qilish uchun qulayroqdir.

Matnni foydalanuvchiga ko'rsatadigan foydalanuvchi interfeysi elementi. *Foydalanuvchi tomonidan tahrirlanadigan matnni taqdim etish imkoniyatini beradi.*

Quyidagi kod namunasi matn ko'rinishining tarkibini o'zgartirish uchun XML tartibi va kodi bilan berilgan:

<LinearLayout

```
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
```

```

<TextView
    android:id="@android:layout_height="wrap_content"
        android:layout_width="wrap_content"
    android:text="@string/hello" />
</LinearLayout>

```

Код TextView в XML:

```

<TextView android:id="@+id/simpleTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="AbhiAndroid" />

```

id: id - bu matn ko'rinishini aniqlash uchun ishlatiladigan atribut. Quyida biz matn ko'rinishining identifikatorini o'rnatadigan namunaviy kod mavjudni keltirib o'tamiz:

```

<TextView
    android:id="@+id/simpleTextView"
        android:layout_width="wrap_content"
    android:layout_height="wrap_content"/>
    layout_width="wrap_content"
    layout_height="wrap_content"/>

```

gravitatsiya atributi - bu chapga, o'ngga, markazga, yuqoriga, pastga, center_vertical, center_horizontal va hk kabi matnlarning joylashuvini boshqarish uchun ishlatiladigan **ixtiyoriy atribut**.

Quyida tushuntirishga ega namuna kodi keltirilgan bo'lib, unda TextView matni uchun center_horizontal gravitatsiyani o'rnatilgan:

```

<TextView
    android:id="@+id/simpleTextView"
        android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="AbhiAndroid"
    android:textSize="20sp"

```

```
android:gravity="center_horizontal"/> <!--center horizontal gravity-->
```



text: Matn atributi matnni namoyish qilishda matnni o'rnatish uchun ishlatiladi.

Matnni xml va java sinfida o'rnatishimiz mumkin.

Quyida biz "AbhiAndroid" matnini matn ko'rinishida joylashtirgan tushuntirish bilan namuna kodini keltiramiz:

```
<TextView
```

```
    android:id="@+id/simpleTextView"
```

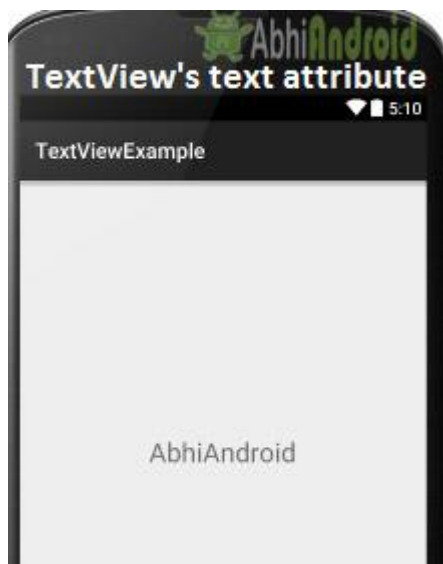
```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_centerInParent="true"
```

```
    android:textSize="25sp"
```

```
    android:text="AbhiAndroid"/>><!--отображать текст как AbhiAndroid-->
```

textColor: textColor atributi matn ko'rinishini matni rangini belgilash uchun ishlatiladi. Rang qiymati "#argb", "#rgb", "#rrggbb" yoki "#aarrggbb" shakllariga ega.

Quyida biz tushuntirilgan kod namunasi bo'lib, unda biz ko'rsatilgan matnning rangini qizil rangga o'rnatdik.

```
Android:параметр textSize="25sp"
```

```
Android:Андроид свойства textColor="#также F00"/><!--red color for text view-->
```

textSize: textSize atributi matn ko'rinishini matni hajmini belgilash uchun ishlatiladi. Matn hajmini sp (masshtabdagi mustaqil piksel) yoki dp (zichlik pikseli) ga o'rnatishimiz mumkin.

```
<TextView
```

```
android:id="@+id/simpleTextView"
```

