

Android қосымшаларын әзірлеуге Java мен Kotlin бағдарламалау

¹*ӘДІЛХАН Ахмет Беркүтұлы, магистрант, adilhan512133@gmail.com,

¹СМАГУЛОВА Асемгуль Сериковна, т.ғ.к., доцент, asemgul_sss@mail.ru,

¹Қарағанды техникалық университеті, Қазақстан, 100027, Қарағанды, Н. Назарбаев даңғылы, 56,

*автор-корреспондент.

Аңдатпа. Технологияның дамып келе жатқан әлемінде мобильді қосымшалар жаһандық мобильді нарықтың тез дамып келе жатқан сегменті болып табылады. Мақаланың мақсаты – Android қосымшасындағы Java мен Kotlin-ді салыстыру және қорытынды жасау. Мақалада Java-ның Android қосымшаларын бағдарламалау платформасы ретінде қалай қолданылатындығы сипатталған. Java-ның да, Kotlin-нің де әртүрлі ерекшеліктерін зерттеуге тырысады, соңында Android әзірлеуде қандай бағдарламалау тілі ыңғайлы екендігі туралы қорытынды жасалады.

Кілт сөздер: Android, Kotlin, Java, кеңейту функция, конструктор, нөлдік қауіпсіздік.

Кіріспе

Технологияның дамып келе жатқан әлемінде мобильді қосымшалар жаһандық мобильді нарықтың тез дамып келе жатқан сегменті болып табылады. Мобильді қосымшалар пайдаланушыларға бай және жылдам пайдаланушы тәжірибесін беру үшін жылдам қарқынмен дамуда. Бұл мақаланың мақсаты – Android қосымшасындағы Java мен Kotlin арасындағы салыстыру және қорытынды жасау. Мақалада Java-ның Android қосымшаларын бағдарламалау платформасы ретінде қалай қолданылатындығы сипатталған. Java-ның, Kotlin-нің әр түрлі мүмкіндіктері салыстырылған, соңында әзірлеушілерге қай бағдарламалау тілі ыңғайлы келетіні туралы қорытынды жасалынған.

2 Android операциялық жүйесі

Google Android-ті Linux платформасында жұмыс істейтін ұялы телефондарға арналған ашық операциялық жүйе ретінде іске қосқан. Бастапқыда Android Inc компаниясы жасаған, кейін Google компаниясы сатып алған.

Android түрлі нұсқаларында шығарылады. Android-тің әр нұсқасы десертке арнап аталынған. Android-тің бірінші аталған нұсқасы Cupcake деп аталады. Marshmallow, Nougat, Oreo, Lollipop, Kitkat, Jellybeans – бұл Android-тің басқа нұсқалары. Мәліметтерге сәйкес, 2021 жылдың ақпан айында Android-тің ең соңғы нұсқасы – Android 12 Snow Cone болып саналынады [7].

Android – 2011 жылдан бастап смартфондарда және 2013 жылдан бастап планшеттерде әлемдегі ең көп сатылатын ОЖ. 2017 жылдың мамыр айындағы жағдай бойынша оның айына екі мил-

лиардтан астам белсенді қолданушылары бар, ал 2021 жылдың қаңтар айындағы жағдай бойынша Google Play Store-да 3 миллионнан астам қосымшалар бар. Қазіргі Android-тың тұрақты нұсқасы – 2020 жылдың 8 қыркүйегінде шығарылған Android 11.

3 Java объектіге-бағытталған бағдарламалау тілі

Java – бұл Sun Microsystems алғаш рет 1995 жылы шығарған бағдарламалау тілі. Оны смартфоннан бастап мейнфреймге дейінгі әртүрлі құрылғыларда табуға болады. Біз оны жұмыс үстелінде, тіпті Raspberry Pi-де де қолдана аламыз.

