

A.B.Xurramov

M.B.Jonqobilov

*e-mail: *azizbekxurramov0102@gmail.com*

Toshkent amaliy fanlar universiteti, Toshkent, O‘zbekiston.

D.I.Jo‘raboyeva

Kimyo xalqaro universiteti, Toshkent, O‘zbekiston

Annotatsiya: *Ushbu maqolada kotlin dasturlash tili imkoniyatlari haqida va uni ishlatadigan sohalar haqida ma’lumotlar ega bo’lish.*

Kalit so‘zlar: *Dastur, kotlin dasturlash tili, kotlindan foydalanish sabablari, kotlin hususiyatlari, smart cast, kotlin arxitekturasi.*

Dastur nima? Dastur oldindan belgilangan tartibda yozilgan ko’rsatmalar to’plami bo’lib, ular maxsus dasturlash tillarida yoziladi. Biz siz bilan endilikda ushbu maqolada dasturlash tillaridan biri bo’lgan kotlin dasturlash tili imkoniyatlarini o’rganamiz.

Kotlin Java Virtual Machine (JVM) da ishlaydigan ochiq manbali statik tarzda yozilgan dasturlash tilidir. U obyektga yo’naltirilgan dasturlashni (OOP) va funktsional dasturlashni cheklanmagan, o’zini o’zi ta’minlaydigan va o’ziga xos platformalarda birlashtiradi. Shuningdek, u miniatyura kodlari bilan funksiyalarni birlashtirishga imkon beradi. **Kotlin** - bu JetBrains tomonidan ishlab chiqilgan umumiy maqsadli dasturlash tili.[1]

Kotlin tarixi

- **2016-yil** - Kotlin v1.0 ishga tushirildi
- **2017-yil** - Googlening Androidda kotlinni birinchi darajali qo’llab-quvvatlashi haqida e’lon.
- **2018-yil** - Kotlin v1.2 JVM va JavaScript o’rtasida kodlarni tarqatish bo’yicha qo’shimchalar bilan birga keladi.
- **2019-yil** - Google Kotlinni Android ilovalarini ishlab chiquvchilar uchun afzal ko’rgan dasturlash tili sifatida e’lon qildi.
- **2021-yil** 20-sentabrda Kotlin v1.5.31 ishga tushirildi.

Kotlin nima uchun keng qo’llanilishining ba’zi muhim sabablari:

- Kotlin - bu o’qish va yozish juda oson bo’lgan statik tarzda yozilgan til.
- Kotlin dasturlari o’z dasturida nuqta, vergulni talab qilmaydi. Bu kodni oson va o’qishni osonlashtiradi.
- Ushbu til Javadan turli usullarda ma’lumot almashish va foydalanish imkonini beradi. Bundan tashqari, Java va Kotlin kodi bitta loyihada birga bo’lishi mumkin.

- Kotlin tipidagi tizim koddan `NullPointerException`ni yo'q qilishga qaratilgan.

- Kotlinida yangi kod yozish uchun sizga kamroq vaqt kerak bo'ladi. Kotlin kodini joylashtirish va uni miqyosda saqlash yanada oson.[2].

Kotlinning xususiyatlari

Bu yerda Kotlinning bir nechta xususiyatlari dasturlash tilining rivojlanish yo'liga ishonch hosil qiladi.

1.Kesilgan kodlash:

Kotlin – bu OOPga asoslangan dasturlash tili bo'lib, kod qatorini 40% gacha qisqartirish mumkin, bu esa Kotlinni dasturiy ta'minot yoki veb-ishlab chiqish uchun ideal tanlov qiladi.

2.Ochiq manba:

Android uchun Kotlin JVM dan foydalanadi va OOP va funktsional yo'naltirilgan dasturlash xususiyatlarini birlashtiradi.

3.Tez kompilatsiya:

Kotlin bilan ishlashda kodni kompilyatsiya qilish oson, bu esa androidni ishlab chiqishda yaxshiroq ishlashga olib keladi, shuningdek, butun kodda qaysi turdagi ma'lumotlar funksiyasidan foydalanish mumkinligini tushuntiradi.

4.Kengaytma funksiyalari:

Kotlin kodga hech qanday o'zgartirish kiritmasdan turli xil kengaytma funksiyalarini qo'llab-quvvatlashi mumkin. Bu mavjud kodni ishlab chiquvchilar uchun yanada jozibali va ajoyib qilishda yordam beradi.

5.Null xavfsizlik:

Kotlinda tizim quyida keltirilgan Kotlin misolida ko'rsatilganidek, null havolalarni emas, balki null havolalarni ajratadi.

Masalan, String o'zgaruvchisi nullni ushlab turolmaydi:

```
Var string1: String = "Bugun yakshanba"
```

```
String1 = null // kompilyatsiya xatosi
```

Nullga ruxsat berish uchun biz null o'zgaruvchiga qo'shilishimiz mumkin:

```
Var string2: String? = "Bugun yakshanba"
```

```
String2 = null
```

```
chop etish (string2)
```

6.Birgalikda ishlash mumkin:

Kotlin kodi Java tomonidan ishlatilishi mumkin, Java kodlari esa Kotlin bilan ishlatilishi mumkin. Shunday qilib, agar siz OOP dasturlash haqida ma'lumotga ega bo'lsangiz, Kotlinning rivojlanishiga o'tish oson.

