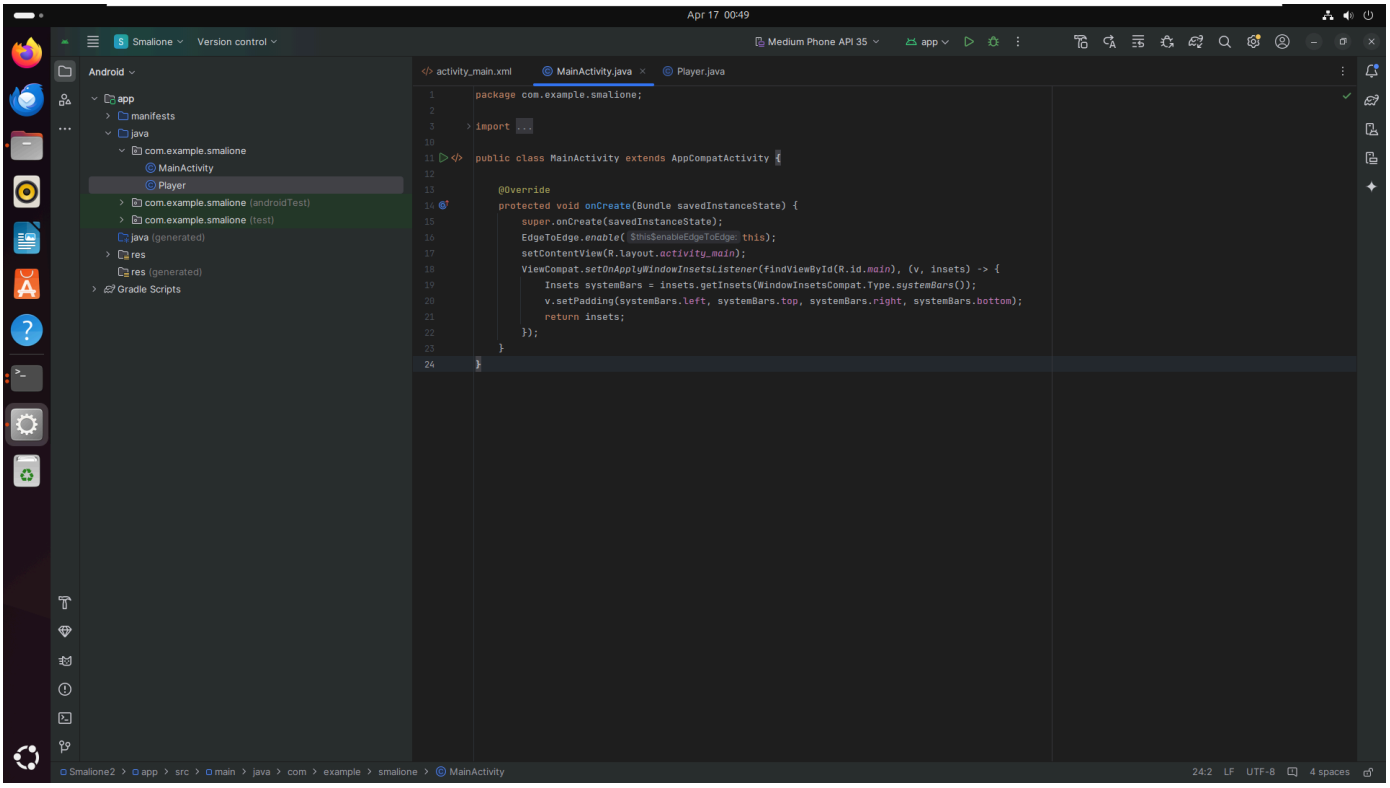
4- Smali

51-Introduction

هنا احنا هننشء apk باستخدام android studio وبعد ما ننشء apk هنفكه باستخدام apktool علشان نحوله smali code



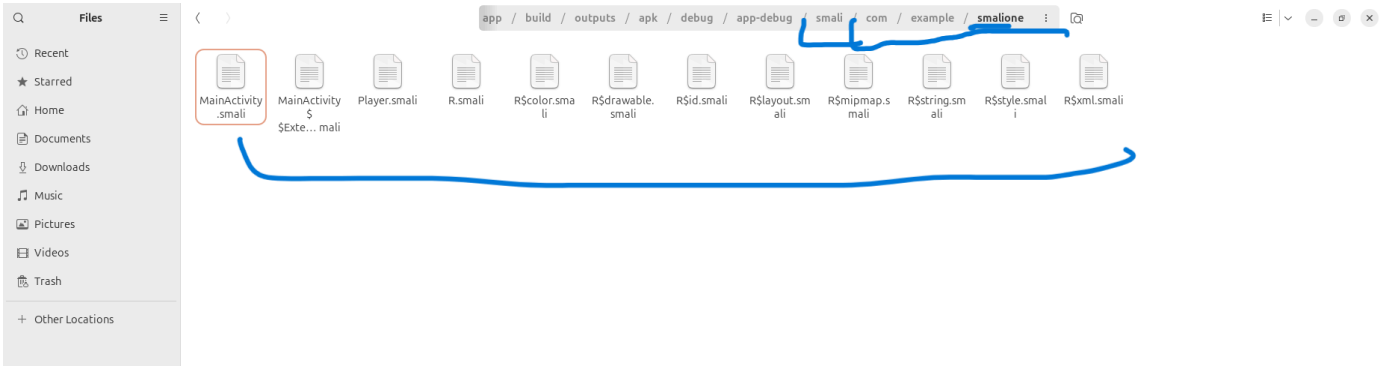

```
➜  debug apktool d app-debug.apk
I: Using Apktool 2.11.0 on app-debug.apk with 4 threads
I: Baksmaling classes.dex...
I: Loading resource table...
I: Decoding file-resources...
I: Loading resource table from file:
/home/ubuntu/.local/share/apktool/framework/1.apk
I: Decoding values */* XMLs...
I: Decoding AndroidManifest.xml with resources...
```

دلوقتي بعد ما فكنا هنلاقي كون folder اسمه smali و هنلاقي في folder ده بيتكون من 3 عناصر هما

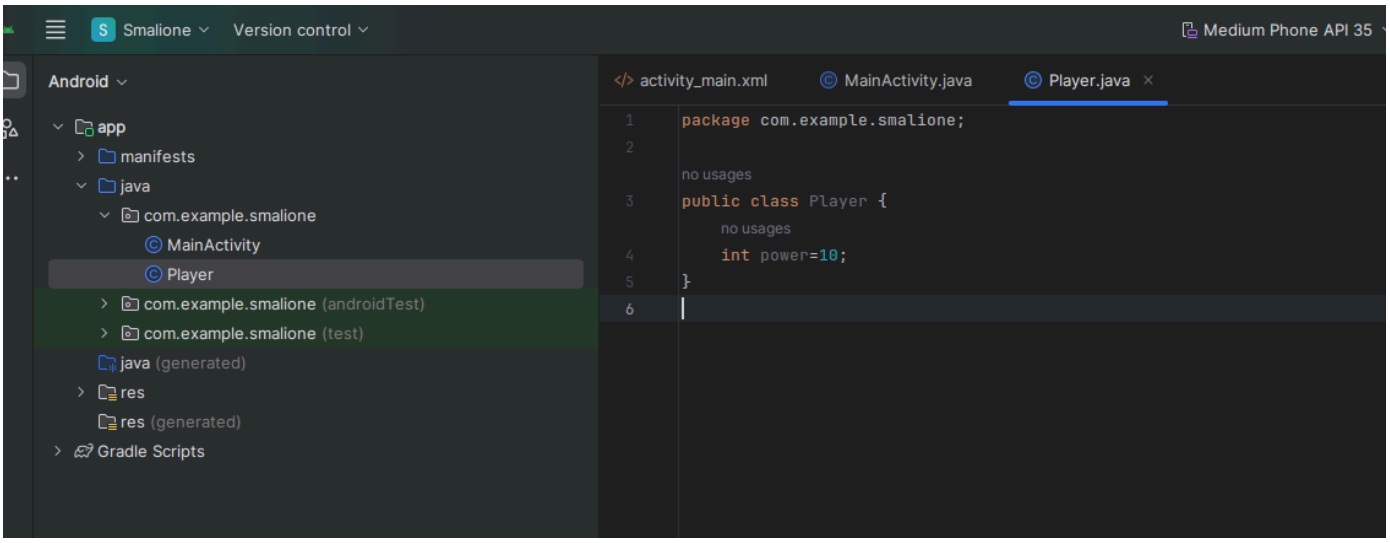
- android
- androidx
- com

خاصين ببنا التطبيق نفسه android,androidx ولكن اللي هنستخدمه هو com

هنا اه ملفات التطبيق نفسه مكتوبة بلغة smali تحت com

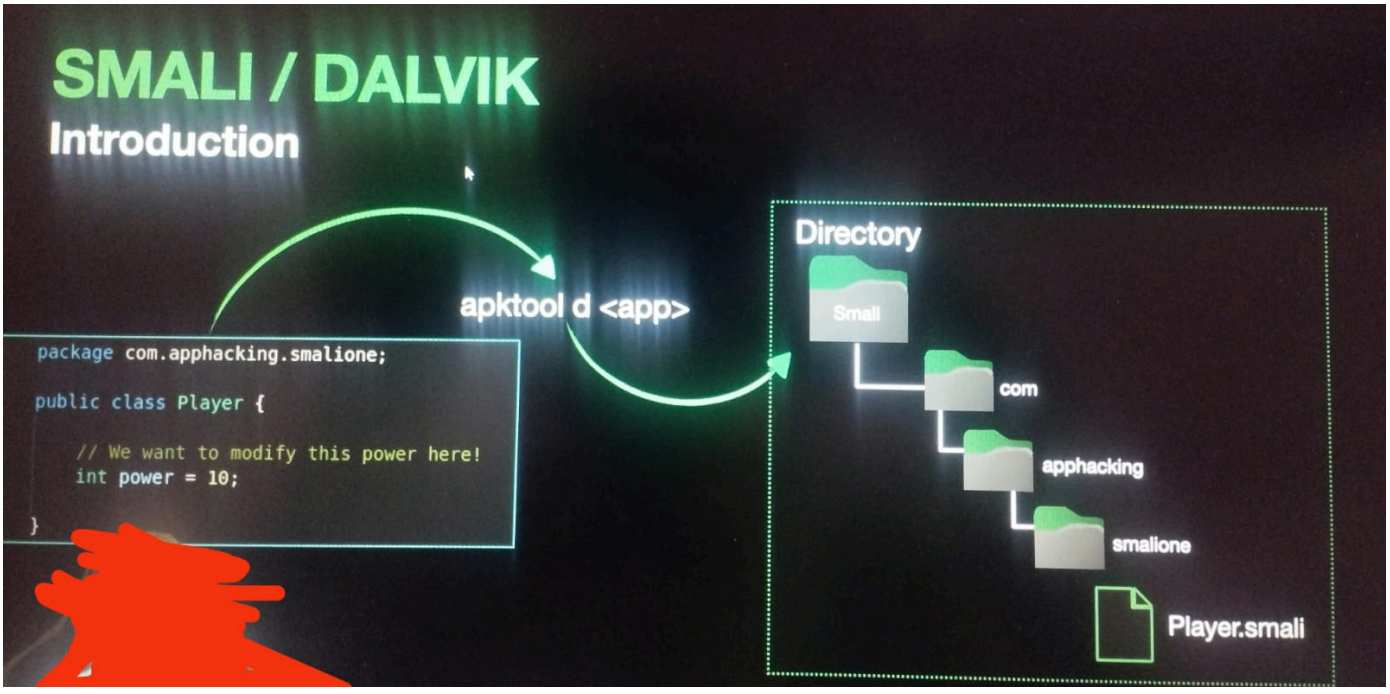


create simple class with java code and see it with smali code



smali code for java

```
.class public Lcom/example/smalione/Player;
.super Ljava/lang/Object;
.source "Player.java"


# instance fields
.field power:I


# direct methods
.method public constructor <init>()V
    .locals 1

    .line 3
    invoke-direct {p0}, Ljava/lang/Object;-><init>()V

    .line 4
    const/16 v0, 0xa

    iput v0, p0, Lcom/example/smalione/Player;->power:I

    return-void
.end method
```



لو جينا بقي نشوف الفرق بين java و smali ممكن الاختلاف اللي نلاحظه ان القيمة اتعرفت في smali قيمة hexa <-- ك **const/16** **v0.0xa**

Feature	Java Code	Smali Code
Class Declaration	`public class Player`	`.class public Lcom/example/smali_code/Player;`
Inheritance	Implicit (`extends Object`)	Explicit: `.super Ljava/lang/Object;`
Field	`int score = 0;`	`.field power:I` (field named `power`)
Field Initialization	`score = 0;` (directly assigned)	`iput v0, p0, Lcom/example/smalione/Player;->power:I` (done in the constructor)
Constructor	Implicit default constructor	Explicit constructor with `invoke-direct` for `super()`
Syntax	High-level, readable	Low-level, assembly-like

52- Smali - Patching

دلوقتي بقي من اللي اتعلمناه عرفنا ان القيمة بتبقي في smali بتبقي hexa دلوقتي لو بقي لو عاوزين نغير في قيمة في تطبيق معين من خلال كود smali

steps to change any thing on apk app using smali

1- decompile apk using `apktool d app.apk`

2- open smali file and change the value

3- build application

`apktool b folder`

4- signing application using this method

- **1- Generate Key**
- keytool -genkey -v -keystore ~/android-app-hack.keystore -alias alias_name -keyalg RSA -keysize 2048 -validity 365

```
keytool -genkey -v -keystore ~/android-app-hack.keystore -alias alias_name -keyalg RSA -keysize 2048 -validity 365
```

- **2- configer lines**
- zipalign -v 4 sieve.apk out.apk

```
zipalign -v 4 sieve.apk out.apk
```

- **3- sign app with apksigner**
- apksigner sign --ks-key-alias my_key -ks ~/android-app-hack.keystore out.apk

```
apksigner sign --ks-key-alias my_key -ks ~/android-app-hack.keystore
out.apk
```

Solving chellagne

1-install app to emulator

```
➜  first_chellange_smali adb install smaliOne.apk
```

2- do logcat to show the value of power

```
➜  first_chellange_smali adb logcat |grep power
```

```
rvice, waited 27850ms
17 02:31:27.134  2978  2978 I System.out: Player power = 10
17 02:31:46.166   847   847 W Looper  : Slow delivery took 248ms main h=com.android.server.power.Notifier$NotifierHandler c=null m=7
17 02:32:13.879   847   847 W Looper  : Slow delivery took 221ms main h=com.android.server.power.Notifier$NotifierHandler c=null m=7
17 02:32:16.900  3975  4244 W PowerStateManagerModule: <DWB> Context manager broadcast triggered power state fence restoration
17 02:32:25.496  3975  4244 W PowerStateManagerModule: <DWB> Context manager broadcast triggered power state fence restoration
17 02:32:46.830   847   879 W Looper  : Slow delivery took 440ms android.fg h=android.os.Handler c=com.android.server.power.hint.HintManagerServi
da1@bce3581 m=0
17 02:33:05.065   847   847 W Looper  : Slow delivery took 233ms main h=com.android.server.power.Notifier$NotifierHandler c=null m=7
```

here power = 10 we need to change it to 20 using smali

3- remove app from emulator

```
➜  ~ adb uninstall com.apphacking.smalione
Success
```

4- decompile app with apktool

```
➜  first_chellange_smali apktool d smaliOne.apk
```

5-open the file of class player on smali and chagne the value from 0xa to 0x14

```
➜  apphacking cd smalione
➜  smalione ls
 BuildConfig.smali   'R$color.smali'         'R$mipmap.smali'
 MainActivity.smali  'R$dimen.smali'         'R$plurals.smali'
 Player.smali        'R$drawable.smali'      'R$string.smali'
'R$animator.smali'   'R$id.smali'            'R$styleable.smali'
'R$anim.smali'       'R$integer.smali'       'R$style.smali'
'R$attr.smali'       'R$interpolator.smali'  'R$xml.smali'
'R$bool.smali'       'R$layout.smali'         R.smali
```

before

```
method public constructor <init>()V
    .locals 1

    .line 3
    invoke-direct {p0}, Ljava/lang/Object;-><init>()V

    .line 5
    const/16 v0, 0xa

    iput v0, p0, Lcom/apphacking/smalione/Player;->power:I
                                    [ Read 23 lines ]
```

after

```
    .line 3
    invoke-direct {p0}, Ljava/lang/Object;-><init>()V

    .line 5
    const/16 v0, 0x14

    iput v0, p0, Lcom/apphacking/smalione/Player;->power:I
```

6- build the file after change the value

```
➜  first_chellange_smali apktool b smaliOne
I: Using Apktool 2.11.0 on smaliOne.apk with 4 threads
I: Checking whether sources have changed...
I: Smaling smali folder into classes.dex...
I: Checking whether resources have changed...
```

the new apk on I: Built apk into: smaliOne/dist/smaliOne.apk

لو جينا بقي نعمله install مش هيرضي علشان احنا غيرنا فيه د

```
db: install requires an argument
 dist adb install smaliOne.apk
erforming Streamed Install
db: failed to install smaliOne.apk: Failure [INSTALL_PARSE_FAILED_NO_CERTIFICATES: Failed to collect certificates from /data/app/vmdl1815250047.tmp/base.apk: Attempt to get length of
ull array]
 dist
```

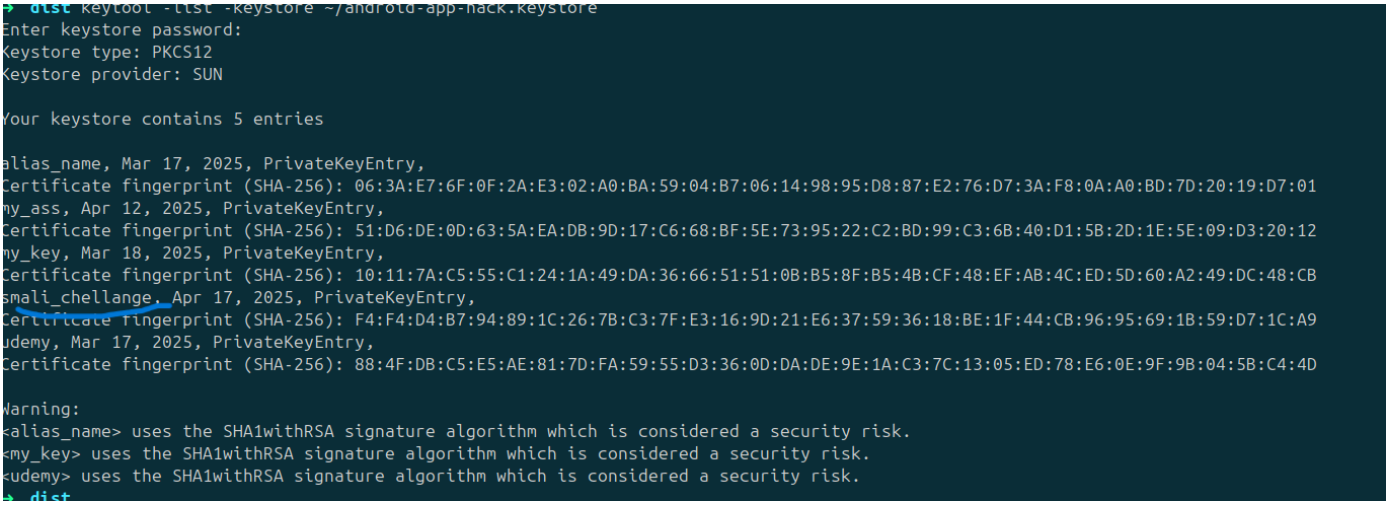
7-signing the application

1- create the key

```
➜  dist keytool -genkey -v -keystore ~/android-app-hack.keystore -keysize
2048 -alias smali_chellange -keyalg RSA -validity 365
Enter keystore password:
```

```
Enter the distinguished name. Provide a single dot (.) to leave a sub-
component empty or press ENTER to use the default value in braces.
```

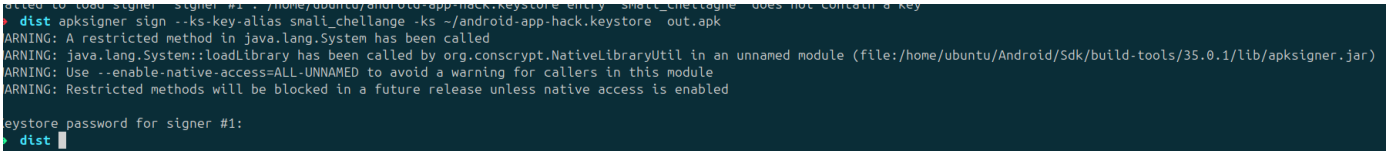
check it with list-key



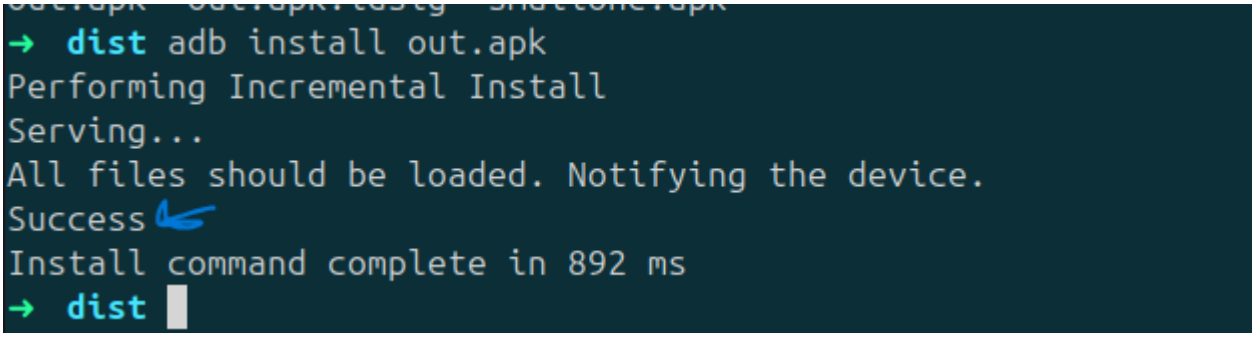
2- configer the lines

```
➜  dist zipalign -v 4 smaliOne.apk out.apk
Verifying alignment of out.apk (4)...
      41 classes.dex (OK - compressed)
```

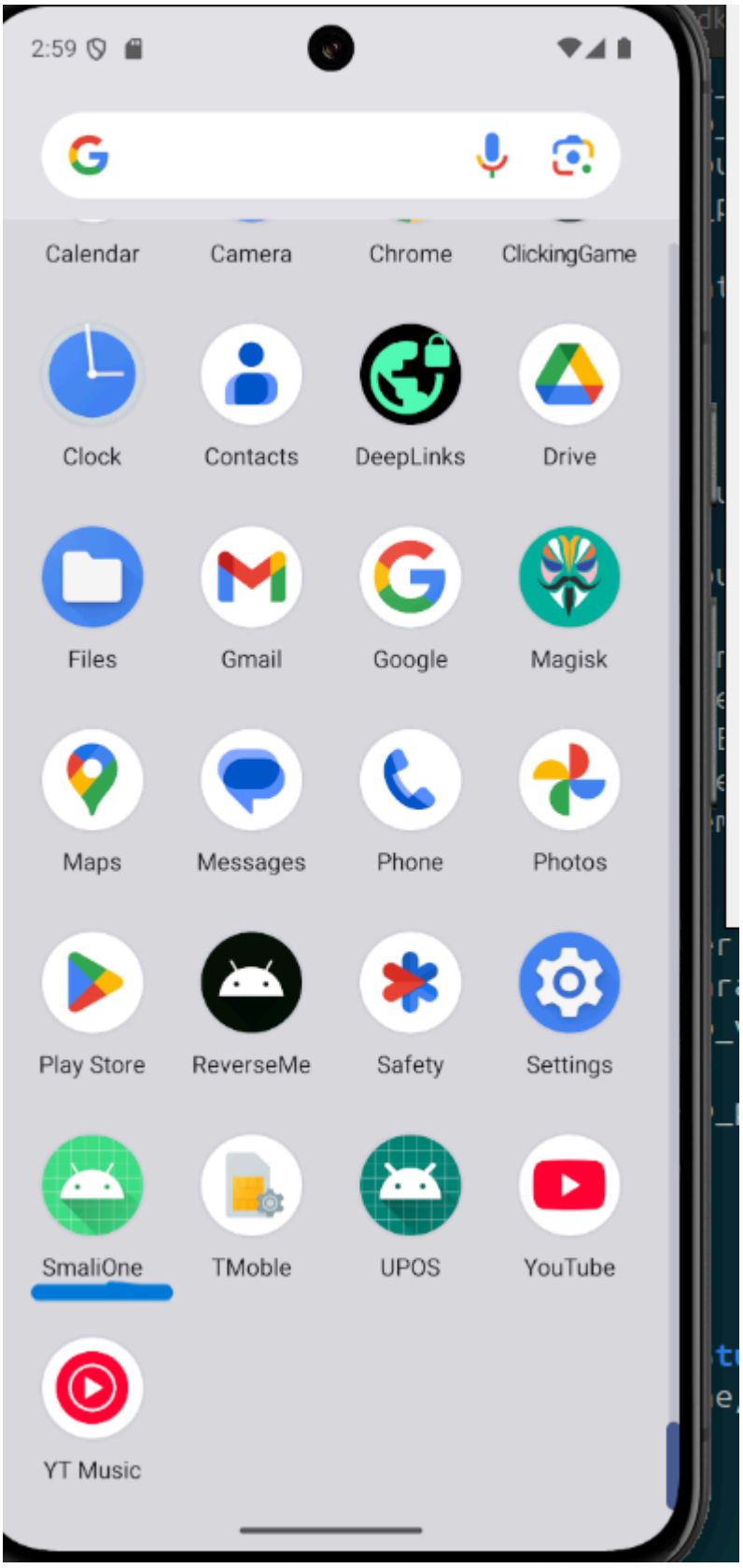
3- sign with



8- upload the apk after change it and check it and we success to change the value because when we upload he give me success