Java тілі Android операциялық жүйесі үшін мобильді қосымшаларды құру үшін белсенді қолданылады. Бұл жағдайда бағдарламалар Dalvik виртуалды машинасында қолдану үшін стандартты емес байт-кодқа жинақталады. Android 5.0 бастап виртуалды машина ART-қа ауыстырылды. Мұндай компиляция үшін қосымша құрал қолданылады, атап айтқанда Google жасаған Android SDK. Қосымшаны әзірлеуді Android Studio, NetBeans, Eclipse бағдарламаларында Android Development Tools плагинінің көмегімен жасауға болады. 2014 жылдың 8 желтоқсанында Android Studio Google-мен Android ОЖ-ның ресми даму ортасы ретінде танылды [2].

4 Kotlin объектіге-бағытталған бағдарламалау тілі

Kotlin – Java Virtual Machine-де жұмыс істейтін және JetBrains жасаған объектіге бағытталған бағдарламалау тілі. Бағдарламалау тілі Кронштадт қаласы орналасқан Финляндия шыға-

нағындағы Котлин аралының атымен аталған. Оның негізгі әзірлеуін Ресейдің Санкт-Петербург қаласында орналасқан JetBrains бағдарламашылар тобы басқарады. Тіл 2010 жылдан бері дамып келеді және 2011 жылдың шілдесінде көпшілікке ұсынылды. Google I/O 2019-да Kotlin бағдарламалау тілі Android әзірлеуде басымдыққа ие болды деп жарияланды [5].

Авторлар Java-ға қарағанда анағұрлым қысқа және типті қауіпсіз және Scala-ға қарағанда қарапайым тіл құруды мақсат етті. Scala-мен салыстырғанда жеңілдетудің салдары IDE-де тезірек компиляция және тілді жақсырақ қолдау болды. Тіл Java-мен толығымен үйлесімді, бұл java әзірлеушілеріне оны біртіндеп қолдануға көшуге мүмкіндік береді [6].

5 Java мен Kotlin арасындағы салыстырмалы зерттеу

А. Кеңейту функциялар

Кеңейту функциялары функционалдылықты белгілі бір типтерге қосуға мүмкіндік береді. Бұл жағдайда типтерді басқа жерде, мысалы, стандартты кітапханада анықтауға болады.

Келтірілген мысалға сәйкес, бізге бірінші және соңғы таңбалар жойылған жаңа жолды қайтаратын String класының функциясы қажет; бұл әдіс String класында қолжетімді емес. Кластан тыс жарияланған кеңейту функциясы алдын ала анықталған функцияларды кеңейтетін көрсетілген кластың функционалдығын жасайды. Бұл функцияны келесідей кеңейтуге болады:

```
fun String.removeFirstLastChar(): String =
    this.substring(1, this.length - 1)
```

```
fun main(args: Array<String>) {
    val myString= "Hello Student"
    val res = myString.removeFirstLastChar()
    println("First string is: $res")
}
```

Kotlin бағдарламалау тілі кеңейту функциялары мен қасиеттерін қолдайды.

Ал Java бағдарламалау тілі кеңейту функцияларды қолдамайды. Әдетте кеңейту функцияларды қолдану үшін Android фреймворктері қолданылады. Бірақ Android фреймворктері кейде жағдайды қиындатады.

Ә. Тексерілген ерекшелік

Java жұмыс кезінде ерекше жағдайларды өңдеу үшін try ... catch блогын қолданады. Мұнда негізінен тексерілген ерекшеліктер қолданылады. Тексерілетін ерекшелік – бұл шақырылатын әдісте жарияланатын немесе ұсталыну керек ерекшелік түрі.[3]

Төменде Java try ... catch блогының синтаксисі келтірілген:

```
try
{ // код }
```

```
catch (e: SomeException)
{ // өңдеуші }
finally
{ // қосымша finally блогы }
```

Нөл немесе одан да көп catch блоктар болуы мүмкін, ал finally блок бір немесе мүлдем болмайды. catch блогы немесе finally блогы қажет.

Java-да, егер әдіс ішіндегі кез-келген код тексерілген ерекшелікті шақырса, онда әдіс не ерекше жағдайды өңдеуге, не пернетақтаның көмегімен throws ерекшелікті көрсетуге тиіс.