Bundan tashqari, Javada yozilgan ba’zi ilovalar mavjud bo’lsa, ular Kotlin muhitida ishlatilishi mumkin.[4]

7.Kuchli asboblari:

Java singari, Kotlin kodi ham IDE yoki buyruq qatori interfeysi yordamida yozilishi mumkin. IDE bilan ishlash oson va sintaktik xatolar ham keskin kamayadi. Shu bilan birga, buyruq qatori interfeysi bilan ishlayotganingizda, avval kodni kompilyatsiya qilish kerak.

8.Smart Cast:

Smart casting asosan dastur xarajatlarini kamaytirishga yordam beradi va uning tezligi yoki ish faoliyatini yaxshilaydi. U turdagi casting yoki o’zgarma qiymatlar yordamida kod samaradorligini boshqarish ustida ishlaydi.

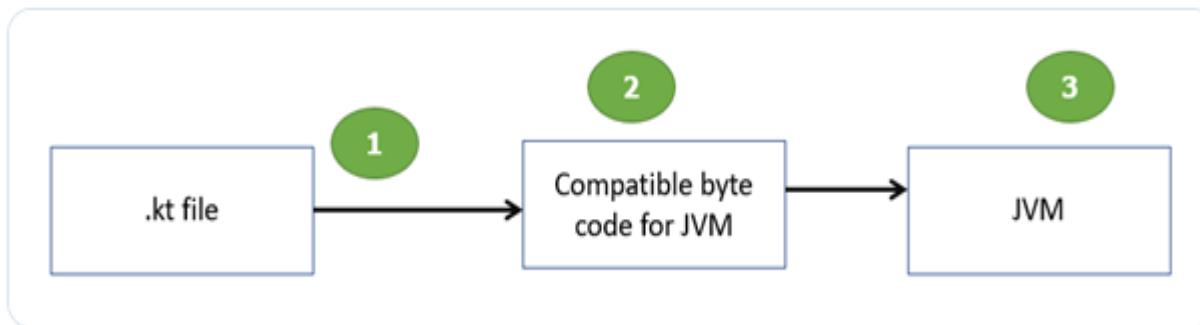
Masalan, agar satr aniqlansa, u holda uning uzunligi va soni va **Smart Cast** funksiyasi yordamida topish mumkin:

```
Val string1: Any = "Bugun dushanba"
qachon(string1)
{
    bu String -> string1.length
    Int -> string1.inc()
}
```

Kotlin arxitekturasi

Yaxshi qurilgan arxitektura dastur uchun o’z xususiyatlarini kengaytirish va oxirgi foydalanuvchi bazasining talablarini qondirish uchun muhimdir. Kotlin xotirani taqsimlash va ishlab chiquvchilar va oxirgi foydalanuvchilar uchun sifatli natijalarga erishish uchun o’ziga xos va o’ziga xos arxitekturaga ega.

Kotlining koroutinlari va sinflari yadroni kamroq kod ishlab chiqarish, unumdorlikni oshirish va samaradorlikni kuchaytirish uchun arxitektura qiladi. Kotlin kompilyatori har xil munosabatda bo’lishi mumkin bo’lgan turli xil ssenariylar mavjud, ayniqsa u har xil tillarni maqsad qilganda.



Kotlin arxitekturasi

Arxitektura diagrammasida kodning bajarilishi uchta oson bosqichda amalga oshirilishi aniq.

1. Birinchi bosqichda kompilyatorga ".kt" yoki kotlin fayli q o'shiladi.
2. Ikkinchi bosqichda Kotlin kompilyatori kodni bayt kodiga aylantiradi.
3. Uchinchi bosqichda bayt-kod **Java virtual mashinasiga** joylashtiriladi va JVM tomonidan bajariladi.

JVM-da bir necha bayt kodli fayl ishlaganda, ular o'zaro aloqani yoqib yuboradilar, shuning uchun Kotlindagi xususiyat Java tug'ilishi uchun o'zaro hamkorlik deb ataladi.[3]

Kotlinning JavaScriptga o'tishi Kotlin JavaScriptni nishonga olganida sodir bo'ladi. JavaScriptning maqsadi tanlanganda, Kotlin bilan ishlaydigan kutubxonaning bir qismi bo'lgan Kotlinning har qanday kodi JavaScript bilan to'kilganidan ko'ra ko'rinadi. Biroq, Java Development Kit (JDK) yoki foydalaniladigan Java kutubxonasi bundan mustasno. Bunday operatsiya davomida Kotlin bo'lmagan fayl e'tiborga olinmaydi. Maqsadli JavaScript .kt fayli JavaScript uchun izchil kod yaratish uchun Kotlin kompilyatori tomonidan ES5.1 ga aylantiriladi. Kotlin kompilyatori optimal o'lchamdagi chiqish, mavjud modul bilan o'zaro ishlash, standart kutubxona bilan bir xil funktsionallik va JavaScript o'qilishi mumkin bo'lgan chiqishga intiladi. Muhokamadan ko'rinib turibdiki, Kotlin kompilyatorlari yanada samaraliroq, malakali va mustaqil kod yaratishi mumkin, bu esa keyinchalik yuqori samarali dasturiy mahsulotga olib keladi.

FOYDALANILGAN ADABIYOTLAR:

1. Antonio Leiva. Kotlin for Android Developers. 2017-06-20.
2. Ted Hagos. Learn Android Studio 3 with Kotlin: Efficient Android App Development. Manila, National Capital Region, Philippine.2018.
3. *Shafirov*. „Kotlin on Android. Now official“ (17-may 2017-yil). — „Today, at the Google I/O keynote, the Android team announced first-class support for Kotlin.
4. „Stability of Kotlin Components“. *Kotlin* (21-may 2021-yil). Qaraldi: 29-iyul 2021-yil.