9- check the app is installed or not



10- check the logcat the value is changed or not

finally the value is changed to 20

```
04-17 02:43:14.459  1062  1062 I AvrcpBrowsablePlayerConnector: Successfully connected to 0 browsable players.
04-17 02:43:14.687  1062  1062 I MediaPlayerList: init: Browsable Player list size is 0
04-17 02:43:29.590  2502  2502 I ExoPlayerImpl: Init b4f33c1 [AndroidXMedia3/1.6.0-beta01] [emu64xa, sdk_gphone64_x86_64, Google, 35]
04-17 02:43:29.623  2502  2502 I ExoPlayerImpl: Init 6f5f8c0 [AndroidXMedia3/1.6.0-beta01] [emu64xa, sdk_gphone64_x86_64, Google, 35]
04-17 03:02:48.276  6171  6171 I System.out: Player power = 20
```

اعمل نفس الخطوات اللي عملتها وهتلاقي ان انتا تغير حاجة باستخدام smali وبكده يبقي احنا حلينا chellange اللي هو طلبه

54-Registers

هنا هنشرح ازاي smali بتشتغل وهنشرح شوية حاجات مهمة وهنبدا بأول حاجة وهي ايه هي high level language --> ودي عبارة عن language ان الانسان بيقدر يفهمها وستخدمها واسهل في التعامل معاها زي المثال ده كده :

هنا دي عبارة عن function بتجمع رقمين ودي بكتوبة ايه ان الانسان يقدر يفهما عادي ونروح لموضوع تاني وهو CPU

```
int add(int a,int b){
    int sum=0;
    sum=a+b;
    return sum
}
```

نيجي لل CPU هنا هو مش بيفهم high level languge هو بتترجم ل assembly وهو يقدر يفهم وتتحول machine language علشان OS يقدر يفهمها بس خلينا نتكلم عن CPU هو بيبقي جواه register كل register بيخزن القيم اللي هي زي a , b sum, وهكا وبيقدر يتعامل معاها علشان يخزنها في memory عن طريق data structure stack وبيستخدم LIFO (Last-In-First-Out) وهكذا نيجي بقي smali بتتعامل كأنها زي register بالظبط وده لما نستخدم apktool بيقدر يكون كود smali وبيقسم لحاجتين local و parmater وان

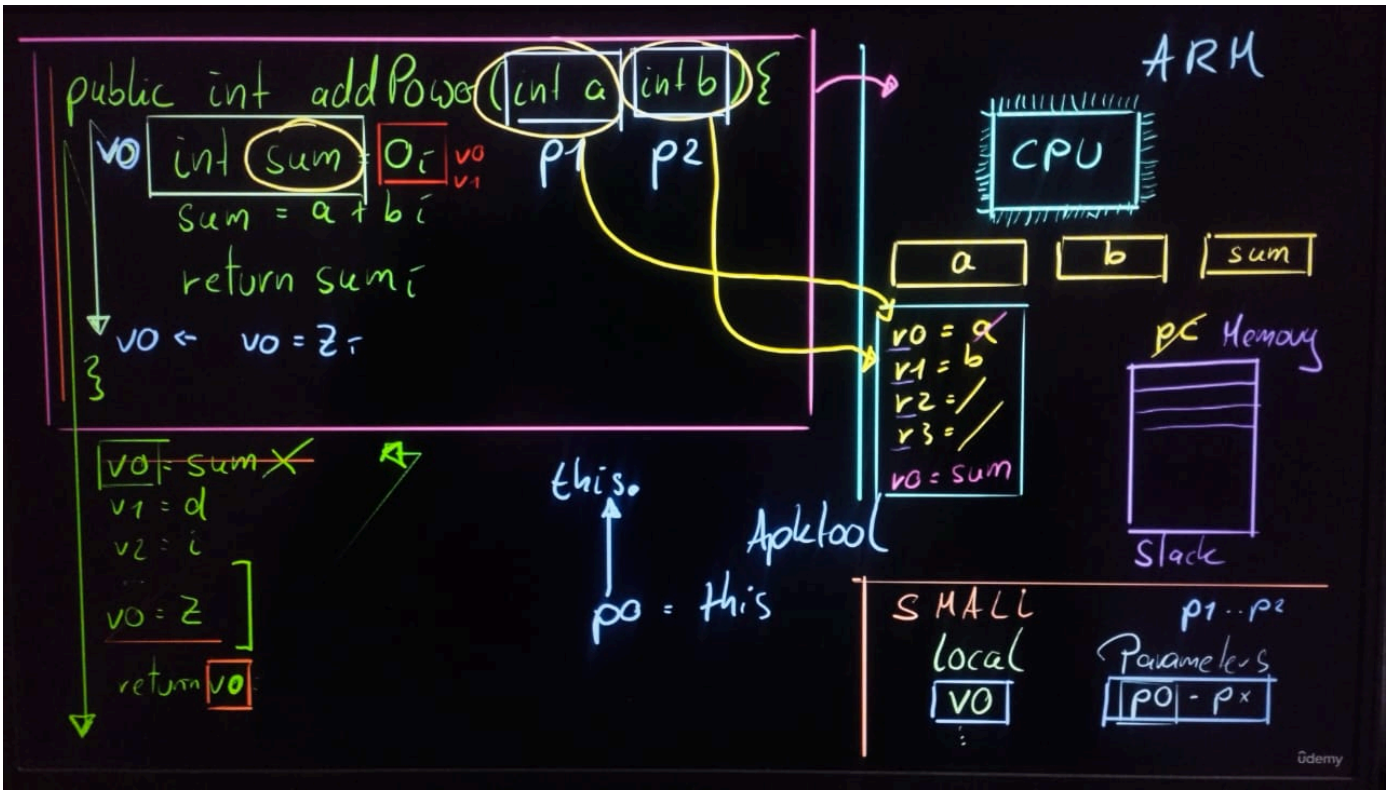
local --> sum --> on smali (V0 -->Vn)

parameters --> a ,b on smali (P0 --> Pn) and P0 assign to this on java

جدول مقارنة بين Java و Smali:

العنصر	Java	Smali
الكائن الحالي	`this`	`p0`
المعامل الأول	`a`	`p1`
المعامل الثاني	`b`	`p2`
المتغير المحلي	`sum`	`v0`

وهنا بقي الصورة دي بنوضح كل حاجة



if we create the add function by java and bulid app and decompile it to show the smali code

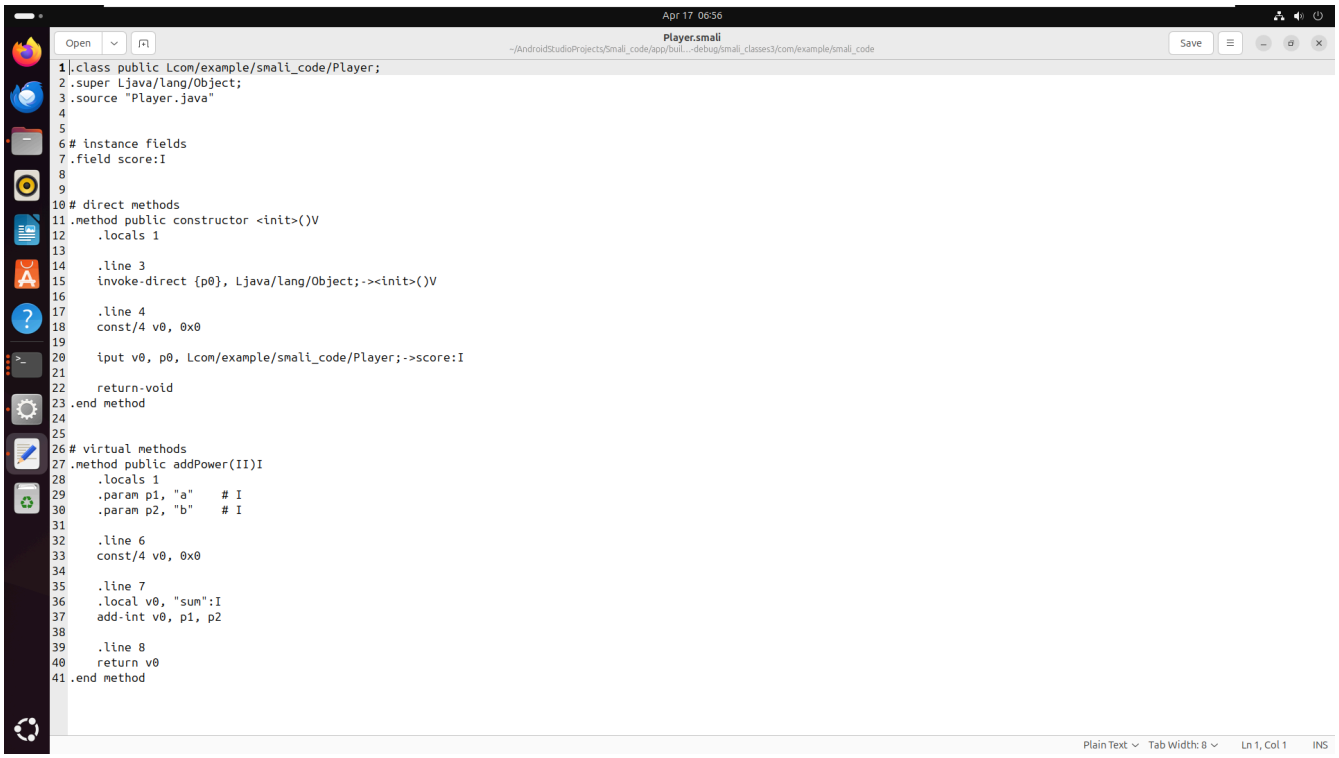
```
package com.example.smali_code;

public class Player {
    int score=0;
    public int addPower(int a,int b){
        int sum=0;
        sum=a+b;
        return sum;
    }
}
```

.method public addPower(II)I ---> public int addPower

.param p1, "a"

.param p2, "b" two mean parameters for function addPower

const/4 v0. 0x0 ---> v0=sum and 0x0 =0



شرح الكود خطوة بخطوة

1. تعريف الدالة

```
.method public addPower(II)I
```

- **`public`**: الدالة عامة ويمكن استدعاؤها من خارج الصف.
- **`addPower`**: اسم الدالة.
- **`(II)I`**:
  - الجزء **`II`** يشير إلى أن الدالة تأخذ مدخلين من النوع `int`.
  - الجزء **`I`** يشير إلى أن الدالة تُرجع قيمة من النوع `int`.

---

2. عدد المتغيرات المحلية

```
.locals 1
```

- يحدد عدد المتغيرات المحلية المستخدمة في هذه الدالة.
- هنا يتم استخدام متغير محلي واحد فقط، وهو **`v0`**.

---

3. تعريف المعاملات

```
.param p1, "a"     # I
.param p2, "b"     # I
```

- **`p1`** و **`p2`** هما المعاملات التي تستقبلها الدالة.
- **`"a"`** و **`"b"`** هما أسماء وصفية لهذه القيم لتوضيح دورها.
- النوع **`I`** يشير إلى أن هذه القيم من النوع `int`.

---

4. تهيئة المتغير المحلي

```
.line 6
const/4 v0, 0x0
```

- يتم تهيئة المتغير `v0` بقيمة صفر.
- `const/4` تُستخدم لتخزين القيم الصغيرة (0-15) بكفاءة.

---

5. عملية الجمع

```
.line 7
.local v0, "sum":I
add-int v0, p1, p2
```

- يتم استخدام التعليمة `add-int` لإضافة المعاملين `p1` و `p2` وتخزين النتيجة في `v0`.
- `v0` يُستخدم هنا لتمثيل مجموع القيم `a` و `b`.

---

6. إرجاع النتيجة

```
.line 8
return v0
```

- يتم إرجاع النتيجة المخزنة في `v0` إلى الدالة المستدعية.

القيم اللي هي line. دي يعني بقنول رقم line اللي في java بنحدده في smali

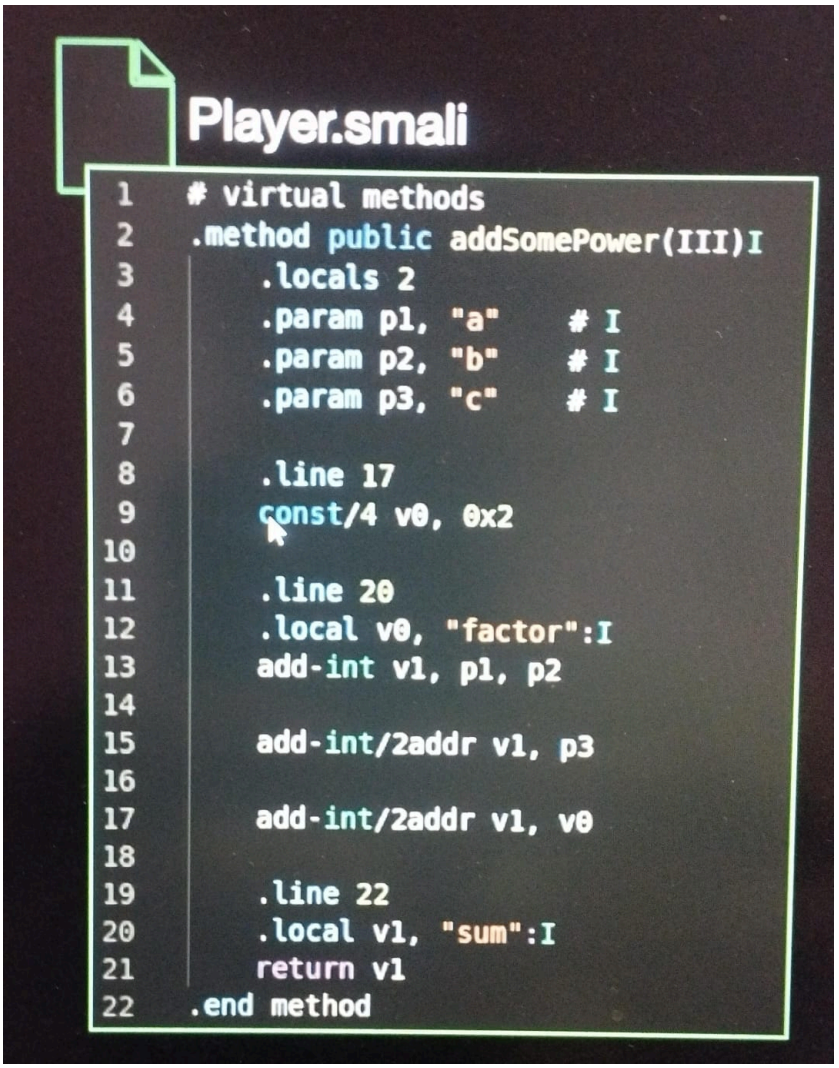
```
.class public Lcom/example/smali_code/Player;
.super Ljava/lang/Object;
.source "Player.java"


# instance fields
.field score:I


# direct methods
.method public constructor <init>()V
    .locals 1

    .line 3
    invoke-direct {p0}, Ljava/lang/Object;-><init>()V

    .line 4
    const/4 v0, 0x0

    iput v0, p0, Lcom/example/smali_code/Player;->score:I

    return-void
.end method


# virtual methods
.method public addPower(II)I
    .locals 1
    .param p1, "a"    # I
    .param p2, "b"    # I

    .line 6
    const/4 v0, 0x0

    .line 7
    .local v0, "sum":I
    add-int v0, p1, p2

    .line 8
    return v0
.end method
```

```
1   package com.example.smali_code;
2
    2 usages
3   public class Player {
        1 usage
4       int score=0;
        no usages
5       public int addPower(int a,int b){
6           int sum=0;
7           sum=a+b;
8           return sum;
9       }
10  }
11
```

يبقي هنا local variable بنعرف ب local. و parameter variable بنعرفهم parameter. وبيدأ بقيمة pn <--- p1 to وان local بيبدأ ب v0 وانا lines في java بتتعرف line.

وان لو عاوز تعرف قيمة ثابتة const.

another example this code with smali so we want to convert it to java code



java code

خلي بالك : .local v0, "factor" كده بقولك ان v0 هيكون فيها المتغير اللي اسمه factor و برضه هكذا مع sum

```
public int addSomePower(int a,int b,int c){
    int factor=2;
    sum=a+b;
    sum+=c;
    sum+=factor;
    return sum;
} this is my analyze
the real code
public int addSomePower(int a,int b,int c){
    int factor=2;
    int sum;
    sum=a+b+c+factor;
    return sum;
}
```

55-Types

Syntax	Meaning
V	Void
B	Byte
Z	Boolean
C	Char
I	Int
J	Long
F	Float
D	Double
[descriptor (e.g [B ---> Byte	array
L	Class name
S	Short

```
# static data
.field public static final SPRITE_NAME:Ljava/lang/String;="alien"
.field public static final SPRITE_NAME_BOSS:Ljava/lang/String;="alienboss"

this mean there are variables:
1-   public static final SPRITE_NAME="alien"   -> Ljava/lang/String --> the data type is string
2-   public static final SPRITE_NAME_BOSS="alienboos"  Ljava/lang/String --> the data type is string
```

ازاي نعرف اسم class

smali code

```
.class public Lde/fgerbig/spacepng/components/ColorAnimatoin;
.super Lcom/artemis/Component;
.source "ColorAnimation.java"
```

java

```
public class ColorAnimation extends Componet{
}
```

Inner classes / enum ($)

in java

```
public class Animaton{
    public class PlayMode{
    }//or
    enum PlayMode{
    }
}
```

in smali

```
.filed public Play:Mode:Lcom/badlogic/gdx/graphics/g2d/Animation$PlayMode;
```

56- P0 - Register

دلوقتي هرنوح نضيف function getpower ونخليها ترجع ليما قيمة power ب استخدام this ونشوفها في كود smali بعد ما نعمل له decompile

get function

```
public int getPower(){
        return this.power;
    }
```

smali

```
method public getPower()I
    .locals 1

    .line 6
    iget v0, p0, Lcom/example/smali_code/Player;->power:I

    return v0
.end method
```

دلوقتي بقي خلينا نشرح كود smali

method public getPower()I : here I mean return integer	**public int getpower(){**
.locals 1	
. line 6 --> for line 6 on code java	line 6 return this.power;

method public getPower()I : here I mean return integer	**public int getpower(){**
iget v0,p0. Lcom/example/smali_code/Player;->power:I	iget-->get value and v0 --> register , p0 --> this , class_name = player and -> for value on class
return v0	return

خلي بالك هنا في حاجة مهمة هو ان احنا مش معرفين variable داخل function ولكن في smali استخدم locals 1 وهنا بيجي لشرح ان زي مثلا لغة assembly ان cpu مش بيتعامل مع قيم معينة ولكن بيتعامل مع register وان القيم لازم تكون مخزنة في register وهنا هو استخدم v0 كانه register هيرجع قيمة power وهنا برده القيمة اللي هي p0 علشان قولنا ان this =p0 و وجنبها بقي اسم class وبيشاور علي القمية اللي جواه اللي هي Power

57-Delvik Opcodes

هنكلم ان operation code اللي في smali ودي زي word اللي في الدرس اللي فات اللي هي Iget ودول عبارة عن العمليات اللي عياز تعملها زي ما قولنا تحط قيمة في REGISTER وعمليات تانية زي add,mutli,.....ولدوقتي هسبلك ملف pdf قي كل operation مثلا iget

52	iget vx, vy, field_id	Reads an instance field into vx. The instance is referenced by vy.	5210 0300 - iget v0, v1, Test2.i6:I // field@0003 Reads field@0003 into v0 (entry #3 in the field id table). The instance is referenced by v1.
53	iget-wide vx,vy,field_id	Reads an instance field into vx[1]. The instance is referenced by vy.	5320 0400 - iget-wide v0, v2, Test2.l0:J // field@0004 Reads field@0004 into v0 and v1 registers (entry #4 in the field id table). The instance is referenced by v2.

54	iget-object vx,vy,field_id	Reads an object reference instance field into vx. The instance is referenced by vy.	iget-object v1, v2, LineReader.fis:Ljava/io/FileInputStream; // field@0002
			Reads field@0002 into v1 (entry #2 in the field id table). The instance is referenced by v2.

55	iget-boolean vx,vy,field_id	Reads a boolean instance field into vx. The instance is referenced by vy.	55FC 0000 - iget-boolean v12, v15, Test2.b0:Z // field@0000 Reads the boolean field@0000 into v12 register (entry #0 in the field id table). The instance is referenced by v15.
56	iget-byte vx,vy,field_id	Reads a byte instance field into vx. The instance is referenced by vy.	5632 0100 - iget-byte v2, v3, Test3.bi1:B // field@0001 Reads the char field@0001 into v2 register (entry #1 in the field id table). The instance is referenced by v3.

57	iget-char vx,vy,field_id	Reads a char instance field into vx. The instance is referenced by vy.	5720 0300 - iget-char v0, v2, Test3.ci1:C // field@0003 Reads the char field@0003 into v0 register (entry #3 in the field id table). The instance is referenced by v2.

ثال لكود smaliونشوفOcode اللي جواه

```
.method public addPower(II)I
    .locals 1
    .param p1,"a" #I
    .param p2,"b" #I
    .line 13
    .const/v4 v0,0x0
    .line 15
    .local v0,"sum":I
    add-int v0,p1,p2
    .line 17
    return v0
.end method
```

هنا ايه add-int بتجمع رقمين وتحطهم في register وبرده هنا const. دي عبارة عن Ocode

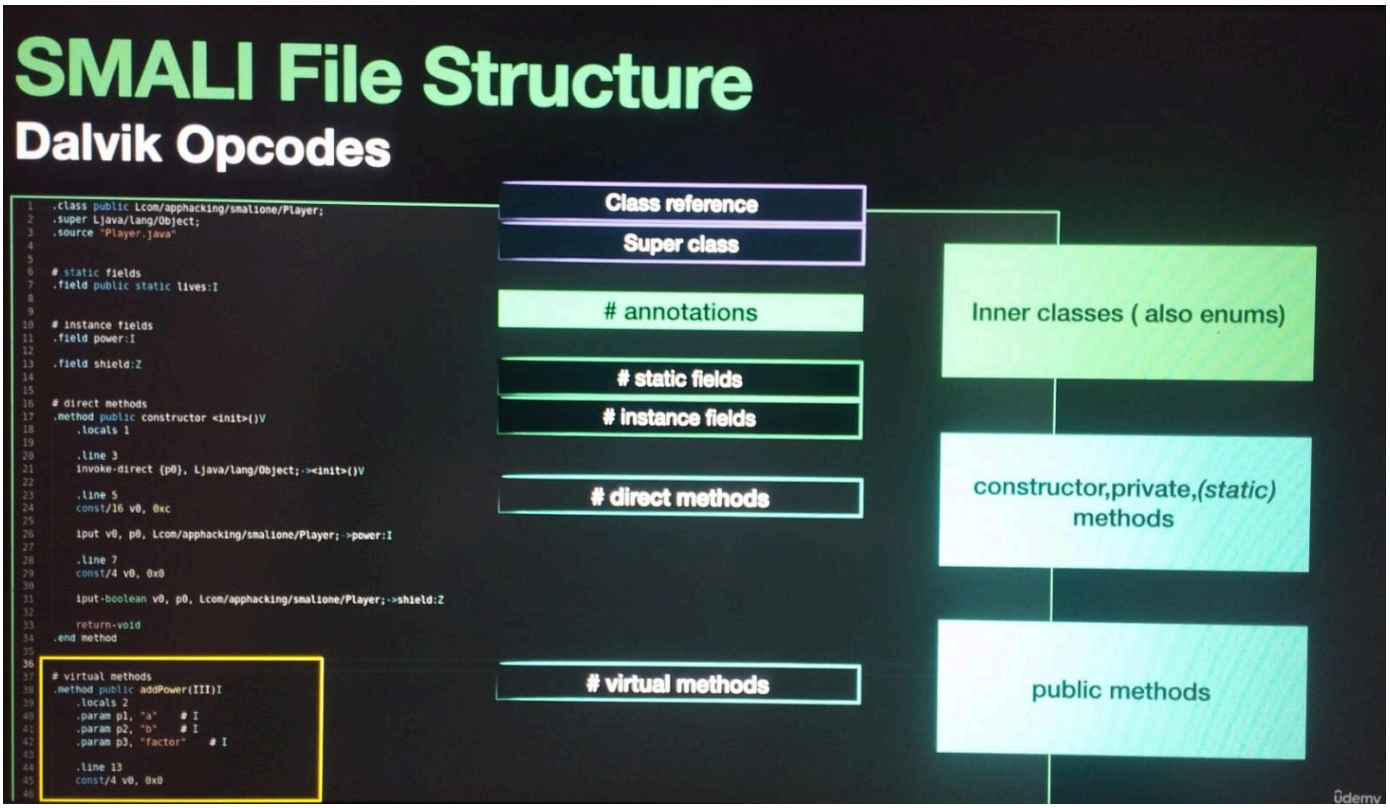
90	add-int vx,vy,vz	Calculates vy+vz and puts the result into vx.	9000 0203 - add-int v0, v2, v3 Adds v3 to v2 and puts the result into v0[4].