Kotlin-нің тексерілген ерекшеліктері жоқ. Kotlin-дегі барлық ерекше класстар – бұл Throwable класының ұрпақтары. Kotlin-де ерекшеліктерді өңдеу үшін throw өрнегін қолдануға болады.

```
fun fail(message: String): Nothing
{ throw IllegalArgumentException(message) }
```

```
val s = person.name ?: throw IllegalArgumentException(
    "Name required")
```

Тексерілген ерекшеліктер логиканы немесе код ағынын бұзуы мүмкін. Әсіресе, қоңырау шалудың көптеген әдістері бар кодтарда тексерілген ерекшеліктерді қолдану, код ағынын жоғалтуға әкелуі мүмкін.

Екіншіден, үлкен бағдарламалар болған жағдайда, тексерілген ерекшеліктер өнімділіктің төмендеуіне және кодтың жақсаруының шамалы немесе мүлдем болмауына әкеледі.

Б. Конструкторлар

Kotlin-де бастапқы конструктор да, екінші конструктор да болуы мүмкін. «constructor» кілт сөзі екінші конструкторды жариялау үшін қолданылады. Ол әрқашан бастапқы конструкторға сілтеме жасауы керек.

Екінші конструктордың мысалы:

```
class Student {
    val name: String
    val faculty: String
    val department: String
    private var elective = false
```

```
    constructor(name: String, faculty: String, department:
String) {
        this.name = name
        this.faculty = faculty
        this.department = marks
    }
```

```
    constructor(name: String, faculty: String, department:
String, elective:
Boolean) : this(name, faculty, department) {
        this.elective = elective
    }
```

Жоғарыда келтірілген мысалда бастапқы кон-

структурада 3 параметр, ал екінші конструкторда 4 параметр бар.

Java-да конструктор шамадан тыс жүктелген жағдайда код келесідей болады:

```
class Student {
    String name;
    String faculty;
    String department;
    Int elective;

    Student (String name, String faculty, String
department)
    {
        this.name = name;
        this. faculty = faculty;
        this. department = department;
        this.elective = 0;
    }

    Student (String name, String faculty, String
department, Int elective)
    {
        this.name = name;
        this. faculty = faculty;
        this. department = department;
        this.elective = elective;
    }
}
```

Екінші конструктор функциясы Java-да жоқ. Екінші конструктордың пайдалылығы – ол код жолдарын қысқартады.

Д. Нөлдік қауіпсіздік

Көптеген бағдарламалау тілдерінде, соның ішінде Java-да жиі кездесетін қателіктердің бірі – нөлдік сілтеме мүшесіне кіру нөлдік сілтемені ерекшелікке әкеледі. Java-да бұл NullPointerException немесе қысқартылған NPE-ге балама болар еді.[4] Kotlin нөлдік күйін өңдеу үшін нөлдік қауіпсіздік деп аталатын функцияны қолданады. Егер ол анық шақырылмаса, Kotlin NullPointerException ерекшелігін жасамайды.

Төменде Java коды көрсетілген:

```
Public static void main(String args[])
{
    String name= null;
    System.out.println(name.length());
}
```

Output: Null Pointer Exception

Төменде Kotlin-дегі код көрсетілген:

```
Fun main(args: Array<String>)
{
    var name: String? = null
    println(name?.length)
}
```

Output: Null

Kotlin-де кодтау кезінде Null Pointer ерекшелік Java-ға қарағанда код ағындарын бұзбайды. Ол NULL нәтижесін береді.

Е. Жалқау жүктеу

Жалқау жүктеу қажет емес кезде қосымша деректерді жүктеуден бас тартуды білдіреді. Оның орнына, деректер жүктелмегені және қажет болған жағдайда жүктеу керек екендігі туралы белгі қойылады. Осылай, жалқау жүктеу функциясы, жүктеу уақытын азайтады. Kotlin жалқау жүктеу функциясын қолдану мүмкіндік ұсынады.

