Kotlin

Const and variable

var a = 6 -> can changing

Val b = 8 -> cannot changing

Data Types

String – Int - Float

Double - Char – Boolean

Ex: var a : Int = 10

Print

Print()

Println()

Read from User

val name = readLine()

Operations

*, / , + , -

+= , -= , *= , /=

++ , --

Conditions

If (condition happen) {

Do your program

}

else (condition not happen) {

Do your program

}

when likes switch

when(something) - >optional {

Case1 -> {

Do your program

}

Case2 {

Do your program

}

else -> default

{

Do your program

}

}

Loops

For (something in 1...9) -> useful when we know the limit will be stop

{

Do your program

}

While (something happens)

-> useful when we not know the limit will be stop

{

Do your program

}

Error handling

try {

```
Do your program

}

catch (e: Exception)

{

Write error message appear to user

}
```

Random number from 0-10

```
val num=Random.nextInt(11)
```

Functions

```
fun sayHi(){

      println("Hi")

   }

fun add(num1: Int): Int{

      return

   }
```

Lists and Arrays

```
x= listOf("Fluffy", "Snoopy")

pets = arrayOf("Fluffy", "Snoopy")
```

ArrayLists

```
val shoppingList = ArrayList()

shoppingList.add("Eggs")

shoppingList.remove("Milk")
```

2D ArrayLists

```
x = arrayListOf<ArrayList<String>>()
```

Dictionaries

```
mapOf(1 to "Sara", 2 to "Jim", 3 to "Jane")
```

OOP

```
class Person(val name: String){

  fun introduction(){

   println("Hi, my name is $name")

}

}

abstract class Vehicle{

      var color = "Blue"

      abstract fun doors()

   }

 class FamilyCar: Vehicle(){

     override fun doors() {

      println("This car has 4 doors")

      }

   }
```

Android Studio

Using the Console

```
Log.d("name of page","print Somthing")
```

Use UI object

```
lateinit var text : TextView

text = findViewById(R.id.textView)
```

Xml

```
android:backgroundTint=

"#F2F2F2"

android:ems="10"

android:hint="Name"

android:textColor="#F2F2F2"
```

android:inputType=

"textPersonName"

android:textColorHint=

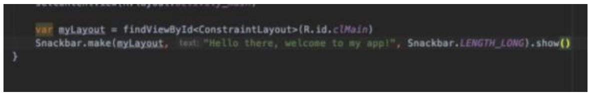
"#F2F2F2"

android:textSize="30dp"

android:textStyle="bold"

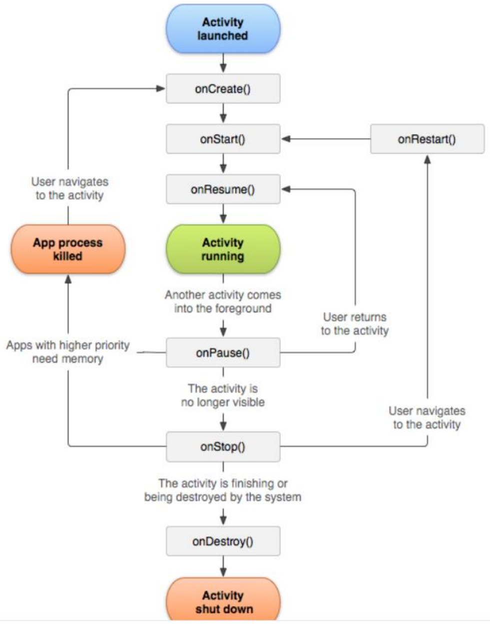
Snackbar

To display message for user



```
    var myLayout = findViewById<ConstraintLayout>(R.id.clMain)
    Snackbar.make(myLayout, text: "Hello there, welcome to my app!", Snackbar.LENGTH_LONG).show()
}
```

Activity Lifecycle


Activity launched
onCreate()
onStart()
onRestart()
User navigates to the activity
onResume()
App process killed
Activity running
Another activity comes into the foreground
User returns to the activity
Apps with higher priority need memory
onPause()
The activity is no longer visible
User navigates to the activity
onStop()
The activity is finishing or being destroyed by the system
onDestroy()
Activity shut down

We can access the remaining states of our activity by overriding the following methods; onStart(), onResume(), onPause(), onStop(), and onDestroy().