12	const/4 vx,lit4	Puts the 4 bit constant into vx	1221 - const/4 v1, #int2 Moves literal 2 into v1. The destination register is in the lower 4 bit in the second byte, the literal 2 is in the higher 4 bit.

java code

```
public int addPower(int a,int b){
    int sum=0;
    sum=a+b;
    return sum;
}
```

58- Smali File Structure



1. تعريف الكلاس (Class Reference):

- يبدأ الملف بتحديد الكلاس باستخدام `.class`.
- يتم تحديد اسم الكلاس مع الـ modifiers (مثل `public`, `final`).

```
.class public Lcom/apphacking/smalione/Player;
```

---

2. الكلاس الأب (Super Class):

- يُحدد الكلاس الأب الذي يرث منه الكلاس الحالي باستخدام `.super`.

```
.super Ljava/lang/Object;
```

---

3. التعليقات التوضيحية (Annotations):

- يمكن أن تحتوي الملفات على **تعليقات توضيحية** تُستخدم لتوثيق أو إضافة خصائص إضافية.

---

4. الحقول (Fields):

أ. الحقول الثابتة (Static Fields):

- يتم تعريف المتغيرات العامة أو الثابتة الخاصة بالكلاس.

```
.field public static lives:I
```

ب. الحقول غير الثابتة (Instance Fields):

- تُعرّف المتغيرات الخاصة بكل نسخة من الكائن.

```
.field power:I.field shield:Z
```

- `I`: تشير إلى نوع Integer.
- `Z`: تشير إلى نوع Boolean.

---

5. الدوال (Methods):

أ. الدوال المباشرة (Direct Methods):

- تتضمن الـ Constructors أو الدوال الخاصة والدوال الثابتة.

```
.method public constructor <init>()V
.locals 1

    invoke-direct {p0}, Ljava/lang/Object;-><init>()V
    return-void
.end method
```

- `.locals 1`: يُحدد عدد السجلات المستخدمة في الدالة.
- `invoke-direct`: يستدعي الدالة من الكلاس الأب.

ب. الدوال الافتراضية (Virtual Methods):

- تُستخدم للدوال العامة التي يمكن استدعاؤها على الكائنات.

```
.method public addPower(III)I
.locals 2

    const/4 v0, 0x0
    return v0
.end method
```

- **`addPower(III)I`**: تشير إلى دالة تأخذ 3 معاملات من نوع Integer وتُعيد قيمة من نوع Integer.
- `const/4`: تُستخدم لتحميل قيمة ثابتة إلى سجل.

---

6. Inner Classes (الكلاسات الداخلية):

- يتم تضمينها في نفس الملف أو كملف مستقل، إذا كانت هناك **كلاسات داخلية** أو **Enums**.

التقسيم:

- **Static Fields**: الحقول الثابتة.
- **Instance Fields**: الحقول الخاصة بالكائنات.
- **Direct Methods**: تشمل الـ Constructors والدوال الخاصة.
- **Virtual Methods**: تشمل الدوال العامة.

59- Practice -Smali

هنا هنتعلم ازاي نفهم code smali ونحاول نفهم اللي جواه ونقارنه ب java code

1- class initializing

```
.class public Lcom/apphacking/smalione/Player;
.super Ljava/lang/Object;
.source "Player.java"


mean : public class Player extends Object(){}
```

2- initializing static and instance variable

```
# static fields
.field public static lives:I


# instance fields
.field power:I

.field shield:Z

mean :
public int lives;
int power;
boolean shield;
```

3- initializing constructor to set values for static and instance variable

```
.method public constructor <init>()V
    .registers 2

    .line 3
    invoke-direct {p0}, Ljava/lang/Object;-><init>()V

    .line 5
    const/16 v0, 0xc

    iput v0, p0, Lcom/apphacking/smalione/Player;->power:I

    .line 7
    const/4 v0, 0x0

    iput-boolean v0, p0, Lcom/apphacking/smalione/Player;->shield:Z
```

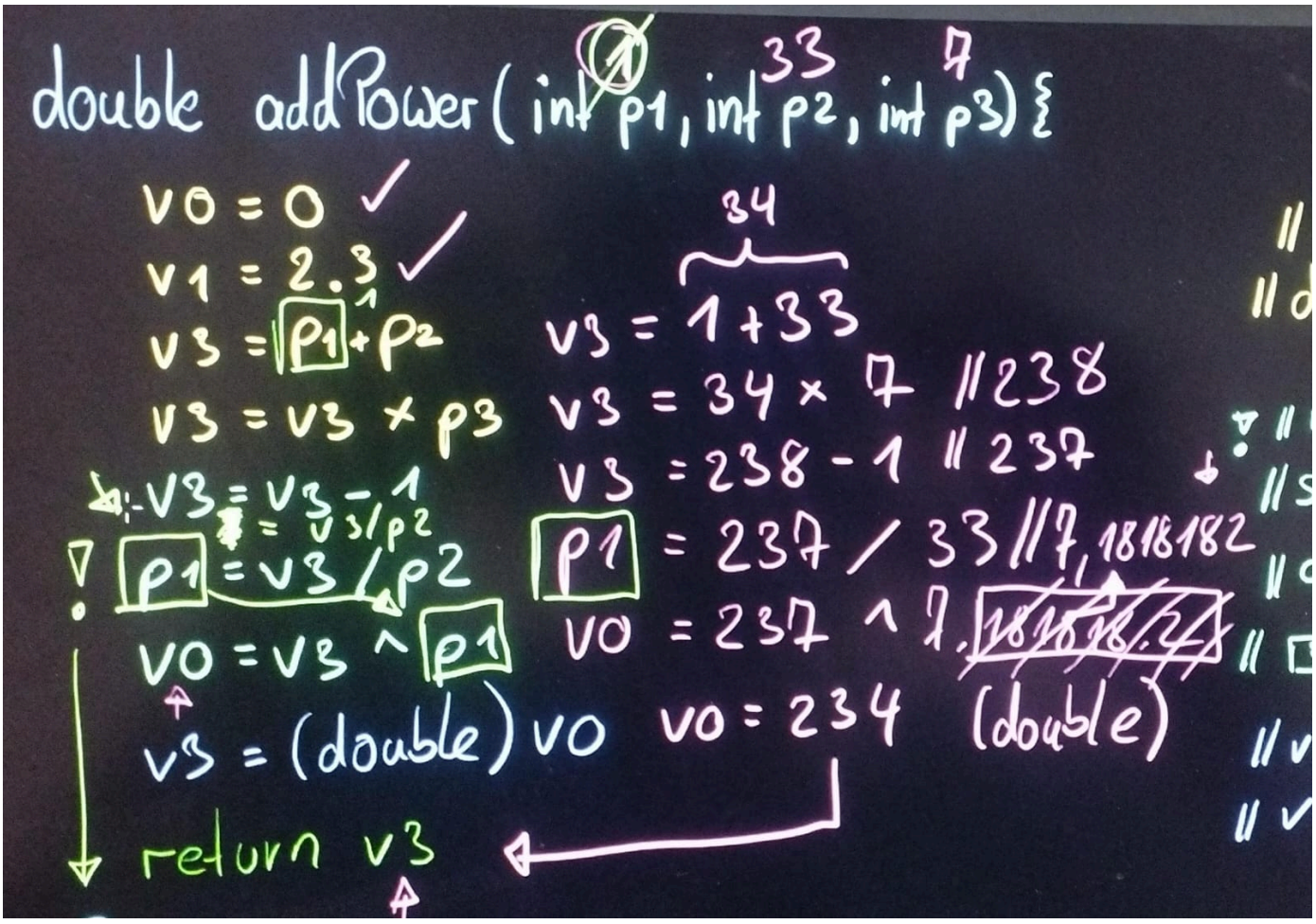
```
    return-void
.end method

mean in line 5 --> power=12
in line 7  --> change the value from 12 to 0 mean (false) --> shiled=false;
```

4-initializing public function

```
.method public addPower(III)D
    .registers 9
    .param p1, "a"    # I
    .param p2, "b"    # I
    .param p3, "factor"    # I

    .line 13
    const/4 v0, 0x0

    .line 15
    .local v0, "sum":I
    const-wide v1, 0x4002666666666666L    # 2.3

    .line 17
    .local v1, "bias":D
    add-int v3, p1, p2

    mul-int v3, v3, p3

    .line 19
    .end local v0    # "sum":I
    .local v3, "sum":I
    add-int/lit8 v3, v3, -0x1

    .line 21
    div-int p1, v3, p2

    .line 23
    xor-int v0, v3, p1

    .line 25
    .end local v3    # "sum":I
    .restart local v0    # "sum":I
    int-to-double v3, v0

    return-wide v3
```

```
.end method

mean
1- .method public addPower(III)D  -->   public Double addPower(int a,int
b,int factor)
2-          .param p1, "a"    # I
            .param p2, "b"    # I
            .param p3, "factor"    # I
          take three parameter a, b, factor and the data type for each one
is int
3- in line 13 : create register v0 and the value of it =0
4- in line 15 : set the value of v0 to sum
5- const-wide v1, 0x4002666666666666L    # 2.3 -->  set value of data type
long to register v1
6- in line 17 : create register v1   =bias the data type of it is double
v1=2.3
7-add-int v3, p1, p2 --> v3 =p1+p2  --> v3=a+b
8-mul-int v3, v3, p3   --> v3=v3*p3   ---> v3=v3*factor      v3 became v3=
(a+b)*factor
9-.local v3, "sum":I --> v3=sum and the data type is int
10- add-int/lit8 v3, v3, -0x1 --> v3=v3-1 ---> v3 became v3=
(((a+b)*factor)-1)
11- in line 21 --> p1=v3/p2 ---> a=sum/b
12- in line 23 --> v0=sum^p1  --> v0 = sum^a  --> xor  if the difference is
1 else 0
13-     int-to-double v3, v0 --> convert the value from v0 int to v3 to be
double
14- return v3 (double)
```

if we set the number for each variable --> a=1 , b=33 , factor=7 the final result will be 234

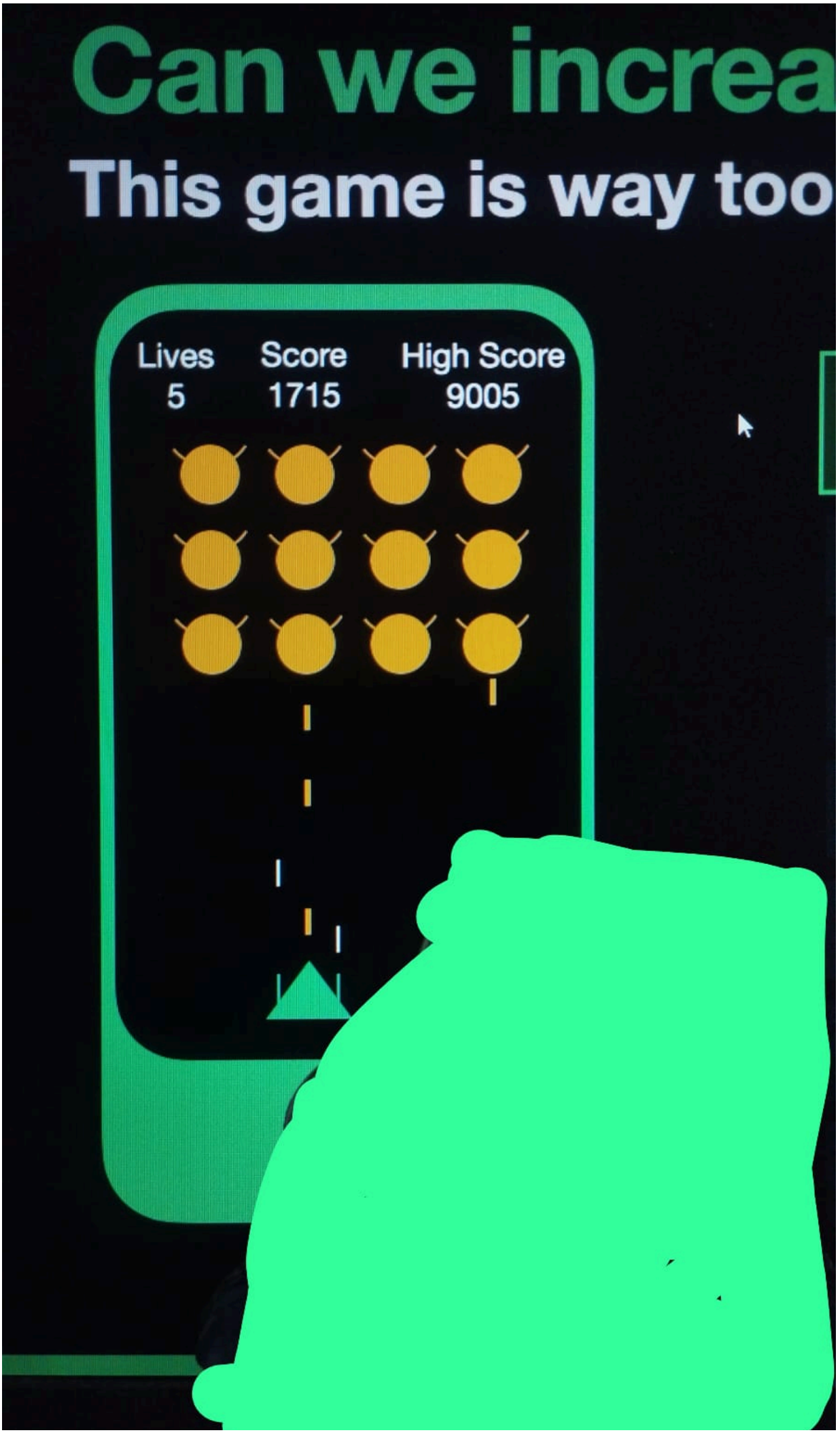


this code by java is

```
ackage com.apphacking.smalione;

/* loaded from: classes.dex */
public class Player {
    public static int lives;
    int power = 12;
    boolean shield = false;

    public double addPower(int a, int b, int factor) {
        int sum = ((a + b) * factor) - 1;
        int a2 = sum / b;
        return sum ^ a2;
    }
}
```

62- Oragne Belt chellange

دلوقتي هو عاوز في التحدي دي ان اغير قيمة Lives من 5 الي 7 ولكن باستخدام كود smali

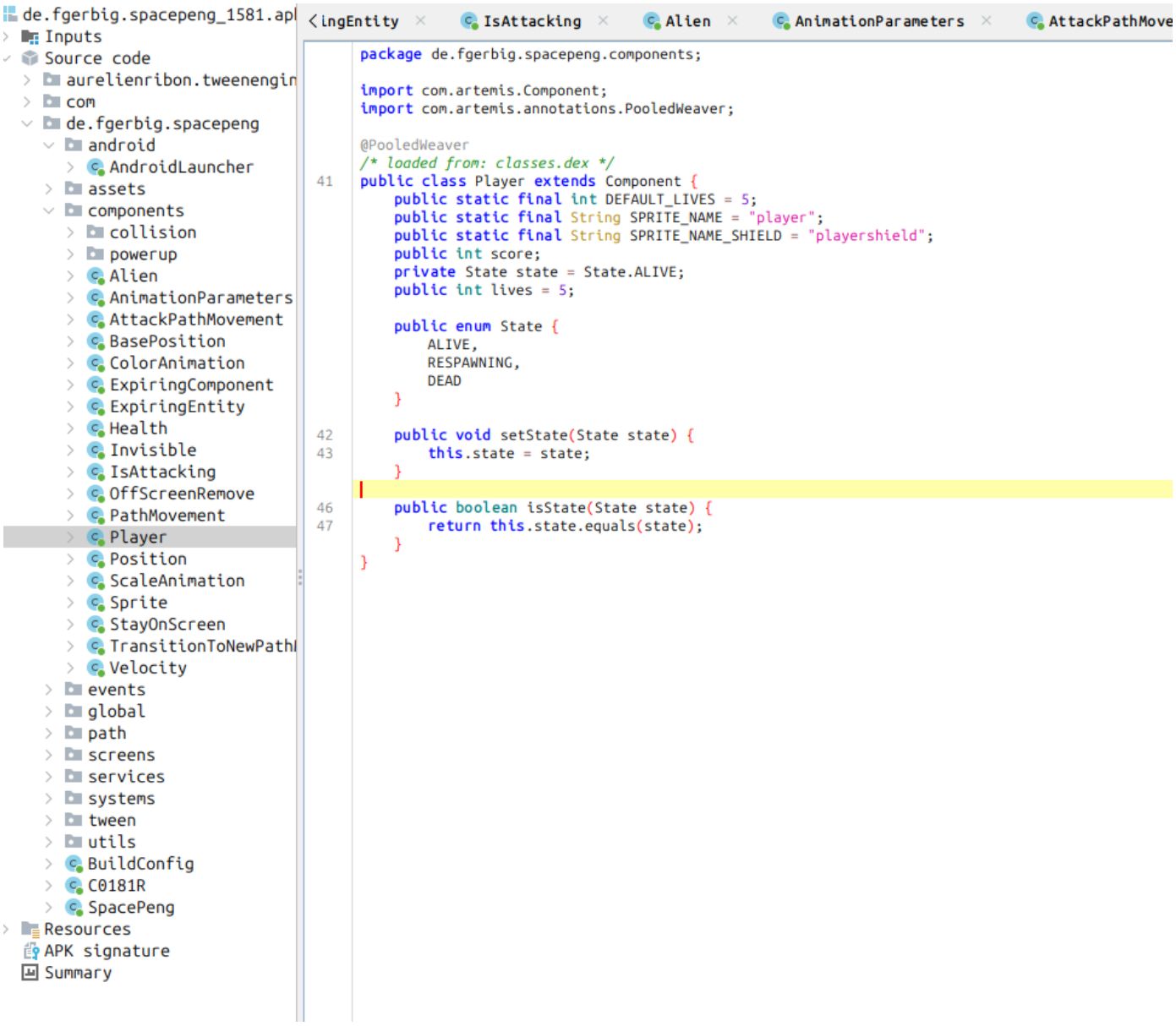

Can we increa
This game is way too
Lives
5
Score
1715
High Score
9005

1- decompile apk app

```
➜  challange_smali apktool d de.fgerbig.spacepeng_1581.apk
I: Using Apktool 2.11.0 on de.fgerbig.spacepeng_1581.apk with 4 threads
I: Baksmaling classes.dex...
I: Loading resource table...
I: Decoding file-resources...
I: Decoding values */* XMLs...
I: Loading resource table from file:
/home/ubuntu/.local/share/apktool/framework/1.apk
I: Decoding AndroidManifest.xml with resources...
I: Regular manifest package...
```

2- open app with jadx-gui tool, and after open with jadx and analyze find the file contain the value of live

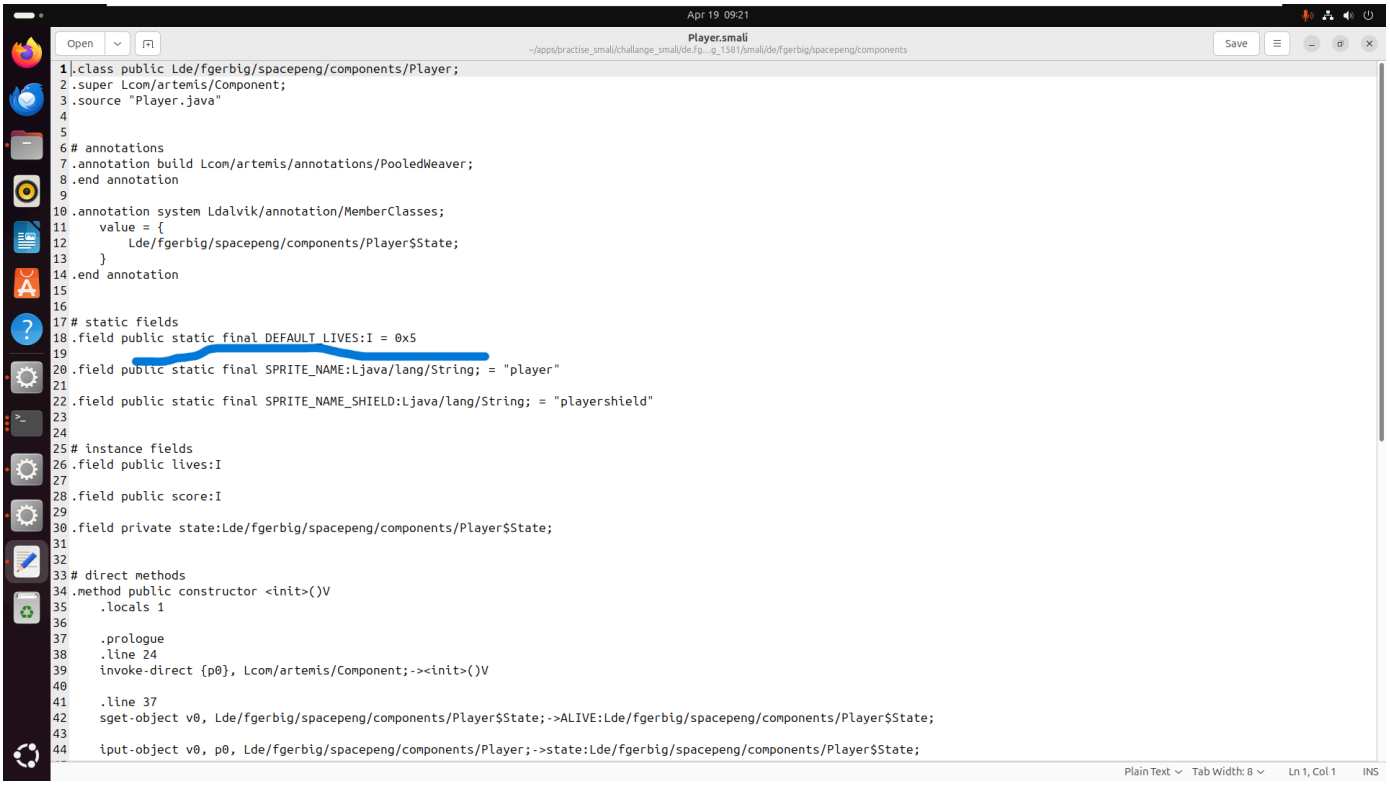
here file /components/player ---> the default value is 5



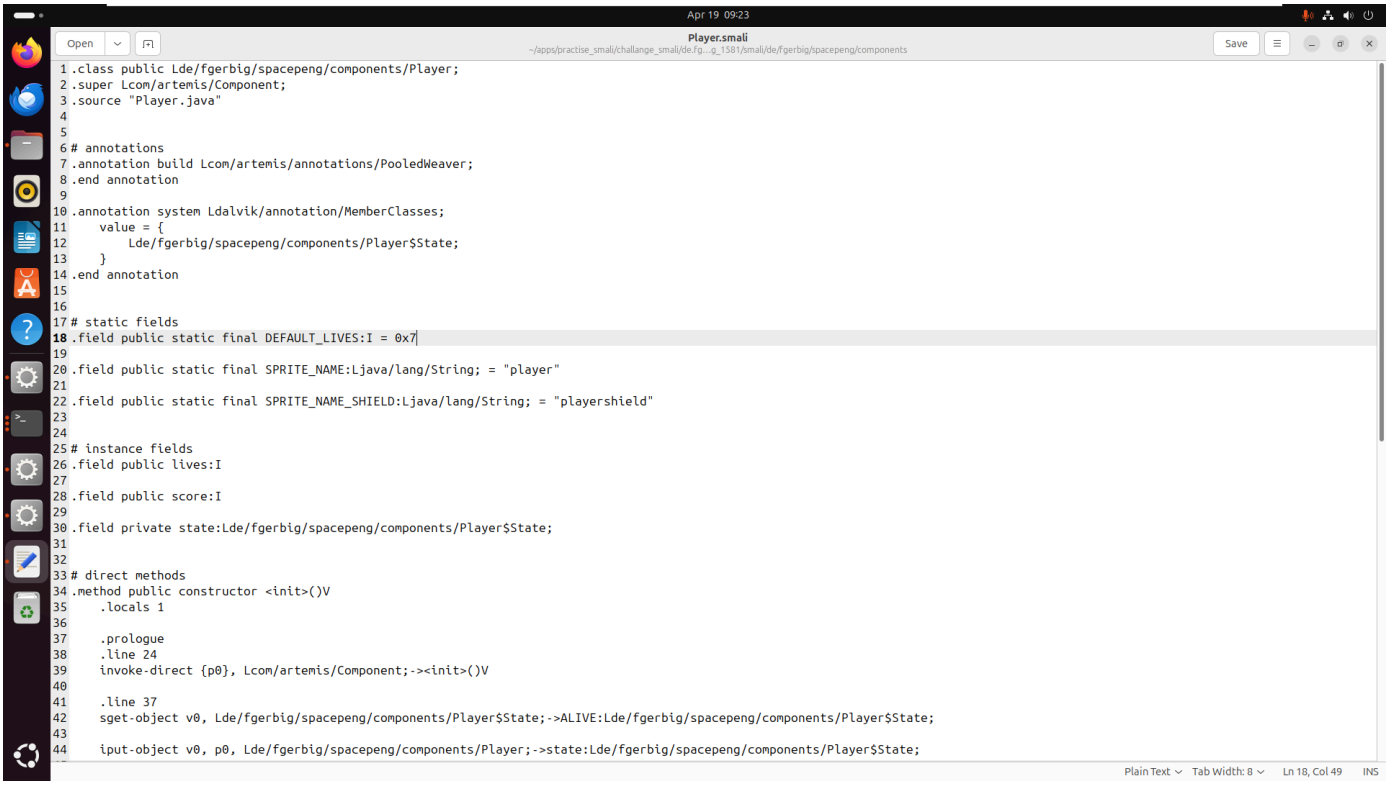
3-find the file location after decompile the app

```
→ components ls
Alien.smali                 collision                Health.smali           PathMovement.smali    powerup                 StayOnScreen.smali
AnimationParameters.smali   ColorAnimation.smali     Invisible.smali        'Player$State.smali'  ScaleAnimation.smali    TransitionToNewPathMovement.smali
AttackPathMovement.smali    ExpiringComponent.smali  IsAttacking.smali      Player.smali          'Sprite$Layer.smali'    Velocity.smali
BasePosition.smali          ExpiringEntity.smali     OffScreenRemove.smali  Position.smali        Sprite.smali
→ components pwd
/home/ubuntu/apps/practise_smali/challange_smali/de.fgerbig.spacepeng_1581/smali/de/fgerbig/spacepeng/components
→ components
```