Java-да жалқау жүктеу сияқты функция жоқ. Сондықтан қосымшаны іске қосу кезінде көптеген керек емес ақпараттар жүктеледі, бұл қосымшаның жұмысын баяулатады.

Қорытынды

Талдау кезінде біз Kotlin және Java-ның артықшылықтары мен кемшіліктері бар екенін білдік. Жаңадан бастаушылар тұрғысынан Java-әзірлеушілер арасында кеңінен қолданылатын өте танымал тіл, сондықтан Java – әзірлеуде тиімді тіл болып саналады.

Android-те әзірлеу – бұл бүкіл әлемде болып жатқан даму мұхитындағы тамшы ғана. Осылайша, бастауыш бағдарламашыларға, Java туралы білім Kotlin мүмкіндіктерін кеңейтуге қарағанда пайдалы.

Екіншіден, Java бағдарламашыларының үлкен қауымдастығы бар, яғни тығырыққа тірелгенде бағдарламалаудың маңызды сұрақтарына жауап табамыз. Бұл өте маңызды, өйткені жаңадан бастаушы ретінде біз әдеттегі сценарийде техникалық қиындықтарға кезігеміз және біз тіреліп қалған кезде қайда бару керектігін білмеуіміз мүмкін. Java проблемалары туралы іздегенде, біз міндетті түрде жауап аламыз.

Kotlin туралы олай айтуға қиын, ол әлі де дамып келе жатқан тіл.

Дегенімен, Kotlin келесі себептерге байланысты әзірлеушілерге тиімдірек болар еді:

1. Тіл мен қоршаған орта жетілген

Басқа бағдарламалау тілдерінен айырмашылығы, Kotlin шығарылымы 1.0 соңғы нұсқасын шығарғанға дейін көптеген кезеңдерден өтті. Kotlin басқа бағдарламалау тілдерінде туындайтын барлық ықтимал проблемалар эволюцияның соңғы кезеңіне дейін шешілгендігін білдіреді.

2. Kotlin Android әзірлеуін едәуір жеңілдетеді

Kotlin бағдарламалауды жеңілдетеді және Android қосымшаларын жақсартады. Kotlin – қазіргі заманғы бағдарламалау тілі. Бұл Android қосымшаларын жасаушыларға мүмкіндіктер терезесін ашады, бұл әзірлеушілерді өнімді етеді.

3. Kotlin сіздің кодыңыздағы қателерді азайтуға көмектеседі

Kotlin компиляторы қателерді табуды едәуір жеңілдетіп, мүмкіндігінше бірқалыпты жұмыс істеуге тырысады. Жұмыс уақытында қателіктерге жол бермеу және қателерді түзету үшін шығындар мен күштерді азайту үшін Kotlin компилято-

ры көптеген тексерулерді орындайды.

4. Kotlin коды қауіпсіздік

Kotlin көмегімен бағдарламалаудың жалпы қателіктерін оңай болдырмауға болады, нәтижесінде жүйенің ақауларын азайтады. Бұл Kotlin коды кез-келген басқа бағдарламалық кодқа қарағанда қауіпсіз екенін дәлелдейді.

5. Kotlin неғұрлым қысқа

Kotlin көптеген жағдайларда кез-келген басқа

бағдарламалау тілдеріне қарағанда қысқа; бұл бізге бірдей мәселелерді аз кодпен шешуге мүмкіндік береді. Бұл техникалық қызмет көрсету ыңғайлылығы мен кодты оқуды жақсартады, яғни инженерлер кодты тиімді жаза, оқи және өзгерте алады.

Android жобаны Kotlin-ге түрлендіруге қолдау көрсететіндіктен, әзірлеуші әрқашан еркін ауыса алады.