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

```
android:layout_height="wrap_content"
```

```
android:text="AbhiAndroid"
```

```
android:layout_centerInParent="true"
```

```
android:textSize="40sp" /><!--Set size-->
```

textStyle: textStyle atributi matn ko'rinishini matn uslubini o'rnatish uchun ishlatiladi. Matnning mumkin bo'lgan uslublari qalin, kursiv va normaldir. Agar

biz matnli taqdimot uchun ikki yoki undan ortiq uslublardan foydalanishimiz kerak bo'lsa, unda buning uchun "|" operatoridan foydalaniladi.

background: fon atributi matn ko'rinishini fonini o'rnatish uchun ishlatiladi. Matn ko'rinishining foniga rang yoki naqsh o'rnatishimiz mumkin.

Edit text—elektron pochta xabarnomasida foydalanish uchun matnli shaklda taqdim etilgan ma'lumotlar formati.

Masalan. Bunda Ism, Familyangizni, Sharifingizni kiritish imkoniyati mavjud:

Password. Bu [autentifikatsiya](#) jarayonida foydalanuvchi identifikatorini tekshirish uchun ishlatiladigan belgilar qatori. Parollar odatda foydalanuvchi nomi bilan kon'yunkturada ishlatiladi; ushbu kod foydalanuvchiga qurilmaga, dasturga yoki veb-saytga kirishga ruxsat berish yaratiladi va ular faqat foydalanuvchiga ma'lum bo'ladi. Parol uzunligi bo'yicha har xil bo'lishi mumkin va harflar, raqamlar va maxsus [belgilarni](#) o'z ichiga olishi mumkin. Parol foydalanuvchi [shaxsiy identifikatsiya raqami](#).

E-mail. Elektron **pochta** (e-mail) — ma'lumotlarni uzatish tarmog'i orqali axborotlarni bir foydalanuvchi elektron qutisidan boshqasining jo'natish, qabul qilish va ma'lum vaqtgacha saqlanishini ta'minlovchi dasturiy texnik vositalar to'plami. E-Mail axborotlarni tarmoqning bir punktidan boshqasiga tezkor uzatishni ta'minlaydi.. Aniqroq aytganda, elektron pochta - bu ma'lum bir shaxsga yoki shaxslar guruhiga tarmoq orqali yuborilgan [matnlar](#) , [fayllar](#) , [rasmlar](#) yoki boshqa [qo'shimchalarni](#) o'z ichiga olgan xabar .

Number signed. Bunda faqat 0 dan 9 gacha raqamlarni kiritish mumkin.

Matn yozish imkoniyati mavjud emas.

Matematikada har qanday bazadagi manfiy sonlar ularni minus ("-") belgisi bilan old qo'shimchalar bilan ifodalanadi.

Biroq, kompyuter texnikasida raqamlar qo'shimcha belgilersiz, faqat bitlar ketma-ketligi sifatida ifodalanadi.

[AutocompleteTextView](#) .

Android-da AutoCompleteTextView - bu EditText-ga o'xshash ko'rinish, faqat foydalanuvchi matn kiritishi bilan avtomatik ravishda tugatish takliflari ro'yxatini

ko'rsatadi. Takliflar ro'yxati foydalanuvchi aslida Editbox tarkibini shu bilan almashtiradigan elementni tanlashi mumkin bo'lgan ochiladigan menyuda ko'rsatiladi.



CheckedTextView

Android-da CheckedTextView tekshiriladigan interfeysni ushlab turadigan va uni ko'rsatadigan muntazam TextView kengaytmasi. Unda bir nechta matn bilan birga tasdiqlash katakchasi mavjud. Bu asosan ListView-da ishlatiladi, bu erda biz qaysi element tanlangan yoki tanlanmaganligini ko'rsatishni xohlaymiz. checkmark atributu grafik yoki chizilgan CheckedTextView ob'ektini ta'minlash uchun ishlatiladi.

Nazorat savollari

1. Flutter: Android Studio interfeysi haqida ma'lumot bering?
2. Android Studio da ilk dasturni yozish va uni emulyatorda ishlatish qay tartibda amalga oshiriladi?
3. Android Studio komponentalari haqida ma'lumot bering?