4- open Player.smali and change the value, here the value by hexa = 0x5, so we will change it to 0x7 --> to be 7 by decimal



change it to 0x7 and save it



5- bulid the app with apktool

```
➜  challange_smali apktool b de.fgerbig.spacepeng_1581
I: Using Apktool 2.11.0 on de.fgerbig.spacepeng_1581.apk with 4 threads
I: Checking whether sources have changed...
I: Smaling smali folder into classes.dex...
I: Checking whether resources have changed...
I: Building resources with aapt2...
I: Building apk file...
I: Importing assets...
I: Importing lib...
I: Importing unknown files...
I: Built apk into:
de.fgerbig.spacepeng_1581/dist/de.fgerbig.spacepeng_1581.apk
```

```
➜  challange_smali cd de.fgerbig.spacepeng_1581
➜  de.fgerbig.spacepeng_1581 cd dist
➜  dist ls
de.fgerbig.spacepeng_1581.apk
➜  dist
```

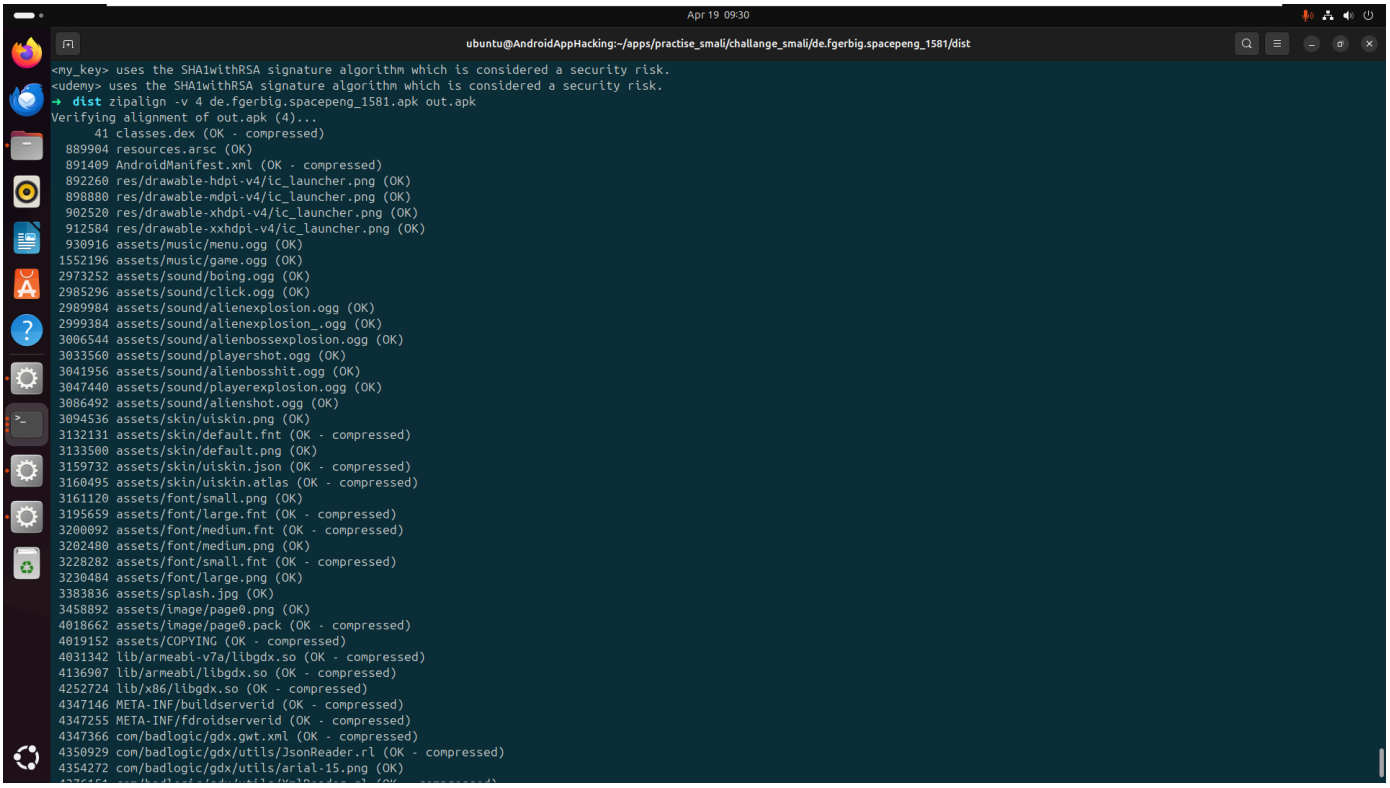
6- create key with alias name to use it in signing the app

```
➜  dist keytool -genkey -v -keystore ~/android-app-hack.keystore -keysize 2048 -alias smalie_challange_key -keyalg RSA -validity 365
Enter keystore password:
Enter the distinguished name. Provide a single dot (.) to leave a sub-component empty or press ENTER to use the default value in braces.
What is your first and last name?
  [Unknown]:
What is the name of your organizational unit?
  [Unknown]:
What is the name of your organization?
  [Unknown]:
What is the name of your City or Locality?
  [Unknown]:
What is the name of your State or Province?
  [Unknown]:
What is the two-letter country code for this unit?
  [Unknown]:
Is CN=Unknown, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown correct?
  [no]:  y

Generating 2048-bit RSA key pair and self-signed certificate (SHA384withRSA) with a validity of 365 days
        for: CN=Unknown, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown
[Storing /home/ubuntu/android-app-hack.keystore]
➜  dist
```

check the key was created

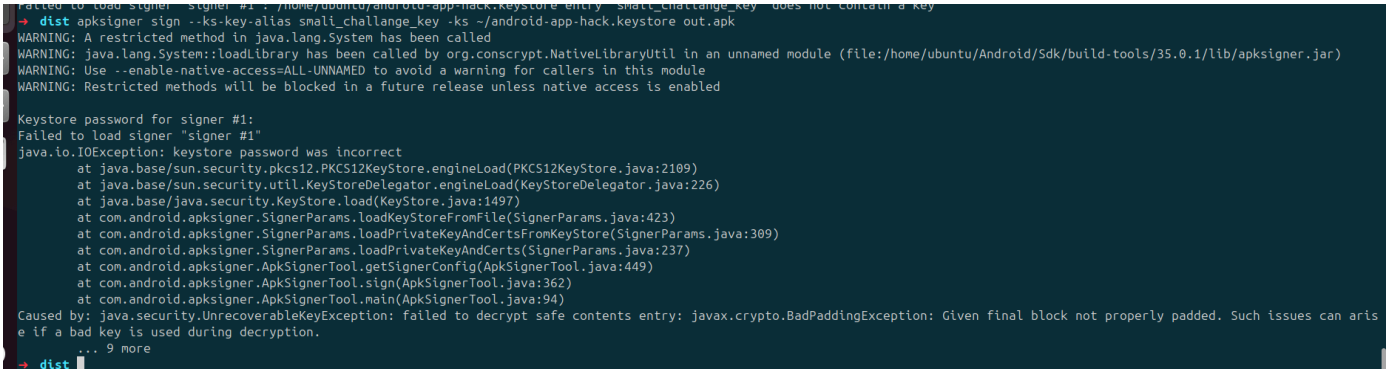
```
➜  dist keytool -list -keystore ~/android-app-hack.keystore
Enter keystore password:
Keystore type: PKCS12
Keystore provider: SUN

Your keystore contains 6 entries

alias_name, Mar 17, 2025, PrivateKeyEntry,
Certificate fingerprint (SHA-256): 06:3A:E7:6F:0F:2A:E3:02:A0:BA:59:04:B7:06:14:98:95:D8:87:E2:76:D7:3A:F8:0A:A0:BD:7D:20:19:D7:01
my_ass, Apr 12, 2025, PrivateKeyEntry,
Certificate fingerprint (SHA-256): 51:D6:DE:0D:63:5A:EA:DB:9D:17:C6:68:BF:5E:73:95:22:C2:BD:99:C3:6B:40:D1:5B:2D:1E:5E:09:D3:20:12
my_key, Mar 18, 2025, PrivateKeyEntry,
Certificate fingerprint (SHA-256): 10:11:7A:C5:55:C1:24:1A:49:DA:36:66:51:51:0B:B5:8F:B5:4B:CF:48:EF:AB:4C:ED:5D:60:A2:49:DC:48:CB
smali_chellange, Apr 17, 2025, PrivateKeyEntry,
Certificate fingerprint (SHA-256): F4:F4:D4:B7:94:89:1C:26:7B:C3:7F:E3:16:9D:21:E6:37:59:36:18:BE:1F:44:CB:96:95:69:1B:59:D7:1C:A9
smalie_challange_key, Apr 19, 2025, PrivateKeyEntry,
Certificate fingerprint (SHA-256): 09:AE:13:29:3D:2C:A5:7F:63:6E:8F:D4:F1:94:14:69:48:DD:0D:9F:3F:E4:C9:C4:0A:39:F5:1E:6C:AF:D2:FD
udemy, Mar 17, 2025, PrivateKeyEntry,
Certificate fingerprint (SHA-256): 88:4F:DB:C5:E5:AE:81:7D:FA:59:55:D3:36:0D:DA:DE:9E:1A:C3:7C:13:05:ED:78:E6:0E:9F:9B:04:5B:C4:4D

Warning:
<alias_name> uses the SHA1withRSA signature algorithm which is considered a security risk.
<my_key> uses the SHA1withRSA signature algorithm which is considered a security risk.
<udemy> uses the SHA1withRSA signature algorithm which is considered a security risk.
➜  dist
```

7- configer the lines zipalign tool



8-Sign the key with apksigner

successful sign the app

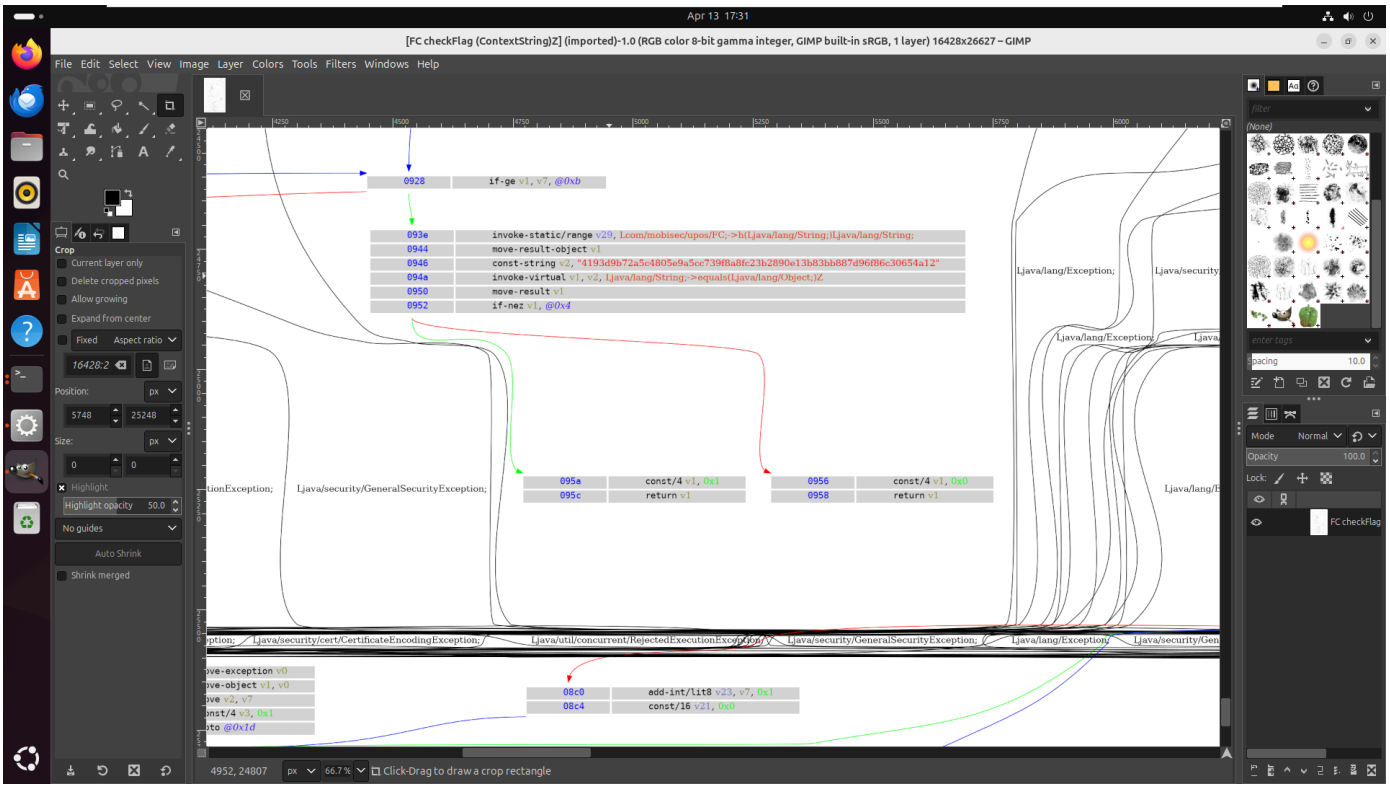


9- upload it with adb shell

```
adb install out.apk
```

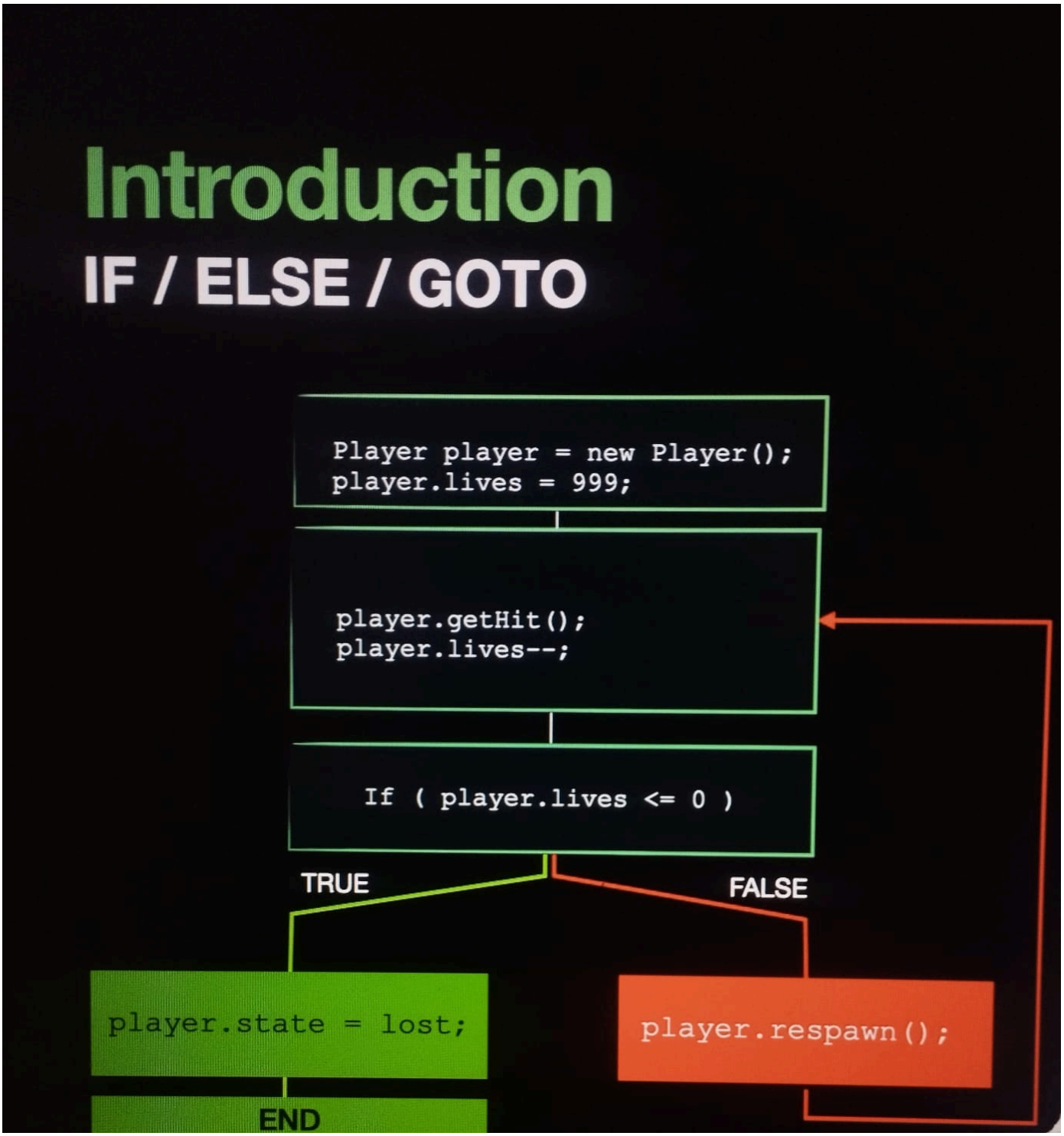
63- IF - intro

هنبدا نتكلم في topics اللي جاية عن if statement وازاي بنستخدمها و ازاي بنعرفها في Flow Graph فمثلا في FG ده if statement هي الخطوط الحمرا والخضرا



64 - IF-ELSE

هنا اه مثلا نفترض ان ده FG فلو ان statement is true هيبقي اللون اخضر ولو العكس احمر وهنا هنا بيوضح حالة lives وان لو ما بتخسر بتقل واحد وان لو هي اصغر من او تساوي صفر هتبقي خسرت ويخرج ولو اكبر من 0 هيكمل عادي ودلوقتي هشنوف smali code for it



دلوقتي هنروح بقي نعمل التطبيق اللي يعملنا الشرط ده من خلال الاكواد دي

1-Player.java

activity_main.xml | MainActivity.java | Player.java

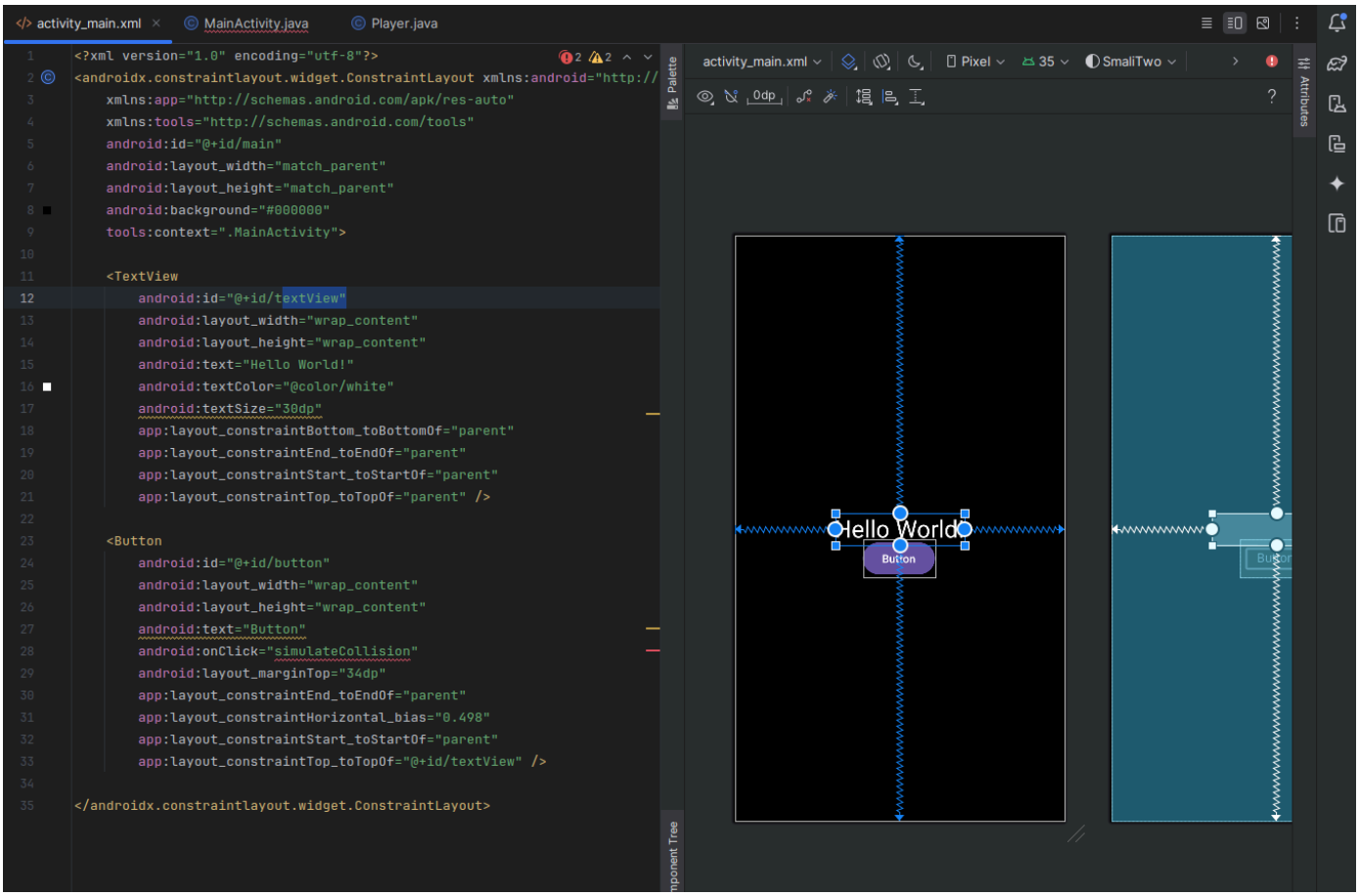
```java
package com.example.smalitwo;

2 usages
public class Player {
    no usages
    int power=10;
    2 usages
    int lives=5;
    2 usages
    boolean state=true;
    1 usage
    public void getHit(){
        this.lives--;
        if(this.lives>0) {
            this.state=true;
        }else {
        this.state=false;
        }
    }
}

```

2-MainActivity.java

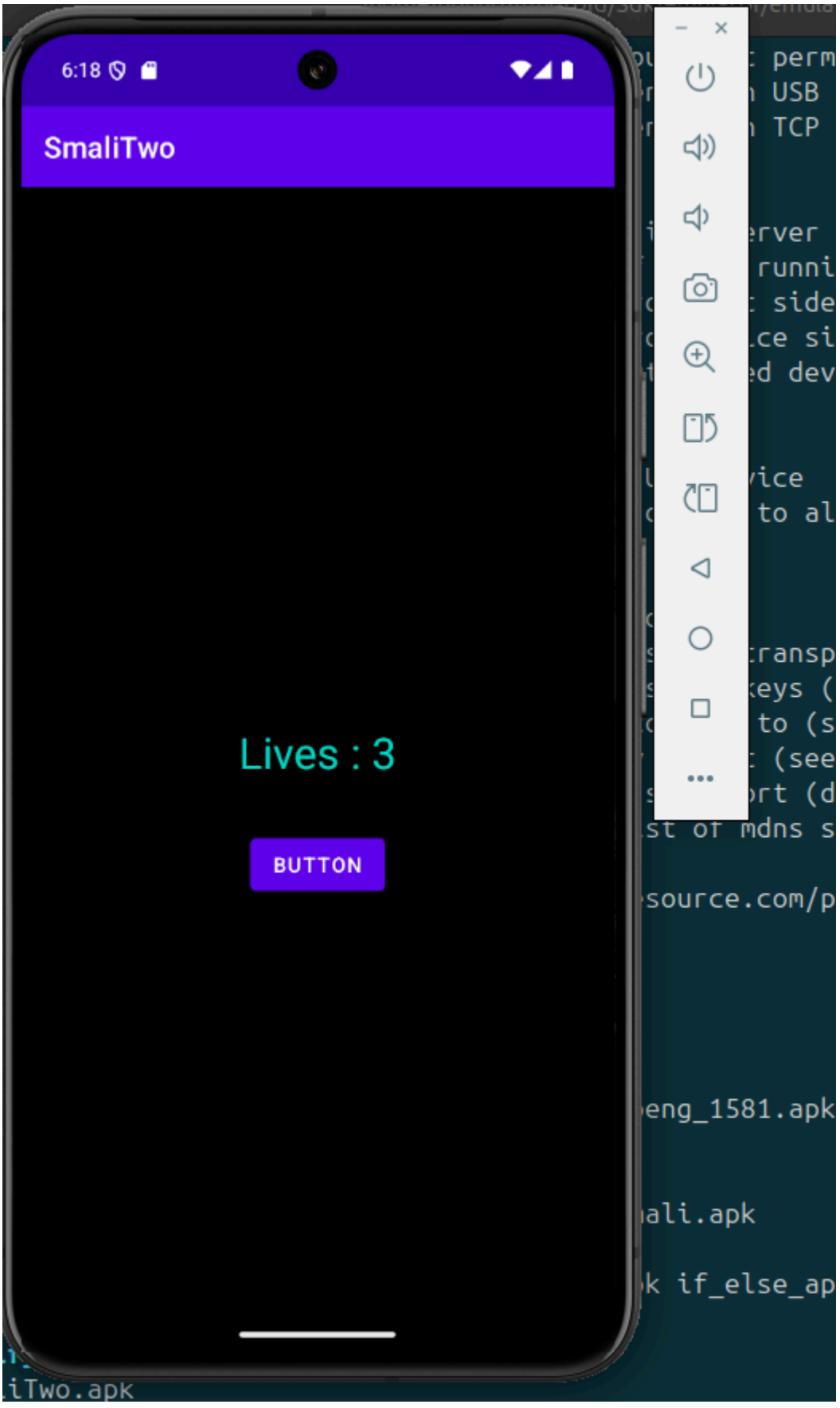
activity_main.xml | MainActivity.java | Player.java

```java
package com.example.smalitwo;

import ...

public class MainActivity extends AppCompatActivity {
    2 usages
    Player player=new Player();
    2 usages
    TextView textview;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable( $this$enableEdgeToEdge: this);
        setContentView(R.layout.activity_main);
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
            Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
            v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
            return insets;
        });
        textview=(TextView) findViewById(R.id.textView);
    }
    1 usage
    public void simulateCollision(View view){

        player.getHit();
        textview.setText("Lives : " + player.lives);

    }
}
```