СПИСОК ЛИТЕРАТУРЫ

1. Альфред, В. Ахо Компиляторы. Принципы, технологии и инструментарий / Альфред В. Ахо и др. – М.: Вильямс, 2015. – 689 с.
2. Савитч, Уолтер Язык Java. Курс программирования / Уолтер Савитч. – М.: Вильямс, 2015. – 928 с.
3. Хабибуллин Ильдар. Самоучитель Java / Ильдар Хабибуллин. – М.: БХВ-Петербург, 2014. – 768 с.
4. Шилдт, Герберт Java 8. Руководство для начинающих / Герберт Шилдт. – М.: Вильямс, 2015. – 720 с.
5. Осипов, Дмитрий Delphi. Программирование для Windows, OS X, iOS и Android / Дмитрий Осипов. – М.: «БХВ-Петербург», 2014. – 464 с.
6. Майер Рето: Android 4. Программирование приложений для планшетных компьютеров и смартфонов. М.: Эксмо, 2013. – 816 с.
7. Смагулова А.С., Қасқатаева Г.Қ. Білім беру жүйесінде оқу үдерісін мониторингілеу үшін мобильді қосымшалар орны // Научно-технический журнал «Автоматика и информатика» №2 [43]. 2018. С. 80-84.

Программирование на Java и Kotlin в разработке приложений для Android

^{1*}**ӘДІЛХАН Ахмет Беркутұлы**, магистрант, adilhan512133@gmail.com,

¹**СМАГУЛОВА Асемгуль Сериковна**, к.т.н., доцент, asemgul_sss@mail.ru,

¹Карагандинский технический университет, Казахстан, 100027, Караганда, пр. Н. Назарбаева, 56,

*автор-корреспондент.

Аннотация. В растущем мире технологий мобильные приложения являются быстрорастущим сегментом глобального мобильного рынка. Цель статьи – сравнить Java и Kotlin в приложениях для Android и сделать выводы. В статье описано, как Java используется в качестве платформы для программирования приложений Android. Анализируются различные особенности как Java, так и Kotlin, и, наконец, делается вывод о том, какой язык программирования удобен в Android разработке.

Ключевые слова: Android, Kotlin, Java, функция расширения, конструктор, нулевая безопасность.

Java and Kotlin Programming in Android Application Development

^{1*}**ADILKHAN Akhmet**, master student, adilhan512133@gmail.com,

¹**SMAGULOVA Asemgul**, Cand. Tech. Sci., Associate Professor, asemgul_sss@mail.ru,

¹Karaganda Technical University, Kazakhstan, 100027, Karaganda, N. Nazarbayev Avenue, 56,

*corresponding author.

Abstract. In the growing world of technology, mobile apps are a fast-growing segment of the global mobile market. The purpose of the article is to compare Java and Kotlin in an Android app and draw conclusions. This article describes how Java is used as a platform for programming Android applications. Various features of both Java and Kotlin are analyzed, and finally, a conclusion is made about which programming language is convenient in Android development.

Keywords: Android, Kotlin, Java, extension function, constructor, zero security.

REFERENCES

1. Al'fred, V. Aho Kompilyatory. Principy, tekhnologii i instrumentarij / Al'fred V. Aho i dr. – Moscow: Vil'yams, 2015. – 689 p.
2. Savitch, Uolter Yazyk Java. Kurs programmirovaniya / Uolter Savitch. – Moscow: Vil'yams, 2015. – 928 p.
3. Habibullin Il'dar. Samouchitel' Java / Il'dar Habibullin. – Moscow: BHV-Peterburg, 2014. – 768 p.
4. SHildt, Gerbert Java 8. Rukovodstvo dlya nachinayushchih / Gerbert SHildt. – Moscow: Vil'yams, 2015. – 720 p.
5. Osipov, Dmitriy Delphi. Programmirovaniye dlya Windows, OS X, iOS i Android / Dmitriy Osipov. – Moscow: "BHV-Peterburg", 2014. – 464 p.
6. Majer Reto: Android 4. Programmirovaniye prilozhenij dlya planshetnyh komp'yuteroi i smartfonov. Moscow: Eksmo, 2013. – 816 p.
7. Smagulova A.S., Qasqataeva G.Q. Bilim beru jüiesinde oqu üderisin monitoringileu üşin mobildi qosymşalar orny // Nauchno-tehnicheskii jurnal "Avtomatika i informatika" no. 2 [43]. 2018. pp. 80-84.