3- activity_main.xml ننشيء عاوزين لو مثلا خلاله من ده button or text .....



app after install it on emulator and each time click on the button the value of lives decrease by 1



show Smali code for this app after decompile it or using jadx-gui + app.apk

Player

```
###### Class com.apphacking.smalitwo.Player (com.apphacking.smalitwo.Player)
.class public Lcom/apphacking/smalitwo/Player;
.super Ljava/lang/Object;
.source "Player.java"


# instance fields
.field lives:I

.field power:I

.field state:Z


# direct methods
.method public constructor <init>()V
    .registers 2

    .line 3
    invoke-direct {p0}, Ljava/lang/Object; -><init>()V

    .line 5
    const/16 v0, 0xa

    iput v0, p0, Lcom/apphacking/smalitwo/Player;  ->power:I

    .line 6
    const/4 v0, 0x5

    iput v0, p0, Lcom/apphacking/smalitwo/Player;  ->lives:I

    .line 11
    const/4 v0, 0x1

    iput-boolean v0, p0, Lcom/apphacking/smalitwo/Player;  ->state:Z

    return-void
.end method


# virtual methods
.method public getHit()V
    .registers 3

    .line 17
    iget v0, p0, Lcom/apphacking/smalitwo/Player;  ->lives:I

    const/4 v1, 0x1

    sub-int/2addr v0, v1

    iput v0, p0, Lcom/apphacking/smalitwo/Player;  ->lives:I

    .line 19
    if-lez v0, :cond_b

    .line 21
    iput-boolean v1, p0, Lcom/apphacking/smalitwo/Player;  ->state:Z

    goto :goto_e

    .line 24
    :cond_b
    const/4 v0, 0x0

    iput-boolean v0, p0, Lcom/apphacking/smalitwo/Player;  ->state:Z

    .line 26
    :goto_e
    return-void
.end method

```

65- IF-Else code analysis

دلوقتي بقي عاوزين نحلل الكود ده (smali)

1- classe name

```
.class public Lcom/apphacking/smalitwo/Player;
.super Ljava/lang/Object;
.source "Player.java"

public class Player extends Object(){}
```

2- instance field

```
# instance fields
.field lives:I

.field power:I

.field state:Z

int lives
int power
boolean state
```

3- direct method --> constructor for initializing the variable

```
# direct methods
.method public constructor <init>()V
    .registers 2

    .line 3
    invoke-direct {p0}, Ljava/lang/Object;-><init>()V

    .line 5
    const/16 v0, 0xa           ----> set  10 for v0 --> v0 =10

    iput v0, p0, Lcom/apphacking/smalitwo/Player;->power:I  --->
this.live=v0   --> this.live=10

    .line 6
    const/4 v0, 0x5        ---->  set 5 for v0  --> v0=5

    iput v0, p0, Lcom/apphacking/smalitwo/Player;->lives:I    --->
this.power=v0   --> this.power=5

    .line 11
    const/4 v0, 0x1   ---> set 1 for v0   --> v0=1

    iput-boolean v0, p0, Lcom/apphacking/smalitwo/Player;->state:Z     --->
```

```
state=v0  --> state=1 ---> state=true

    return-void
.end method
```

4- virtual method --> public method on code we create it

```
.method public getHit()V   ---> public void getHit()
    .registers 3

    .line 17
    iget v0, p0, Lcom/apphacking/smalitwo/Player;->lives:I     --> get the
this.live and set it on v0

    const/4 v1, 0x1    ---> set 1 to v1  --> v1=1

    sub-int/2addr v0, v1     --> v0=v0-v1 --->       v0--

    iput v0, p0, Lcom/apphacking/smalitwo/Player;->lives:I  --> put the
value from v0 to this.lives --> this.lives--

    .line 19
    if-lez v0, :cond_b         ---> if <= 0  it true jump to cond_d else
complete        --> v0 <= 0

    .line 21
    iput-boolean v1, p0, Lcom/apphacking/smalitwo/Player;->state:Z    -->
put the value of v1=1 to state --> this.state=1 = true

    goto :goto_e   ---> jump to goto_e

    .line 24
    :cond_b      ---> else
    const/4 v0, 0x0    --> v0=0

    iput-boolean v0, p0, Lcom/apphacking/smalitwo/Player;->state:Z  ---> put
value from v0 to this.state --> this.state=false

    .line 26
    :goto_e
    return-void
.end method
```

28	goto target	Unconditional jump by short offset[2].	28F0 - goto 0005 // -0010 Jumps to current position-16 words (hex 10). 0005 is the label of the target instruction.

3A	if-ltz vx,target	Checks vx and jumps if vx<0[2].	3A00 1600 - if-ltz v0, 002d // +0016 Jumps to the current position+16H words if v0<0. 002d is the label of the target instruction.

هنا بقول ان لو if v0 < 0 كده بشوف الشرط ولوelse بنحط ديtarget: ودي مهناها ان هو jump to target

```
    if-lez v0, :cond_b          ---> if v0<= 0 true jump to cond_d and  and
if false complete

    .line 21
    iput-boolean v1, p0, Lcom/apphacking/smalitwo/Player;->state:Z    -->
put the value of v1=1 to state --> this.state=1 = true

    goto :goto_e   ---> jump to goto_e

    .line 24
    :cond_b      ---> else
    const/4 v0, 0x0    --> v0=0

    iput-boolean v0, p0, Lcom/apphacking/smalitwo/Player;->state:Z  ---> put
value from v0 to this.state --> this.state=false

    .line 26
    :goto_e
if v0 < 0:cond_d  ---> else
:cond_b      ---> else
const/4 v0, 0x0    --> v0=0  ---> else
```

انى واحده وهي target: goto هي ان لو الشرط ده صح ننتقل لل target وخلي بالك في الاتنين لو عاوز تنتقل للحاجة اللي هي مثلا target:
بتحددها في الكود تبقي target:

```
goto :goto_e   --> if ture jump goto_e

:goto_e        --> here goto_e before return-void because when jump it end
the class and return void
return-void
```

خلي بالك هان هو في التحقق من الشرط هو بيعمل عكس java بس بنفس المعني يعني في java كنا بنتحقق ان this.lives> 0 لا اهو بيعمل العكس بس بنفس المعني if-lez v0 فلو فال من او يساوي 0 هيعمل jump to target ولو الشرط false هيكمل

Syntax	Description	
if-eqz vx,target	Equal Zero	if vx == 0
if-nez vx,target	Not Equal Zero	if vx != 0
if-ltz vx,target	Less Than Zero	if vx < 0
if-gez vx,target	Greater Equal Zero	if vx >= 0
if-gtz vx,target	Greater Than Zero	if vx > 0
if-lez vx,target	Less Equal Zero	if vx <= 0

Comparison against 0.

target = Location to JMP after this check.

Syntax	Description	
if-eq vx,vy,target	Equal	if vx == vy
if-ne vx,vy,target	Not Equal	if vx != vy
if-lt vx,vy,target	Less Than	if vx < vy
if-ge vx, vy,target	Greater Equal	if vx >= vy
if-gt vx,vy,target	Greater Than	if vx > vy
if-le vx,vy,target	Less Equal	if vx =< vy

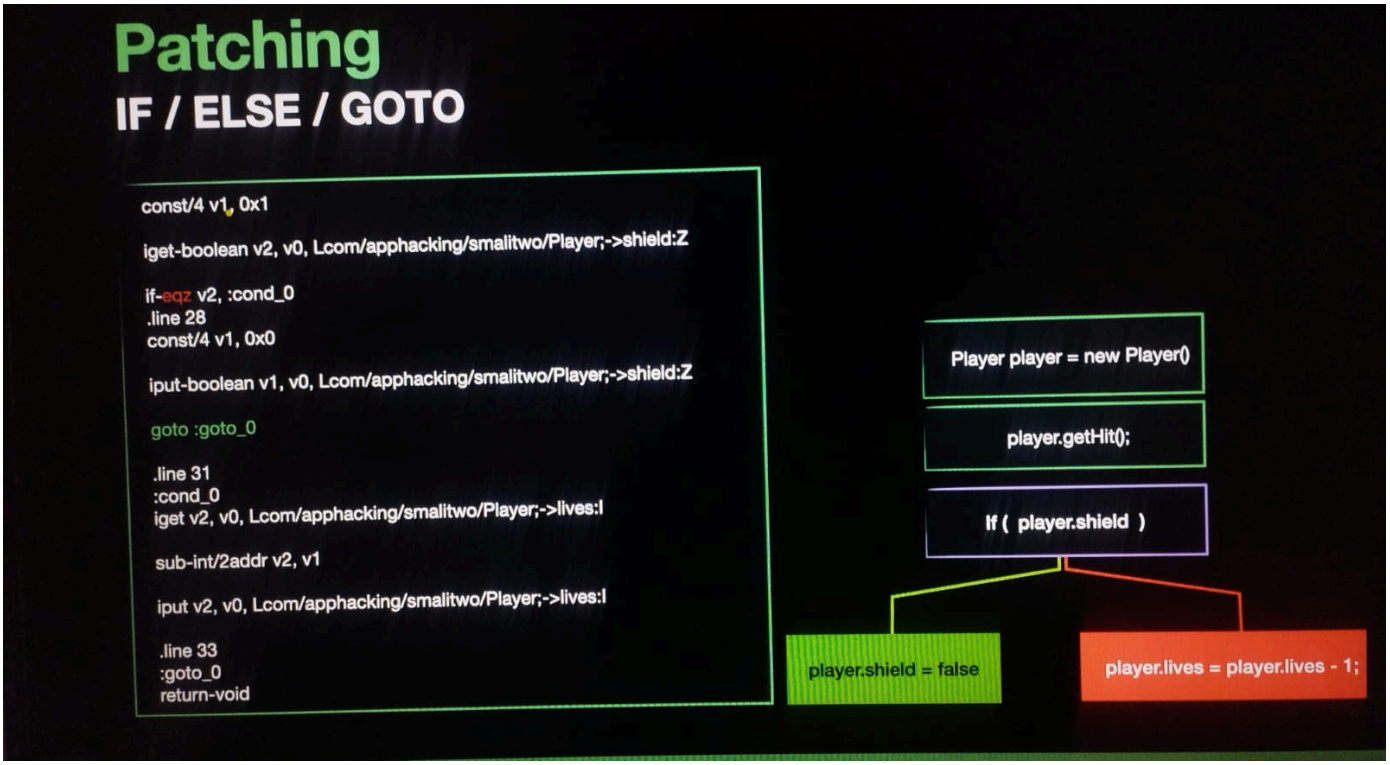
Comparison against a local register.

target = Location to JMP after this check.

68- Flipping the logic

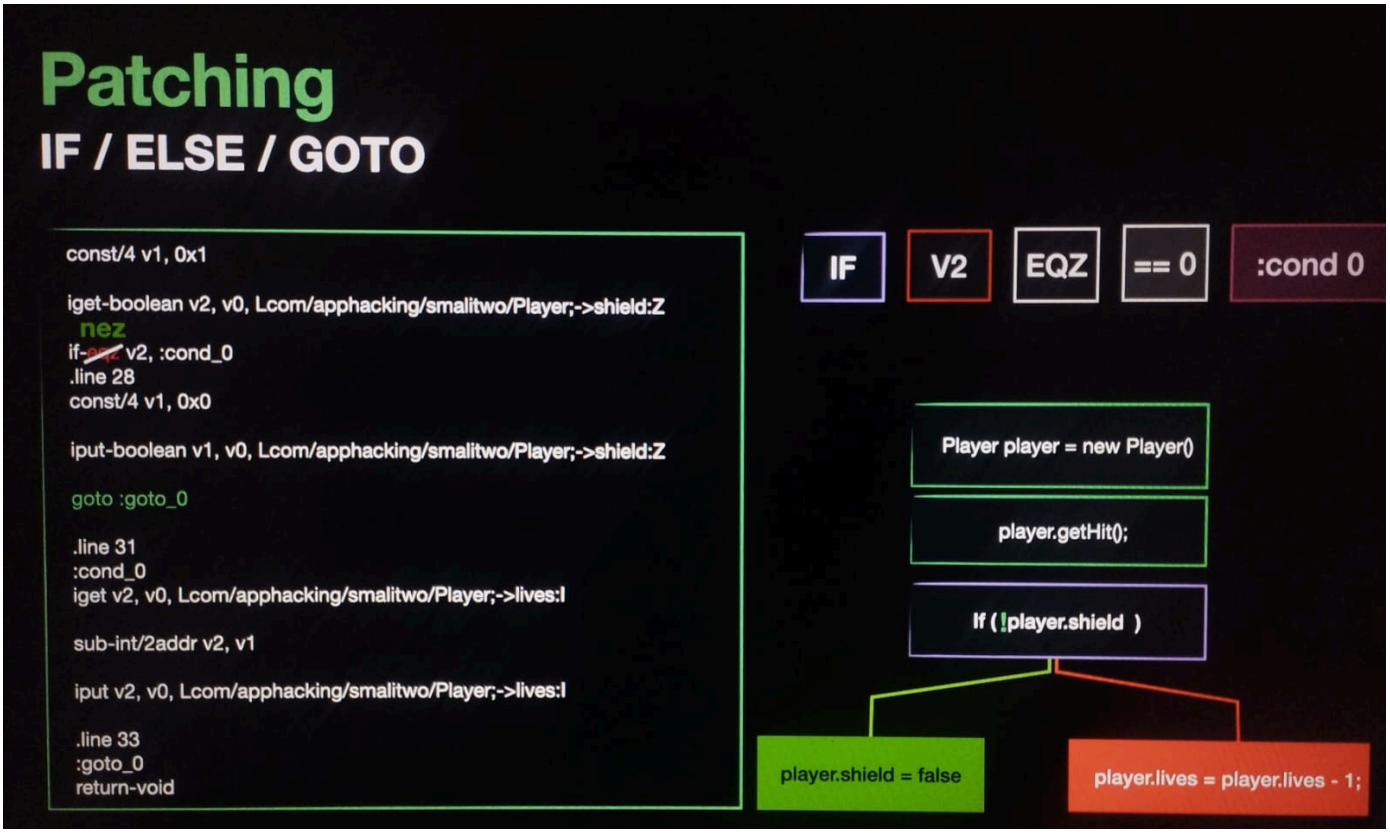
1- method1 logic flipping

هنا مثلا الكود ده ان لو هو لابس درع فالدرع ده هيشيله ولو بس لابس فهو هيموت




```
const/4 v1,0x1 ---> v1 =1
iget-boolean v2,p0,Lcom/apphacking/smalitwo/Player;->shiled:Z   --> v2
=this.shiled
if-eqz v2:cond_0  --> if v2 ==0  jump to cond_0 else complete
.line28
const/4 v1,0x0  --> v1=0
iput-boolean v1,p0,Lcom/apphacking/smalitwo/Player;->shiled:Z   --> put the
value of v1 on this.shiled --> this.shiled=0 = false
goto: goto_0  --> jump goto_0
.line 31
:cond_0  --> the condition    if v2==0
iget v2,p0,Lcom/apphacking/smalitwo/Player;->lives:I  --> get the value from
this.lives and set it into v2 --> v2=this.lives
sub-int/2addr v2,v1 --> v2=v2-v1
iput v2,p0,Lcom/apphacking/smalitwo/Player;->lives:I  --> put the value of
v2 into this.lives
.line 33
:goto_0
return-void
```

علشان نحاول نستغل دي نحول الشرط زي كده مثلا : هنا اه مثلا بعكس الشرط لو انا هبدا مش لابس درع كده هيبقي (false)! كده هيبقي true ويروح ينفذ الشرط انه يشيب درع وهكذا وتبقي infinite loop بس مش هينقص live

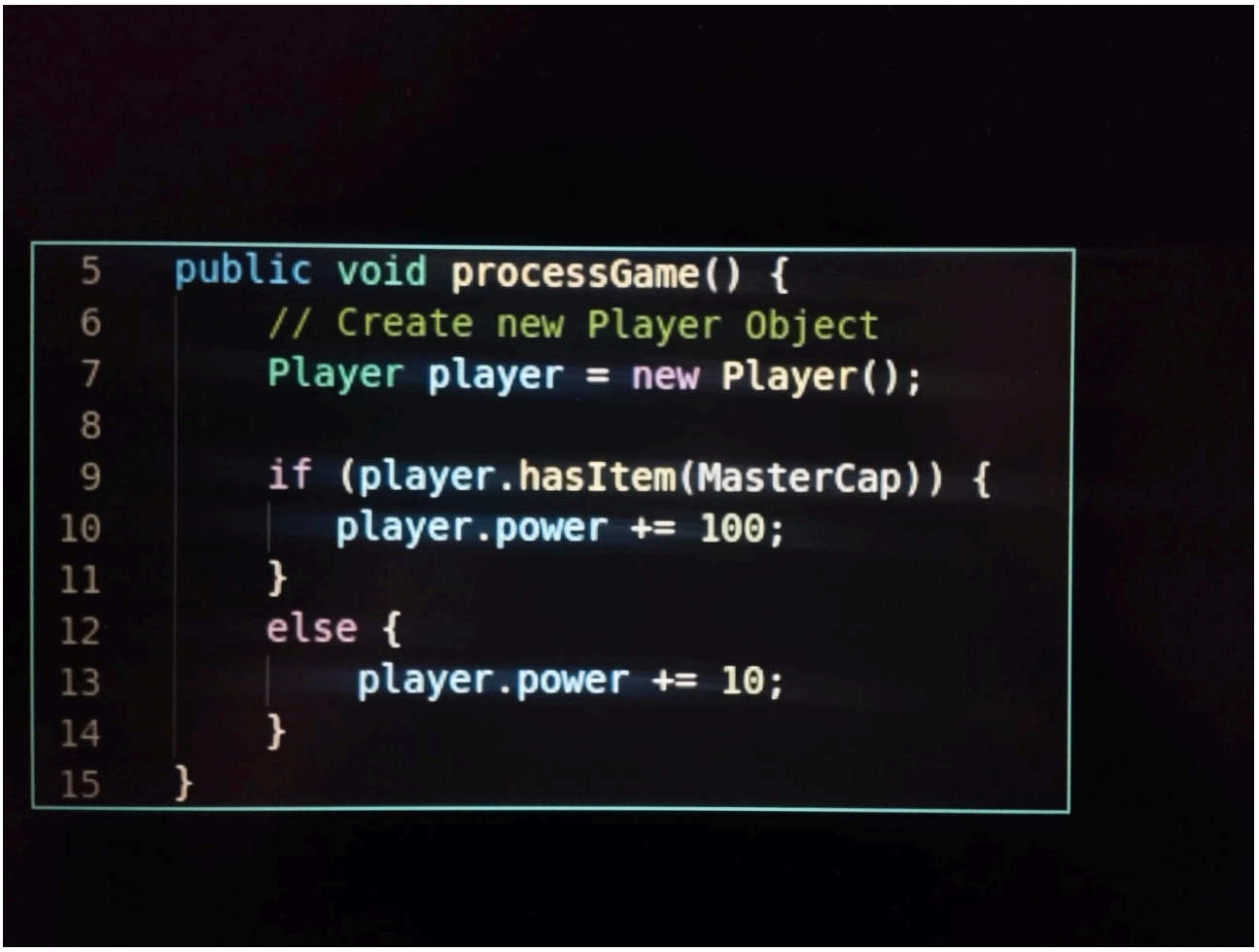


نروح بقي نغير القيمة عادي ل nez في smali code ونعمل built من ونعمل نفس الخطوات اللي نعمل signing وبكده يبقي احنا غيرنا app logic وبكده دي اول طريقة من الطرق اللي من خلاله ازاي نستغل smali code

---

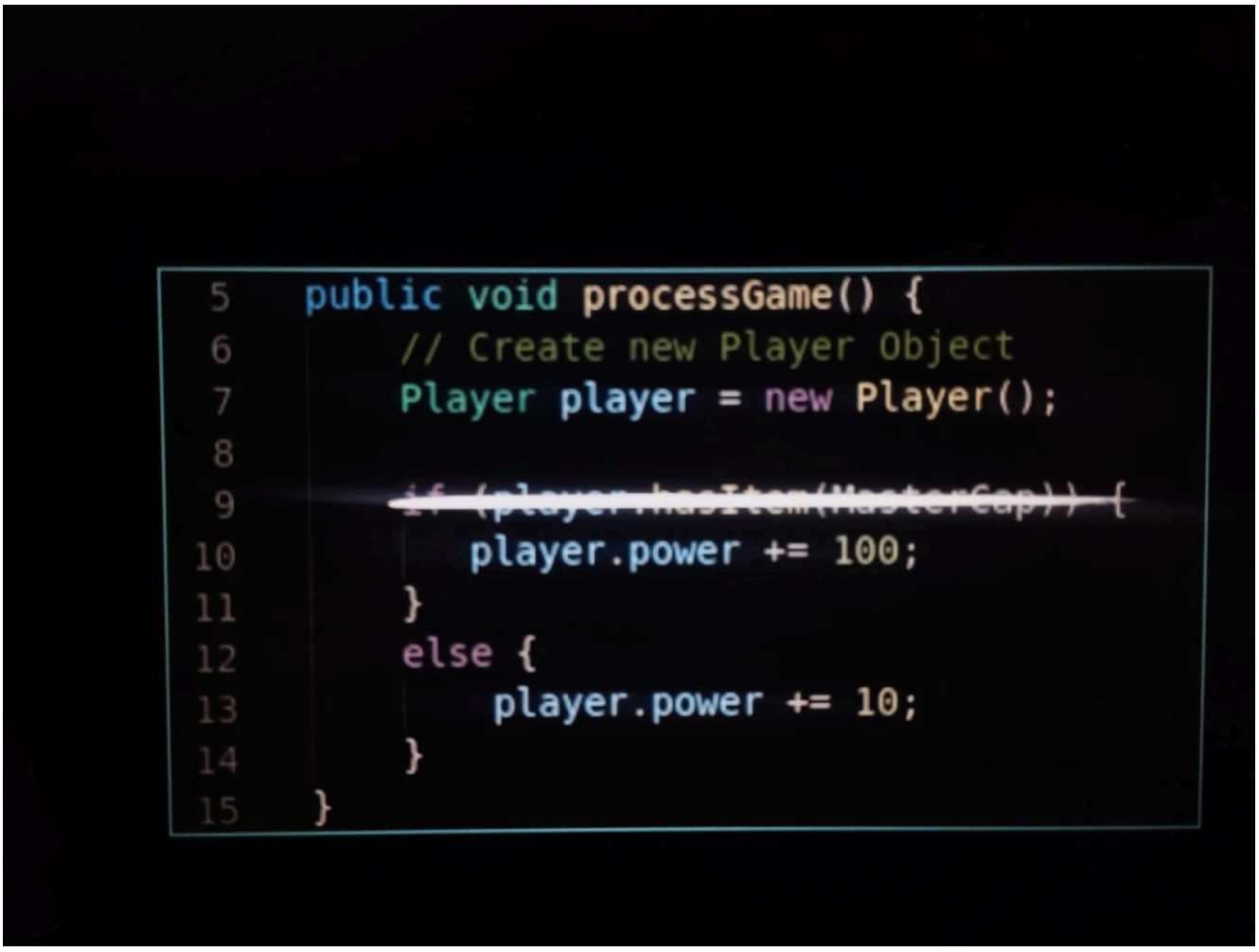
2-methdo 2 Deleting Code

هنا في الكود ده هو بيشوف هل الاعب عند MasterCap ولا لا ولو عنده يزود power ب 100 ولو معندوش يزودها ب 10



```
5   public void processGame() {
6       // Create new Player Object
7       Player player = new Player();
8
9       if (player.hasItem(MasterCap)) {
10          player.power += 100;
11      }
12      else {
13          player.power += 10;
14      }
15  }
```

اللي بقي نعمله هو ان احنا نشيل الشرط بتاع IF ولكن لو شيلنها مش هيبقي logic في java علشان هيبقي في }else{} ولكن في smali هيبقي عادي زي ما هنشوف دلوقتي وبعد ما نعمله هنحصل علي power من غير اي شرط



code java for player

هنا اه هو قيمة masterCap =false فكده هو هيزد الpower ب 10

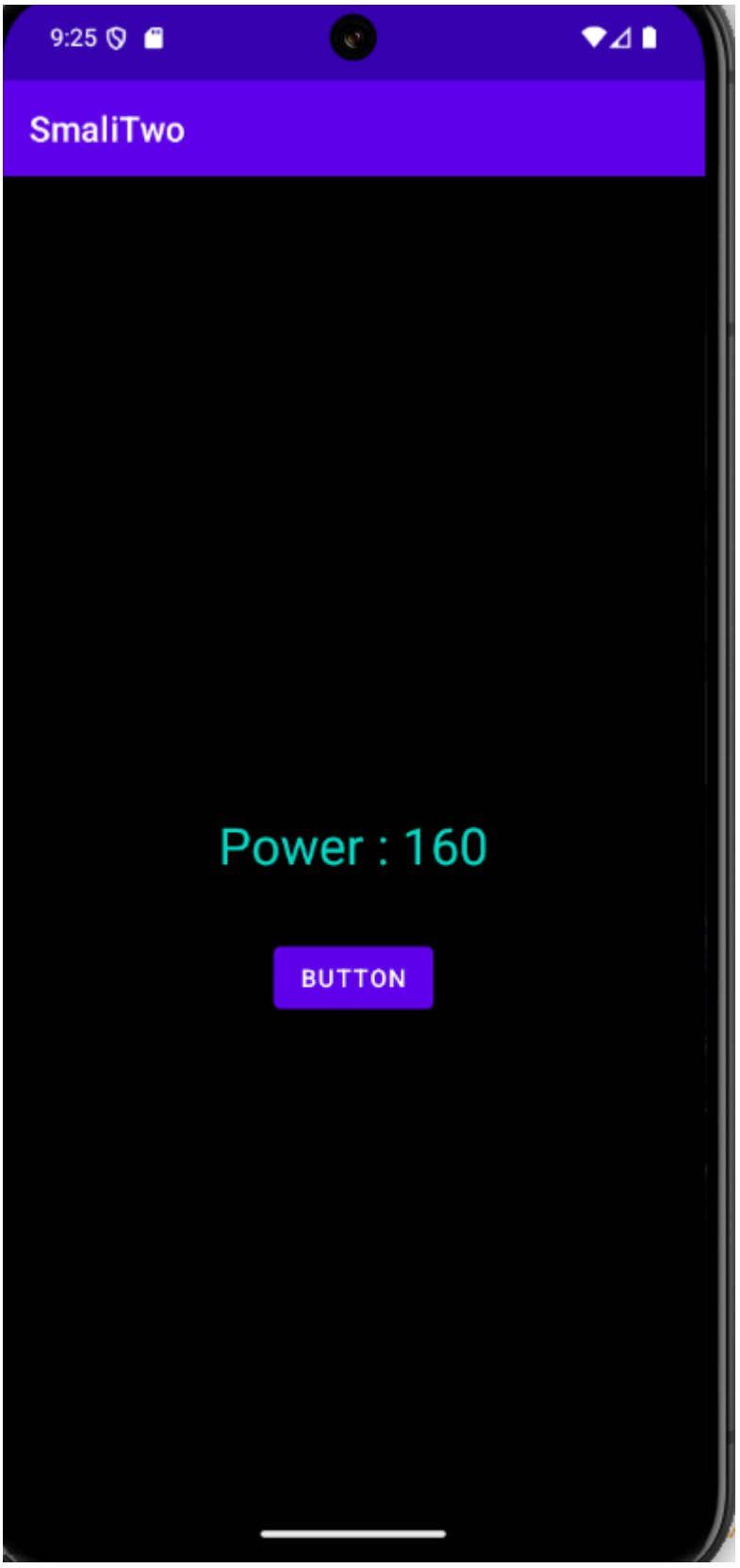
```java
package com.apphacking.smalitwo;

/* loaded from: classes2.dex */
public class Player {
    int highscore;
    int power = 10;
    int lives = 5;
    int score = 100;
    boolean shield = true;
    boolean state = true;
    boolean masterCap = false;

    public void hasItem(Player player) {
        if (player.masterCap) {
            player.power += 100;
        } else {
            player.power += 10;
        }
    }

    public void checkHighScore(Player player)
        int i = player.score;
        if (i < player.highscore) {
            return;
        }
        player.highscore = i;
        if (player.lives >= 5) {
            player.score = i + 1000;
        }
    }

    public void getHit() {
        if (this.shield) {
            this.shield = false;
            return;
        }
        int i = this.lives - 1;
        this.lives = i;
        if (i > 0) {
            this.state = true;
        } else {
            this.state = false;
        }
    }
}
```

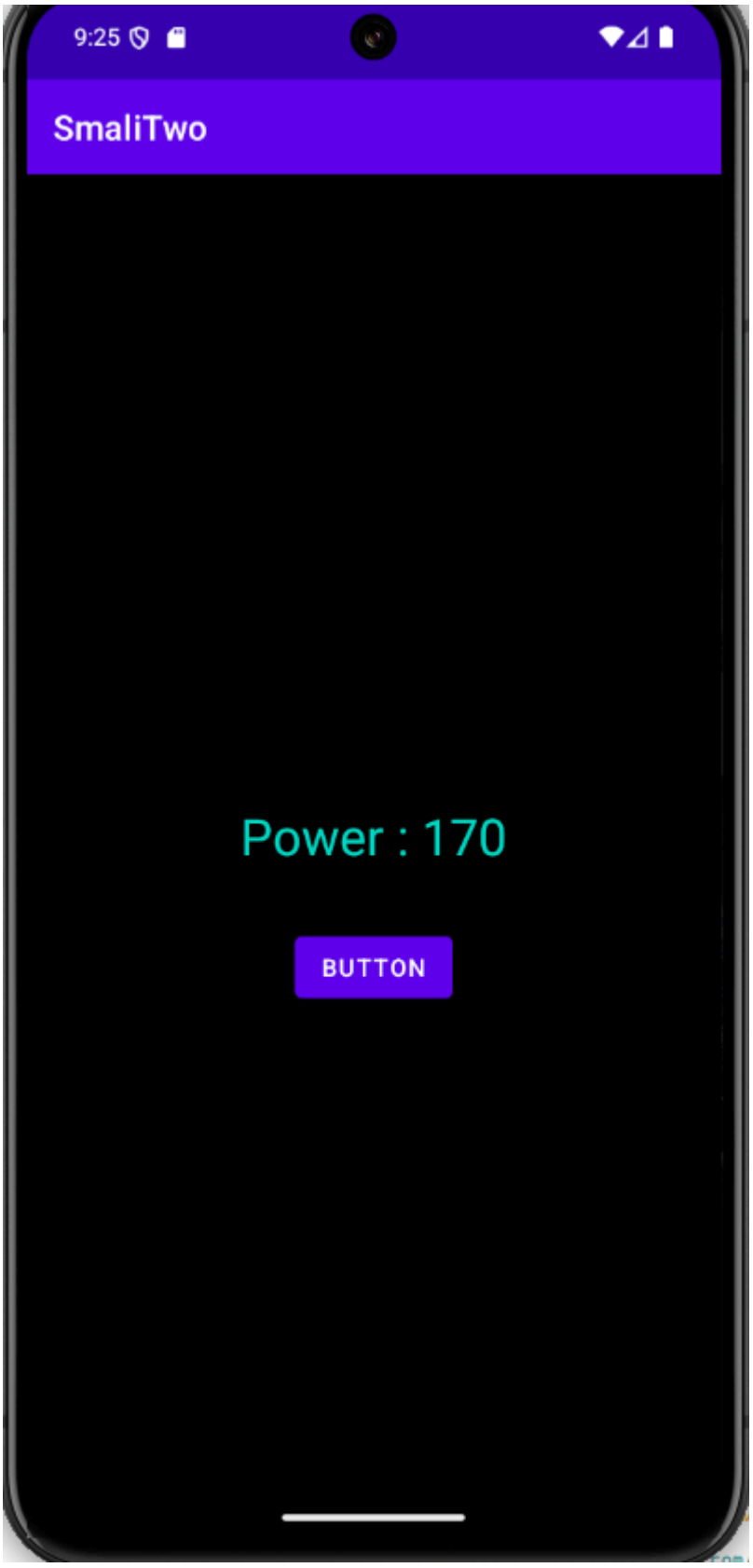
```
.method public hasItem(Lcom/apphacking/smalitwo/Player; )V
    .registers 3
    .param p1, "player"    # Lcom/apphacking/smalitwo/Player;

    .line 20
    iget-boolean v0, p1, Lcom/apphacking/smalitwo/Player; ->masterCap:Z

    if-eqz v0, :cond_b

    .line 22
    iget v0, p1, Lcom/apphacking/smalitwo/Player; ->power:I

    add-int/lit8 v0, v0, 0x64

    iput v0, p1, Lcom/apphacking/smalitwo/Player; ->power:I

    goto :goto_11

    .line 25
    :cond_b
    iget v0, p1, Lcom/apphacking/smalitwo/Player; ->power:I

    add-int/lit8 v0, v0, 0xa

    iput v0, p1, Lcom/apphacking/smalitwo/Player; ->power:I

    .line 28
    :goto_11
    return-void
.end method
```

هنا بقي في smali لو حزفنا if-eqz v0,:cond_b هيكمل حاجة من غير ما يتحقق من حاجة ويزودها ب 100

```
if-eqz v0, :cond_b    --> if we remove this line
```

هنا اه بيزود ب 10

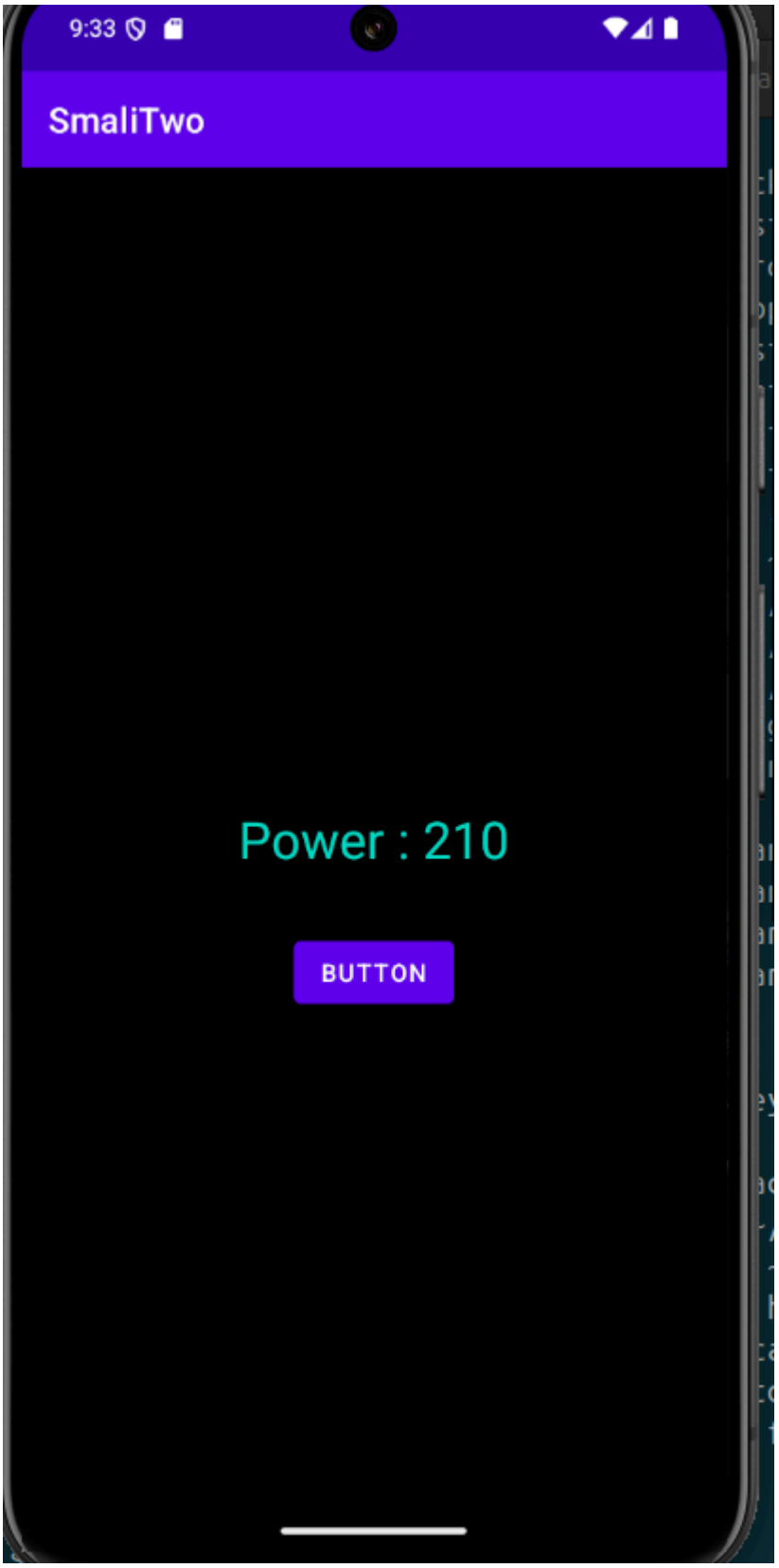

9:25
SmaliTwo
Power : 160
BUTTON

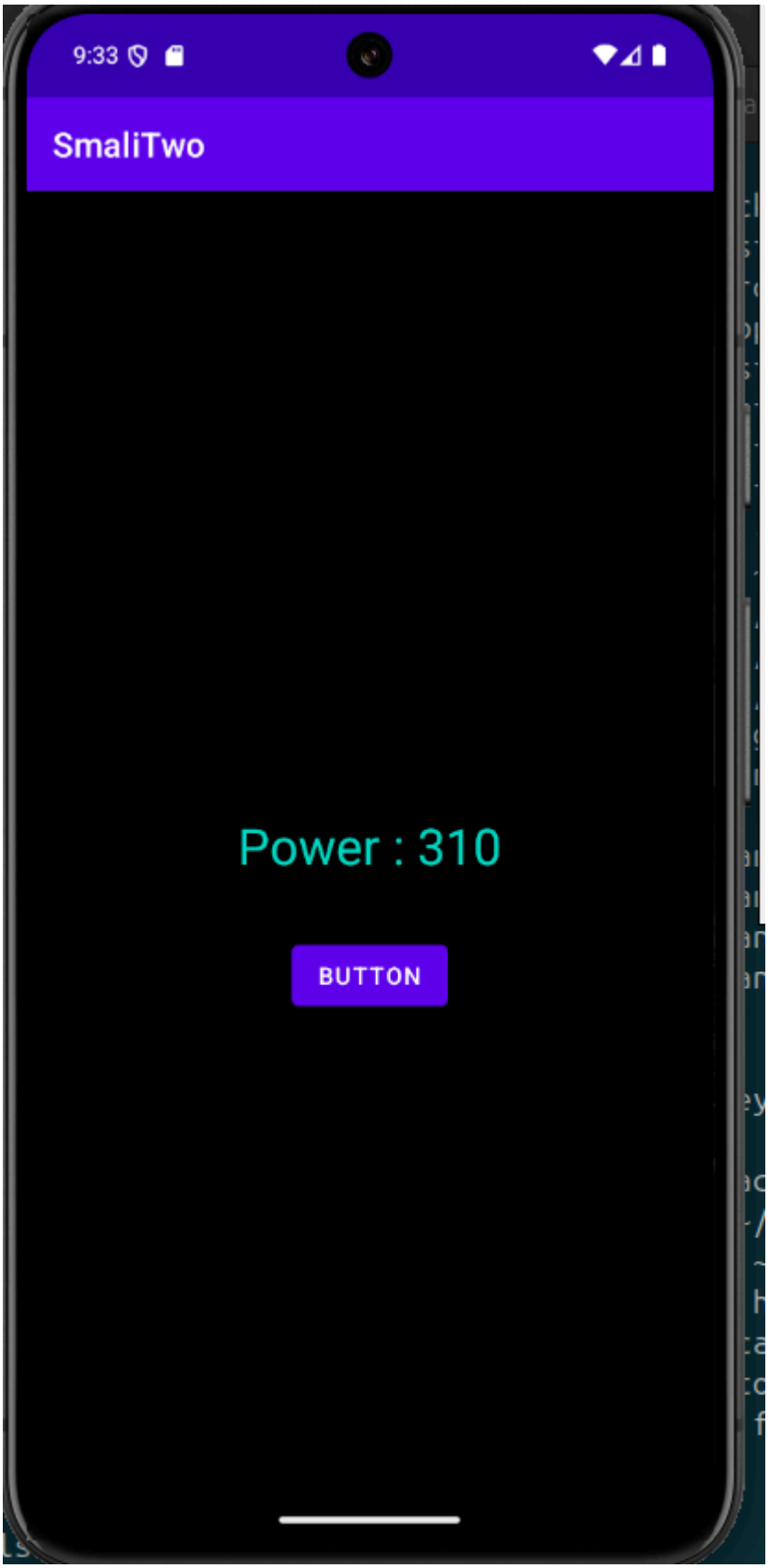


دلوقتي هنحذف الامر بتاع if statement on smali

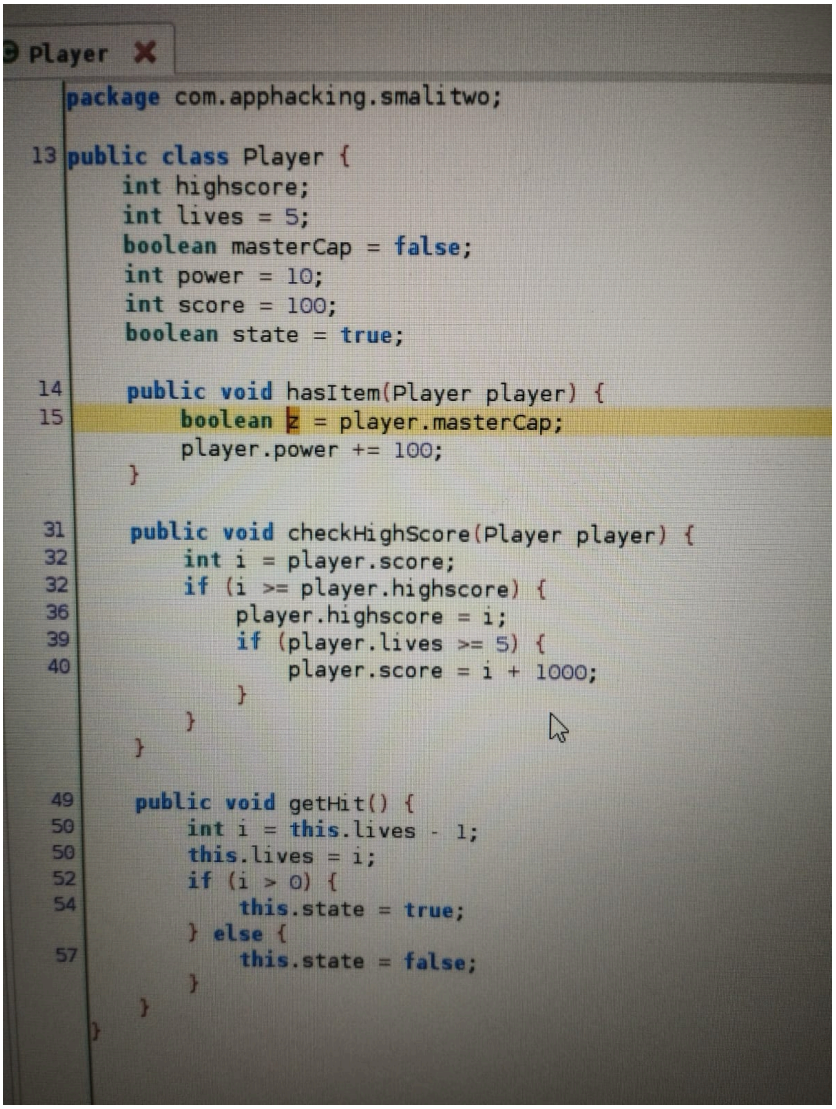
```
.method public hasItem(Lcom/apphacking/smalitwo/Player;)V
    .locals 1
    .param p1, "player"    # Lcom/apphacking/smalitwo/Player;

    .line 20
    iget-boolean v0, p1, Lcom/apphacking/smalitwo/Player;->masterCap:Z


    .line 22
    iget v0, p1, Lcom/apphacking/smalitwo/Player;->power:I

    add-int/lit8 v0, v0, 0x64

    iput v0, p1, Lcom/apphacking/smalitwo/Player;->power:I

    goto :goto_0

    .line 25
    :cond_0
    iget v0, p1, Lcom/apphacking/smalitwo/Player;->power:I

    add-int/lit8 v0, v0, 0xa

    iput v0, p1, Lcom/apphacking/smalitwo/Player;->power:I

    .line 28
    :goto_0
    return-void
.end method
```

نعمل build بقي و signing ونحمله ونشوف ودلوقتي انا هعمله كده وهحمله ودلوقتي انا عملته sign وبعد ما حذفنا السطر فعلا بقي يزود القمية بتاعت power ب 100


9:33
SmaliTwo
Power : 210
BUTTON


9:33
SmaliTwo
Power : 310
BUTTON

او برضه ممكن نحذا اي شرط ونخليه يزود وخلاص المهم هنا هو بنعدل في الكود علشان نغير function اللي هو بيعملها

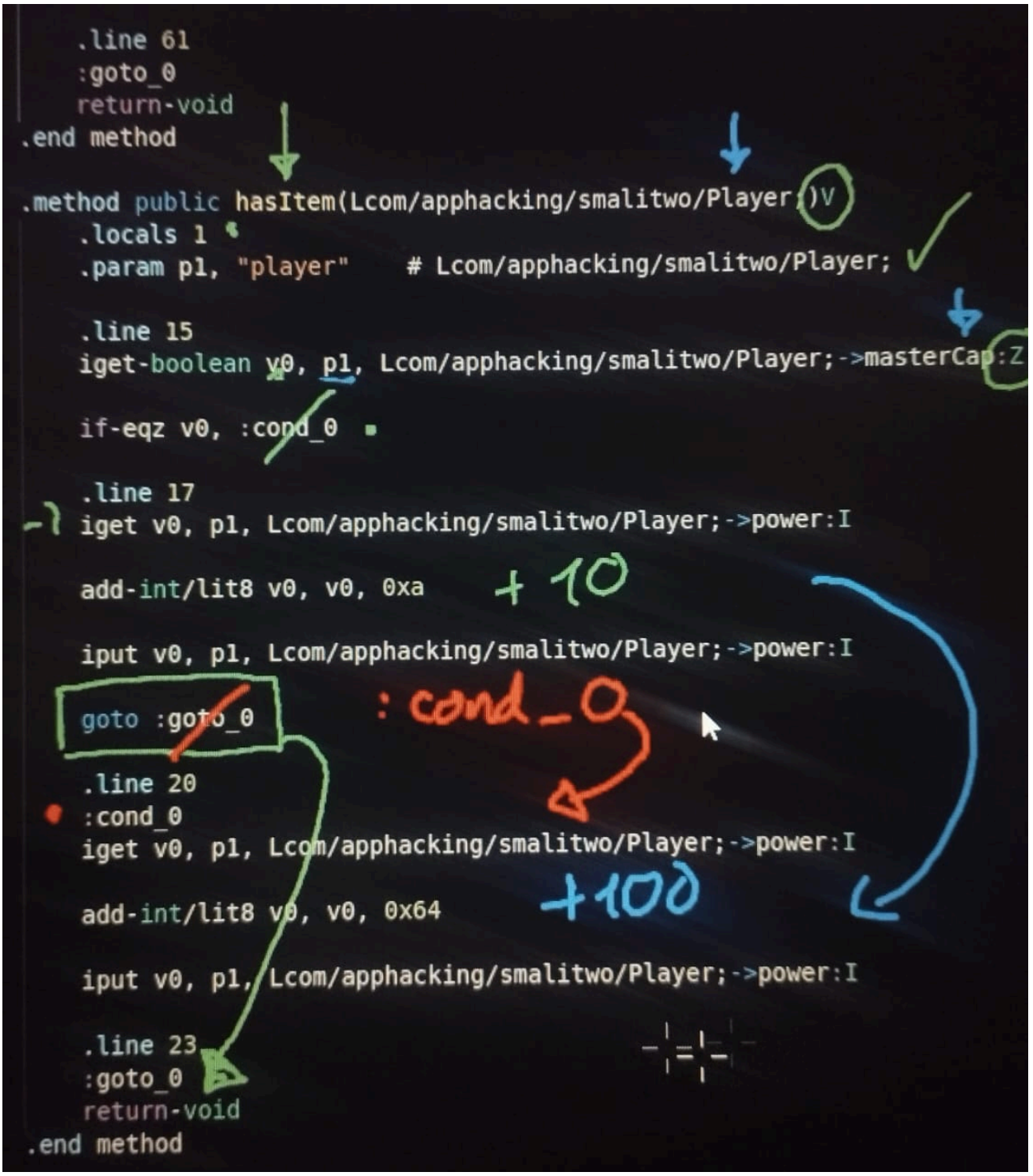


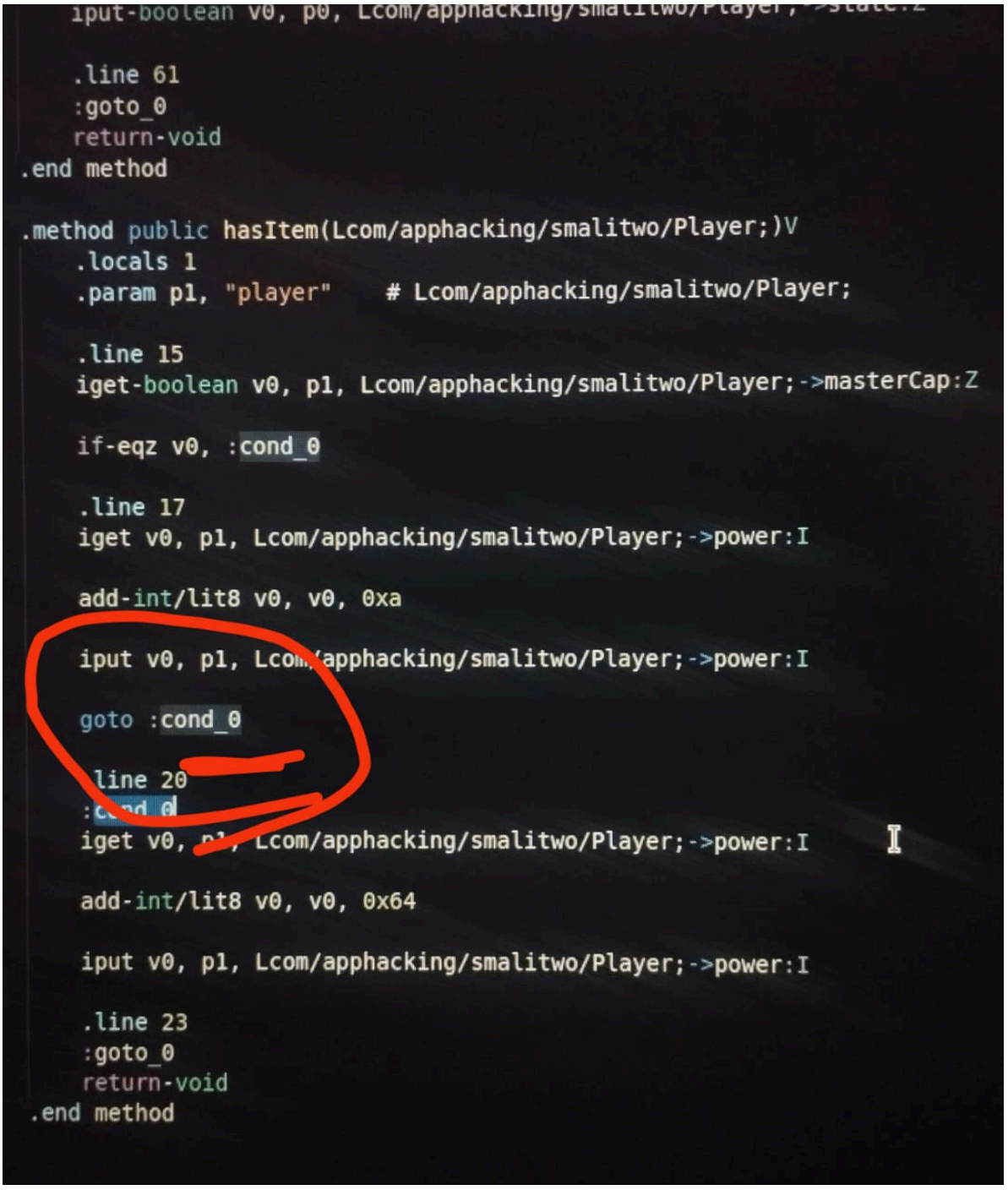
---

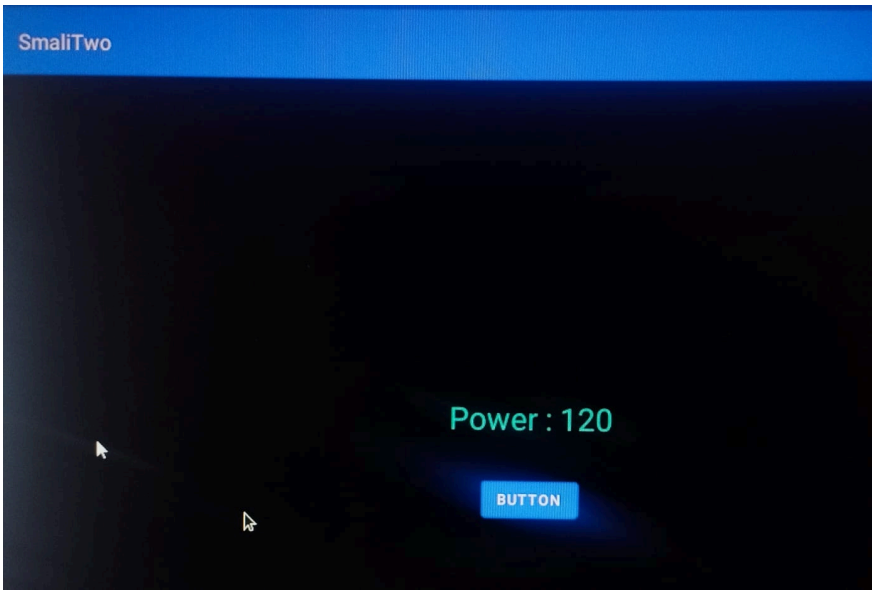
70 -Method 3 : Jumps

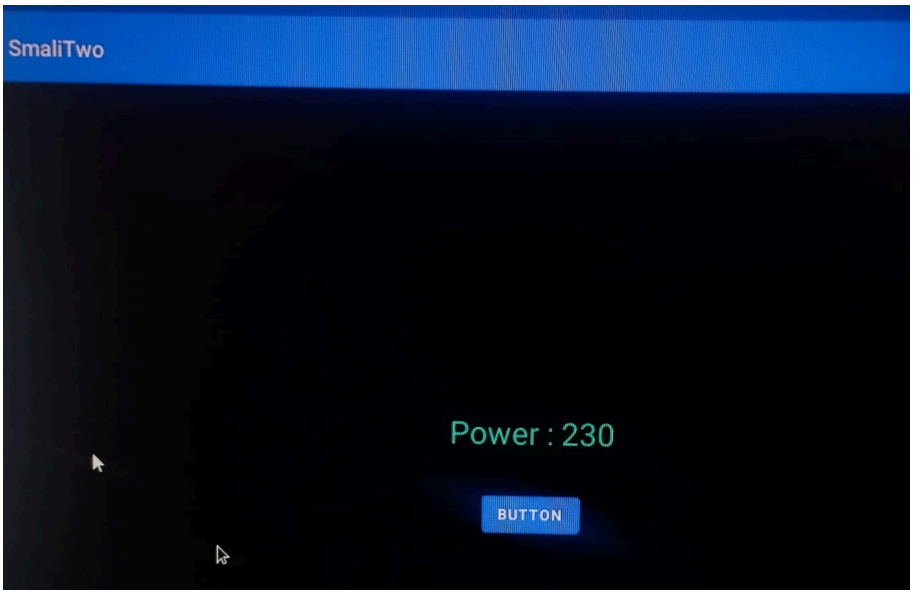
هنا ان احنا نعمل jump علي command تاني باستخدام goto فهنا لما نتسخدمها مثلا الشرط هيتحقق ونزود power ب 10 وفي نفس scope بتاع الشرط نعمل jump للشرط التاني اللي هو بيزود power ب 100 وبكده يبقي في كل مرة هنزودها بقيمتين هما 10 و 100

```
goto: target
```


.line 61
:goto_0
return-void
.end method
.method public hasItem(Lcom/apphacking/smalitwo/Player;)V
.locals 1
.param p1, "player" # Lcom/apphacking/smalitwo/Player;
.line 15
iget-boolean v0, p1, Lcom/apphacking/smalitwo/Player;->masterCap:Z
if-eqz v0, :cond_0
.line 17
iget v0, p1, Lcom/apphacking/smalitwo/Player;->power:I
add-int/lit8 v0, v0, 0xa
+10
iput v0, p1, Lcom/apphacking/smalitwo/Player;->power:I
goto :goto_0
:cond_0
.line 20
:cond_0
iget v0, p1, Lcom/apphacking/smalitwo/Player;->power:I
add-int/lit8 v0, v0, 0x64
+100
iput v0, p1, Lcom/apphacking/smalitwo/Player;->power:I
.line 23
:goto_0
return-void
.end method

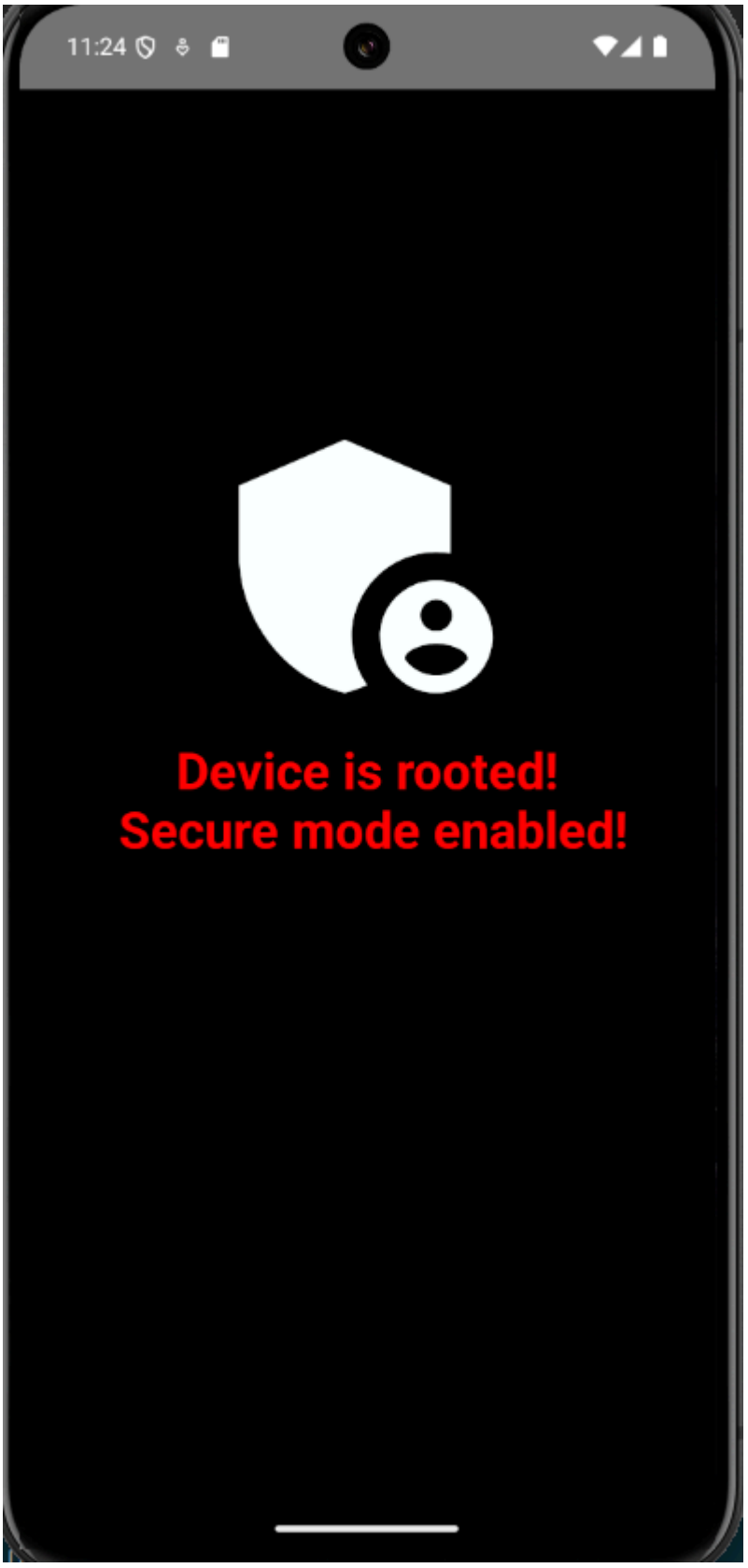

iput-boolean v0, p0, Lcom/apphacking/smalitwo/Player;
.line 61
:goto_0
return-void
.end method
.method public hasItem(Lcom/apphacking/smalitwo/Player;)V
.locals 1
.param p1, "player" # Lcom/apphacking/smalitwo/Player;
.line 15
iget-boolean v0, p1, Lcom/apphacking/smalitwo/Player;->masterCap:Z
if-eqz v0, :cond_0
.line 17
iget v0, p1, Lcom/apphacking/smalitwo/Player;->power:I
add-int/lit8 v0, v0, 0xa
iput v0, p1, Lcom/apphacking/smalitwo/Player;->power:I
goto :cond_0
.line 20
iget v0, p1, Lcom/apphacking/smalitwo/Player;->power:I
add-int/lit8 v0, v0, 0x64
iput v0, p1, Lcom/apphacking/smalitwo/Player;->power:I
.line 23
:goto_0
return-void
.end method





Assignemnt

هنا في chellange ده هو عاوزيني اعمل bypass root detectoin



1- decompile the app and analyze it

هنا بقي هنشوف في Main activity هو بيعمل Check ان هو root من خلال function في RootUtil

```
        if (RootUtil.isDeviceRooted()) {
            this.btn.setVisibility(4);
            this.txtview.setTextColor(Color.parseColor("#FF0000"));
            this.txtview.setText("Device is rooted!\n Secure mode enabled!");
        } else {
            this.btn.setVisibility(0);
            this.txtview.setTextColor(Color.parseColor("#00FF00"));
            this.txtview.setText("Device is NOT rooted!\n No login possible!");
        }
    }

    public void login(View view) {
        Intent intent = new Intent(this, (Class<?>) Login.class);
        startActivity(intent);
    }
```

2- open file RootUtil

هنا اه هو check root من خلال three method كل واحدة بترجع true

```
package com.apphacking.rootcheck;

import android.os.Build;
import java.io.BufferedReader;
import java.io.File;
import java.io.InputStreamReader;

/* loaded from: classes.dex */
public class RootUtil {
    public static boolean isDeviceRooted() {
        return checkRootMethod1() || checkRootMethod2() || checkRootMethod3();
    }

    private static boolean checkRootMethod1() {
        String buildTags = Build.TAGS;
        return buildTags != null && buildTags.contains("test-keys");
    }

    private static boolean checkRootMethod2() {
        String[] paths = {"/system/app/Superuser.apk", "/sbin/su", "/system/bin/su", "/system/xbin/su", "/data/local/xbin/su", "/data/local/bin/su", "/system/sd/xb
        for (String path : paths) {
            if (new File(path).exists()) {
                return true;
            }
        }
        return false;
    }

    private static boolean checkRootMethod3() {
        Process process = null;
        try {
            process = Runtime.getRuntime().exec(new String[]{"/system/xbin/which", "su"});
            BufferedReader in = new BufferedReader(new InputStreamReader(process.getInputStream()));
            if (in.readLine() != null) {
                if (process != null) {
                    process.destroy();
                }
                return true;
            }
            if (process != null) {
                process.destroy();
            }
            return false;
        } catch (Throwable th) {
            if (process != null) {
                process.destroy();
            }
            return false;
        }
    }
}
```

علشان بقي منخلهوش ان check root ممكن نعمل two scenario هو نعكس condition بتاع root في RootUtil والسناريو التاني ان نحاول نغير في RootUtil بحيث نخلي three method return false

1- بص اللي هنعمله هو ان احنا في condition بتاع RootUtil اللي هو check isRooted نعكس بقي condition

```
public class RootUtil {
    public static boolean isDeviceRooted() {
        return checkRootMethod1() || checkRootMethod2() || checkRootMethod3();
    }
```

في كود smali نحاول ان هو return false علشان لما بروح لل Main Activitiy يبقي قيمة ان هو root =false وبالتالي يبقي عرفنا نعمل bypass root detection

smali code for this function

دلوقتي هنا اه هو بيروح يجيب الحاجة اللي بنعملها return من كل method وطبعا علشان موجودة هتبقي true فلما تبقي true يعمل jump to cond_1 اللي هو بيرجع true

```
.method public static isDeviceRooted()Z
    .locals 1

    .line 10
    invoke-static {}, Lcom/apphacking/rootcheck/RootUtil;->checkRootMethod1()Z

    move-result v0

    if-nez v0, :cond_1

    invoke-static {}, Lcom/apphacking/rootcheck/RootUtil;->checkRootMethod2()Z

    move-result v0

    if-nez v0, :cond_1

    invoke-static {}, Lcom/apphacking/rootcheck/RootUtil;->checkRootMethod3()Z

    move-result v0

    if-eqz v0, :cond_0

    goto :goto_0

    :cond_0
    const/4 v0, 0x0

    goto :goto_1

    :cond_1
    :goto_0
```

```
    const/4 v0, 0x1

    :goto_1
    return v0
.end method
```

to solving this

في حلين بقي اول حل هو ان نخليه يعمل jump to cond_0 من غير ما يخش عليها اصلا زي كده هيروح ل cond_0 و return false

```
.method public static isDeviceRooted()Z
    .locals 1
    goto :cond_0
    .line 10
    invoke-static {}, Lcom/apphacking/rootcheck/RootUtil;-
>checkRootMethod1()Z

    move-result v0

    if-nez v0, :cond_1

    invoke-static {}, Lcom/apphacking/rootcheck/RootUtil;-
>checkRootMethod2()Z

    move-result v0

    if-nez v0, :cond_1

    invoke-static {}, Lcom/apphacking/rootcheck/RootUtil;-
>checkRootMethod3()Z

    move-result v0

    if-eqz v0, :cond_0

    goto :goto_0

    :cond_0
    const/4 v0, 0x0

    goto :goto_1

    :cond_1
    :goto_0
```

```
    const/4 v0, 0x1

    :goto_1
    return v0
```

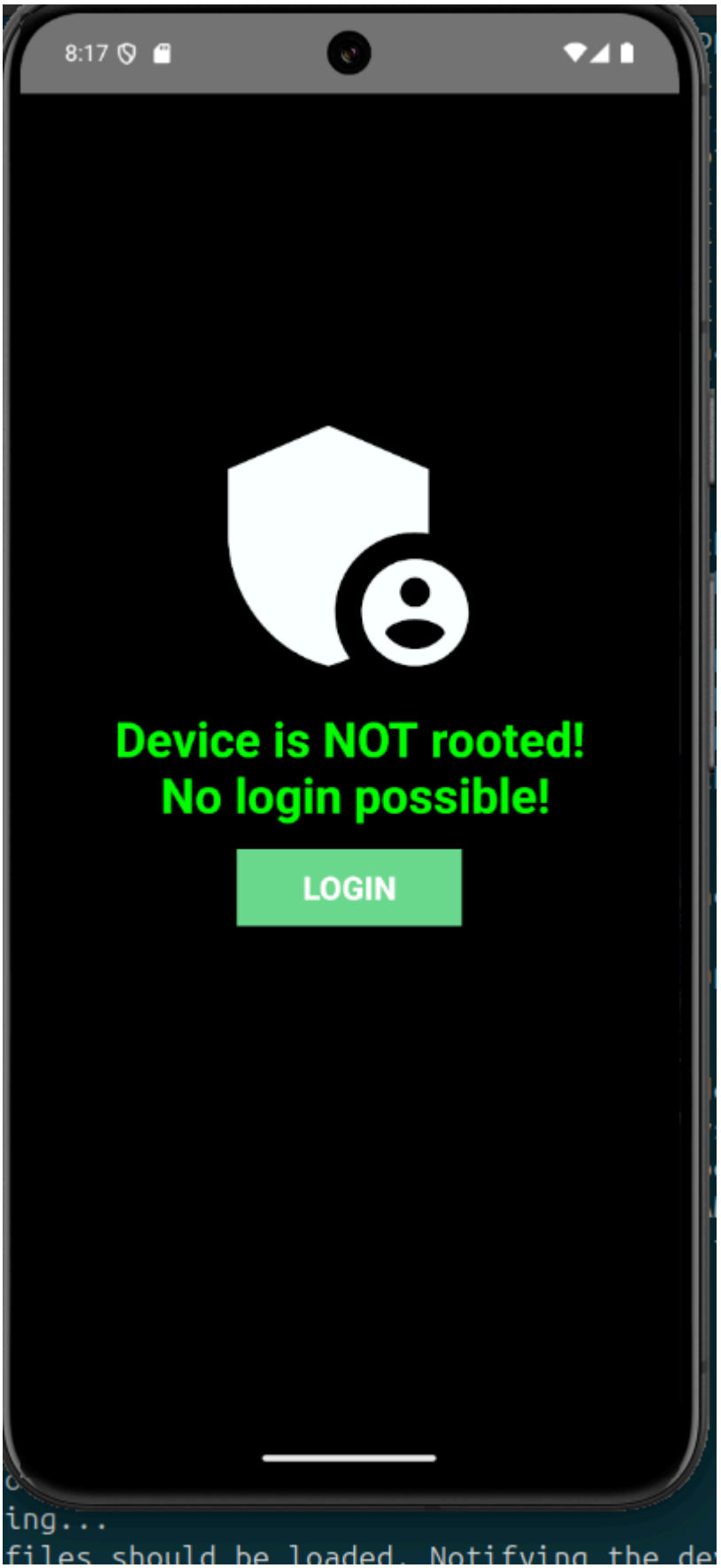
الحل التاني هو ان نعدل condition نغير logic بتاعه بدل ما يكون if_nez v0 :cond_1 نخليها if_eqz v0 :cond_1 ففي كل مرة هيلاقي ان الشرط غلط فمش هيعمل jump to cond_1 ونعكس برده دي if-eqz v0, :cond_0 نخليها if-nqz v0, :cond_0 علشان كده هيبقي في v0 contain value فلما يلاقي فعلا ان هي مش بتساوي zero يعمل jump to cond_0 والكود هيبقي كده

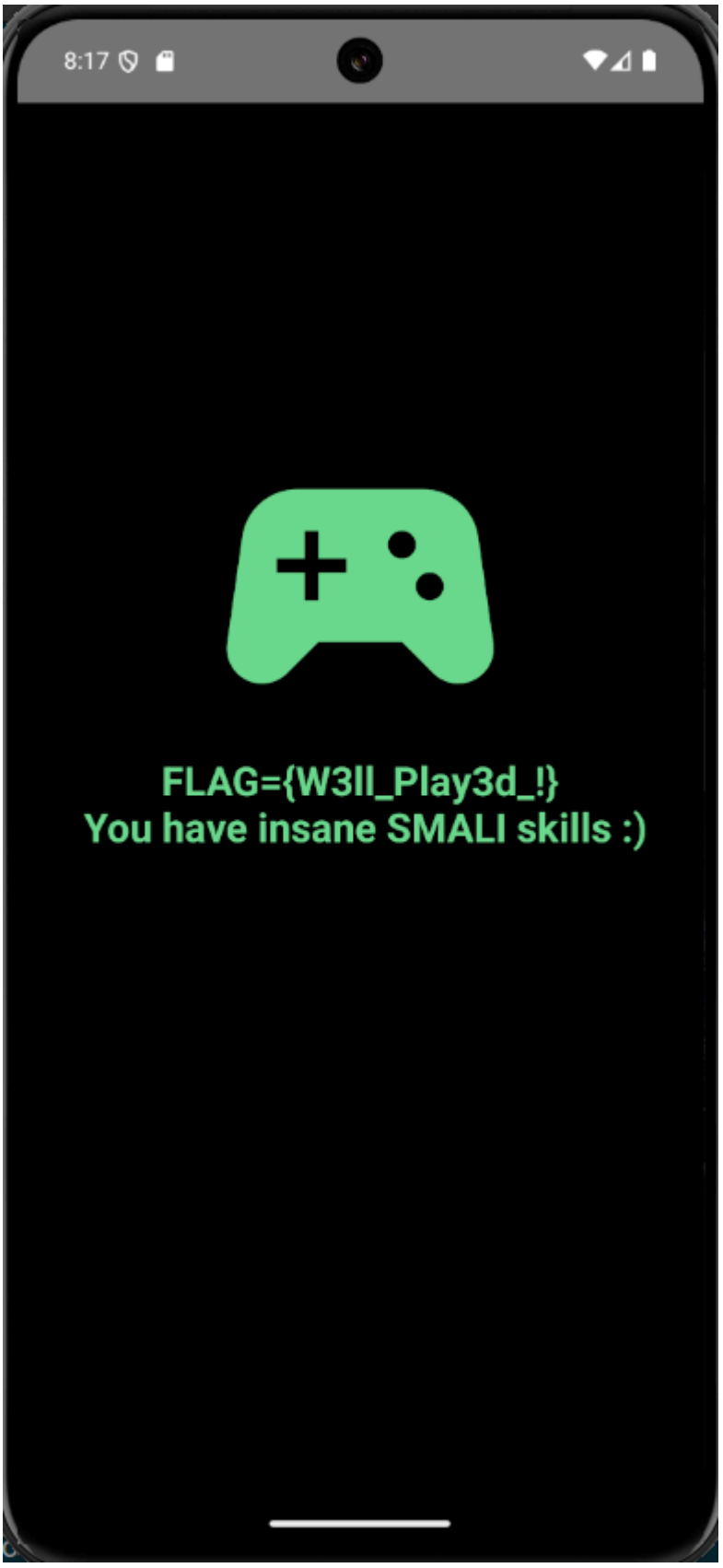
```
.method public static isDeviceRooted()Z
    .locals 1

    .line 10
    invoke-static {}, Lcom/apphacking/rootcheck/RootUtil;-
>checkRootMethod1()Z

    move-result v0

    if-eqz v0, :cond_1

    invoke-static {}, Lcom/apphacking/rootcheck/RootUtil;-
>checkRootMethod2()Z

    move-result v0

    if-eqz v0, :cond_1

    invoke-static {}, Lcom/apphacking/rootcheck/RootUtil;-
>checkRootMethod3()Z

    move-result v0

    if-nqz v0, :cond_0

    goto :goto_0

    :cond_0
    const/4 v0, 0x0

    goto :goto_1

    :cond_1
    :goto_0
```

```
    const/4 v0, 0x1

    :goto_1
    return v0
```

هنغير بقي و نعمل signing ونشوف ايه اللي هيحصل : خلي بالم لما نعمله signing احنا بنحذف التطبيق القديم قبل ما نعمل install للتطبيق الجديد

successfully we change it


8:17
Device is NOT rooted!
No login possible!
LOGIN
ing...



في اكثر من حل ولكن ده اسهل حل

خلي بالك كان في حل ممكن نعمله بقي من غير ما نغير اي حاجة وهو ان start activity for login activity وده علشان لو منا لاقينا ان login activity have one of this option 1- exported="true" or have <intent-filter>f ولكن هن اه في الصورة ملهاش اي

حاجة من اللي احنا قولناه

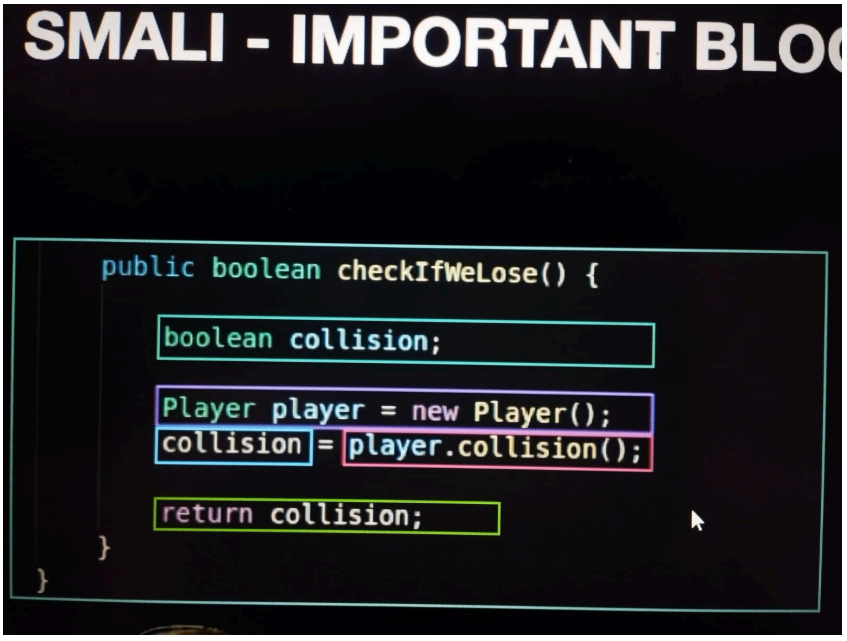
```
        android:roundIcon="@mipmap/ic_launcher_round"
        android:appComponentFactory="androidx.core.app.CoreComponentFactory">
20      <activity android:name="com.apphacking.rootcheck.Login"/>
21      <activity android:name="com.apphacking.rootcheck.MainActivity">
22          <intent-filter>
23              <action android:name="android.intent.action.MAIN"/>
25              <category android:name="android.intent.category.LAUNCHER"/>
22          </intent-filter>
21      </activity>
11  </application>
2 </manifest>
```

73-Object and Method

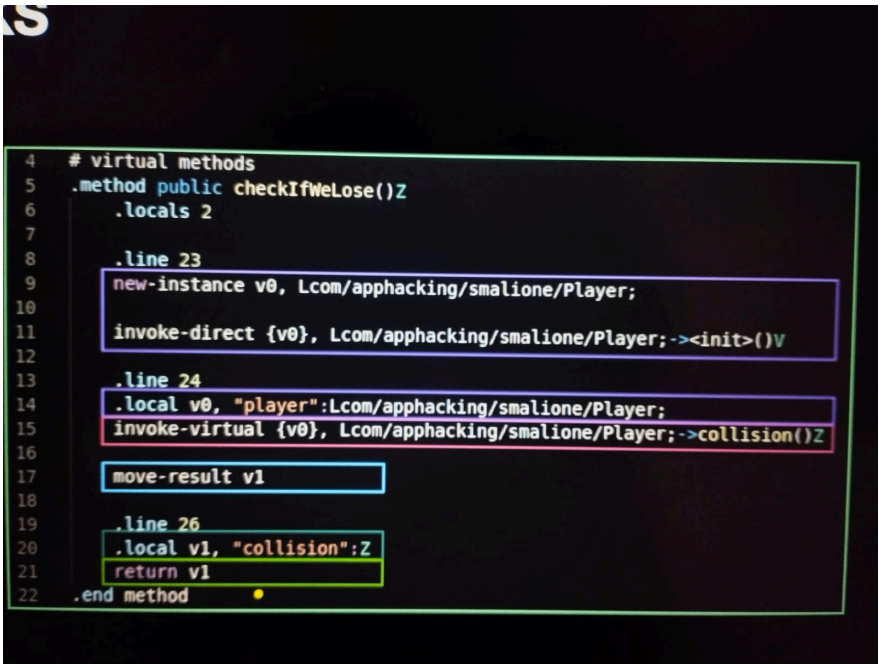
هو ان انتا بتعمل create object من class معين زي مثلا كده

هنا اه هو بيعمل create object player from class Player and assign value collision on object to collision

java code



smali code



هنا اه استخدم new-instance علشان بعرف class اللي هو Player ويبقي في v0 وبعد كده استخدم invoke-direct وخلاه <init> ان هو بينشء حاجة initial وبعد كده في السطر 24 خلي ان القيمة اللي انشئها اللي هي <init> يساويها لل player وبعد كده استخدم invoke-virual علشان ستدعي قمية من object اللي هي collision وبعد كده استخدم move-result v1 عيني الاحجة اللي هتجبها من object تحطها في v1 وبعد كده خد القيمة اللي في v1 وخلاها تبقي collision وتبقي هتبقي boolean وف يالاخر عمل return v1 اللي هو هيعمل return collision

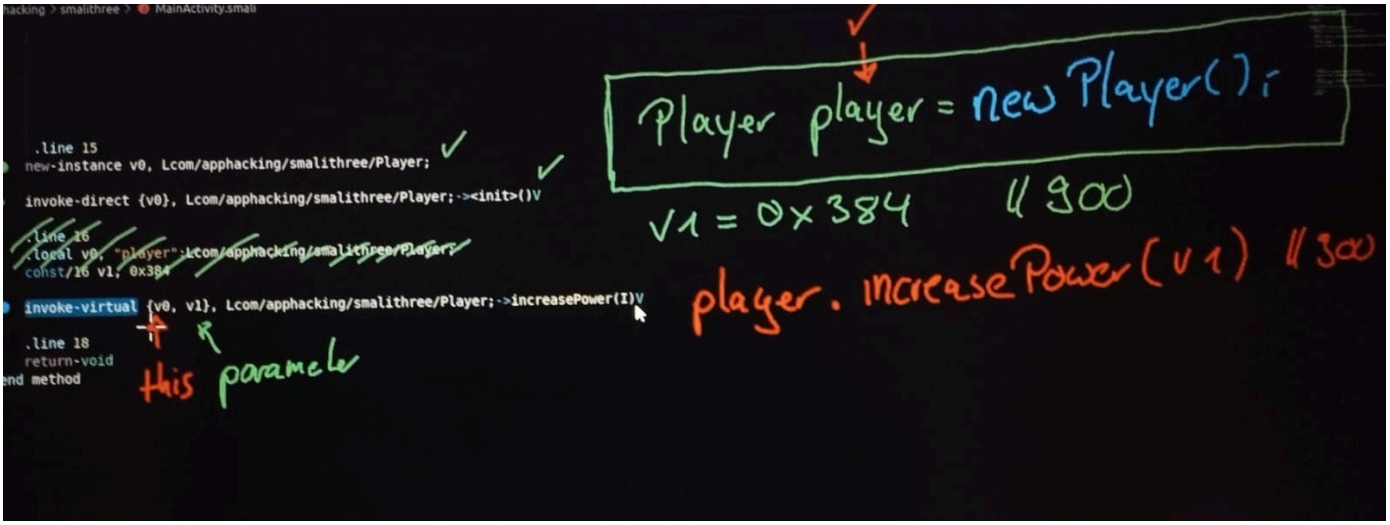
22	new-instance vx,type	Instantiates an object type and puts the reference of the newly created instance into vx.	2200 1500 - new-instance v0, java.io.FileInputStream // type@0015 Instantiates type@0015 (entry #15H in the type table) and puts its reference into v0.

invoke-virtual { parameters }, methodtocall	Invokes a virtual method with parameters.	6E53 0600 0421 - invoke-virtual { v4, v0, v1, v2, v3}, Test2.method5:(IIII)V // method@0006 Invokes the 6th method in the method table with the following arguments: v4 is the "this" instance, v0, v1, v2, and v3 are the method parameters. The method has 5 arguments (4 MSB bits of the second byte)[5].

70	invoke-direct { parameters }, methodtocall	Invokes a method with parameters without the virtual method resolution.	7010 0800 0100 - invoke-direct {v1}, java.lang.Object.<init>:()V // method@0008 Invokes the 8th method in the method table with just one parameter, v1 is the "this" instance[5].

0A	move-result vx	Move the result value of the previous method invocation into vx.	0A00 - move-result v0 Move the return value of a previous method invocation into v0.

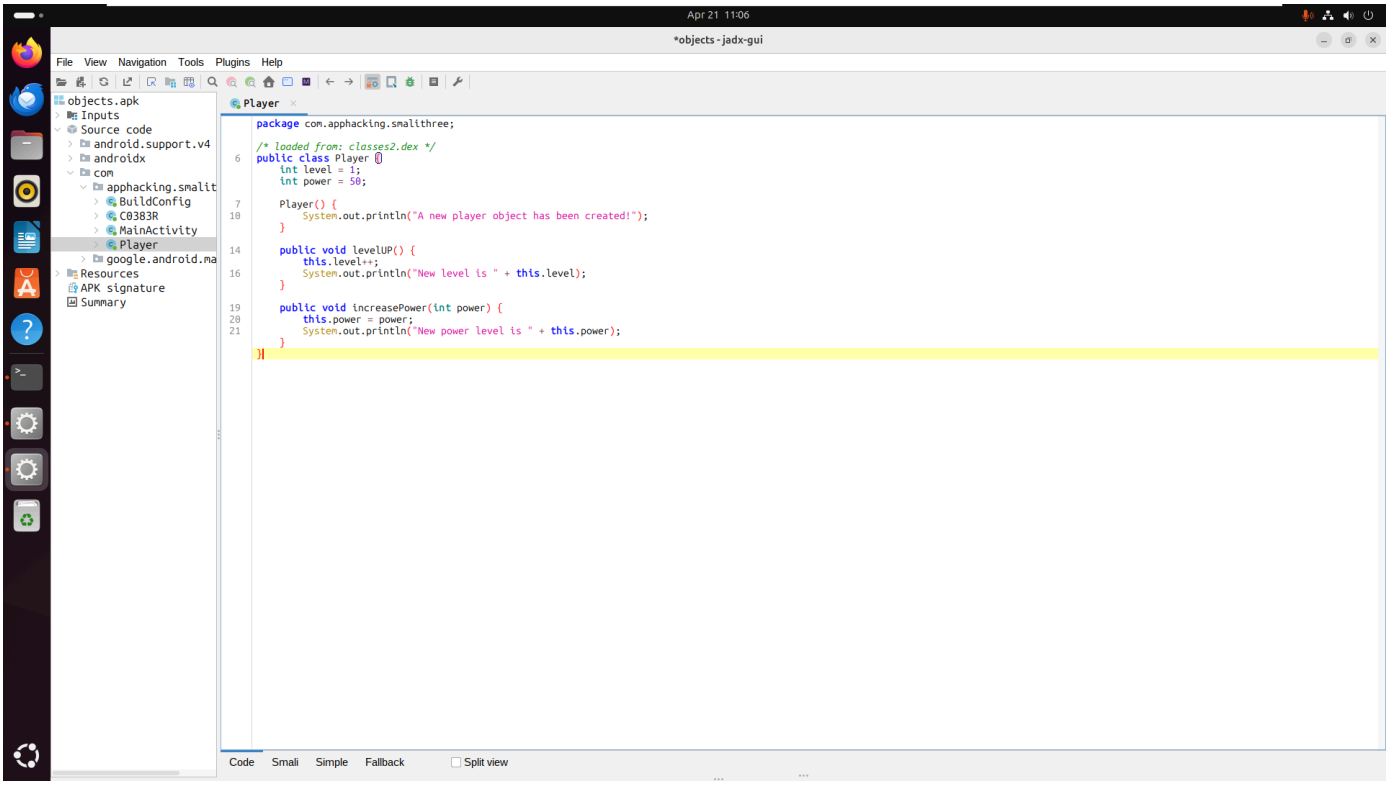
هنا بقي ده مثال تاني اه لو عندك invoke-virtual كده بستدعي حاجة داخل object فهو هنا {v0,v1} --> player.increasepower(900)

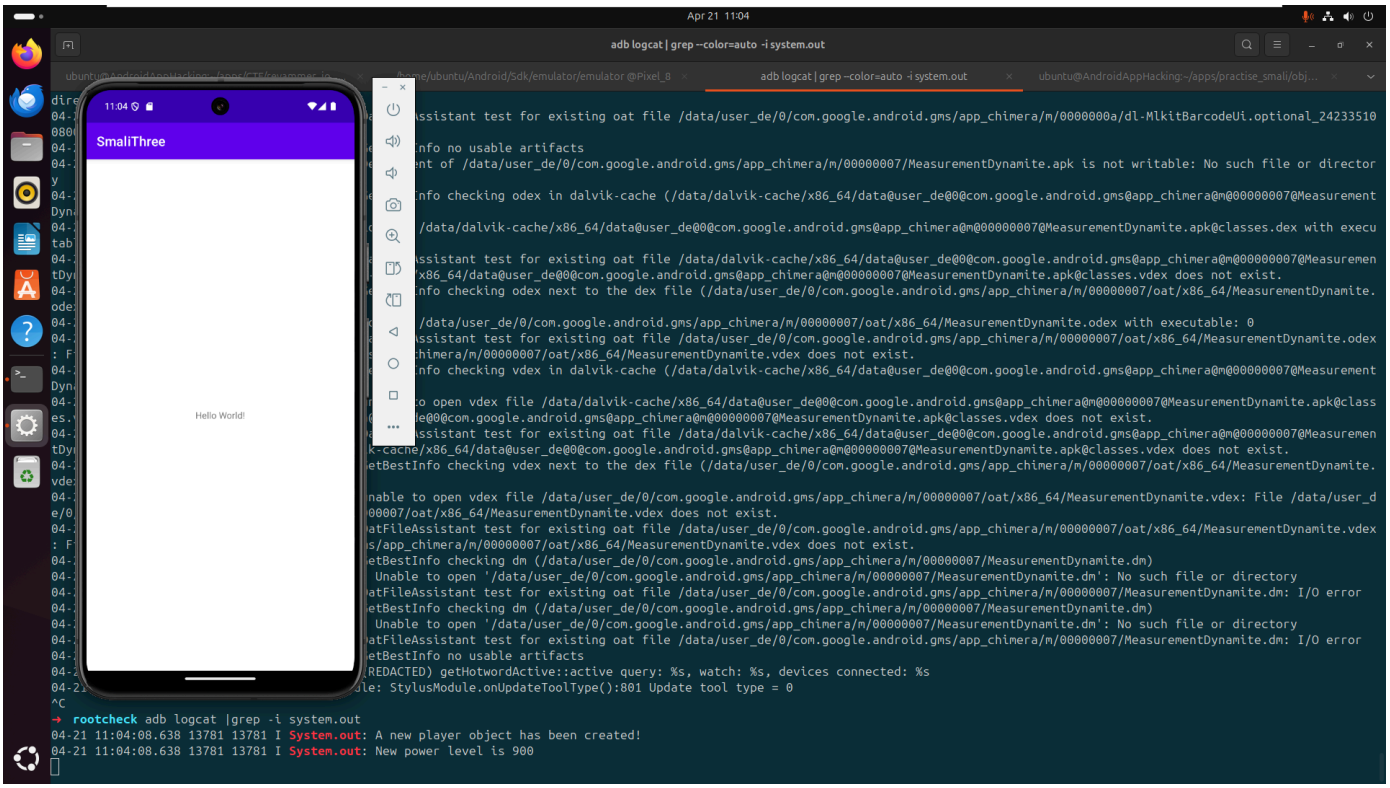


هنا بقي هنستخدمها في ايه هو ان نعمل create public methodخاصة بينا نعمل بها للي عاوزين نعمله مثلا ننشيئ حاجة مش موجودة نتخطي حاجة وده خلي باستخدام invoke-virtual to create public method

ex : in this app after install the app and show the logcat and after analyze the system we find he contain method for levelup

so how we declare this method on my smali code and execute it



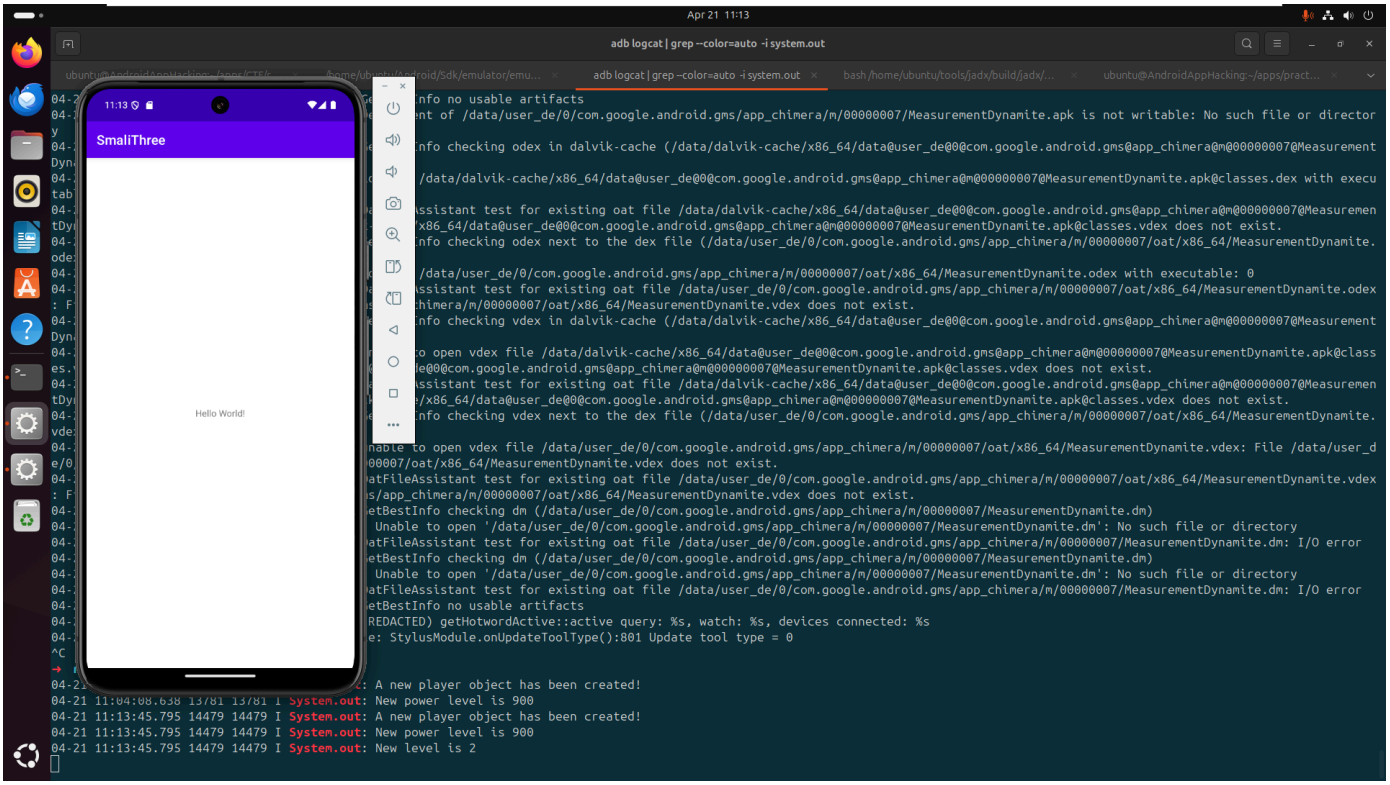


so on file MainActivity.smali we will add this line to execute this method

```
invoke-virtual {v0}, Lcom/apphacking/smalithree/Player;->levelUP()V
```

bulid the app and signing it and remove it and reinstall it

here we make the method of level and we increase the level to level two by using this public method



وده كان مثال زي نستخدم public method علشان نستدعيها او لو مش وجودة ننشيئها وممكن نعمل بيها حاجات كتير

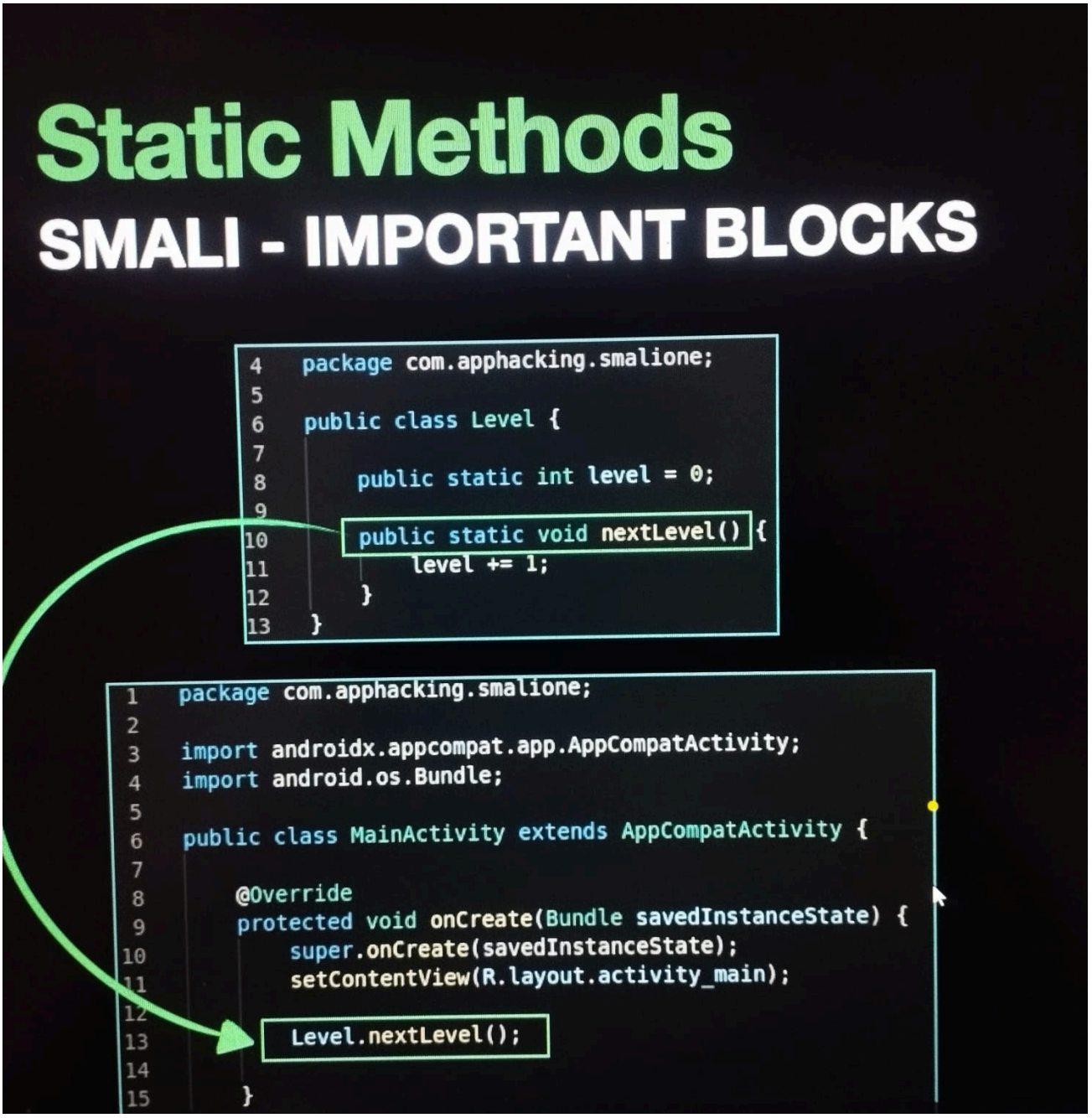
74-Static Method

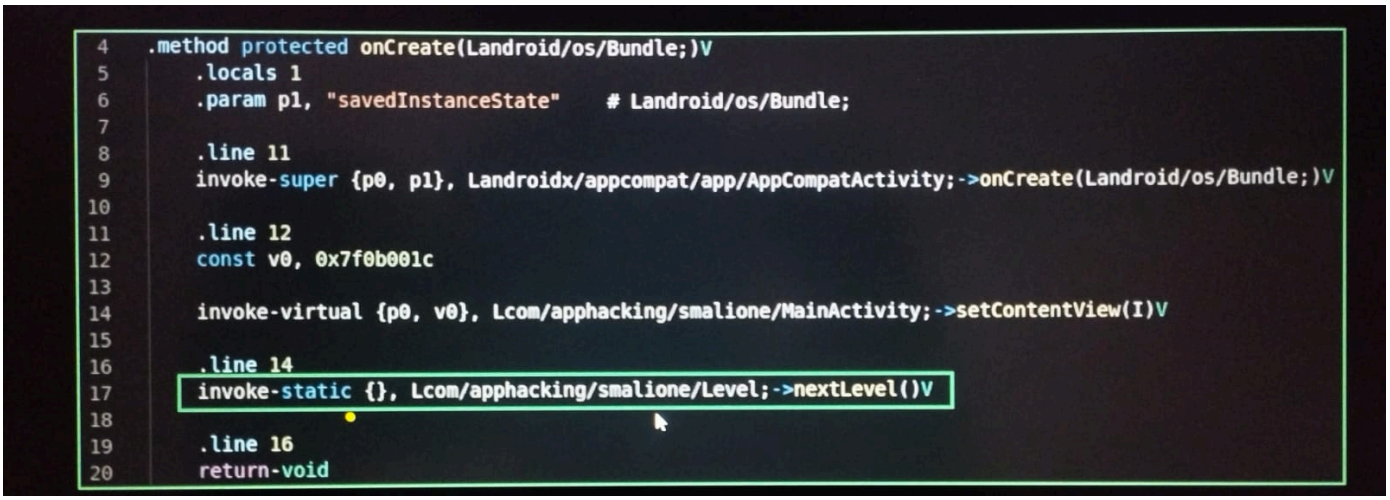
لو عاوزين بقي نستدعي static method برضه هي نفسها زي public method ولكن بدل invoke-virtual هتبقي invoke-static

تُستخدم لاستدعاء دالة ثابتة (**Static Method**) داخل الكود

الدوال الثابتة يمكن استدعاؤها مباشرة دونالحاجة إلى إنشاء كائن (Object) من الكلاس.

71	invoke-static {parameters}, methodtocall	Invokes a static method with parameters.	7110 3400 0400 - invoke-static {v4}, java.lang.Integer.parseInt:(Ljava/lang/String;)I // method@0034 Invokes method@34 static method. The method is called with one parameter, v4[5].





خلي بالك لو system بيطبع حاجة مثلا زي ("system.out,printl("this is static method ده هنا بيظهر في logcat

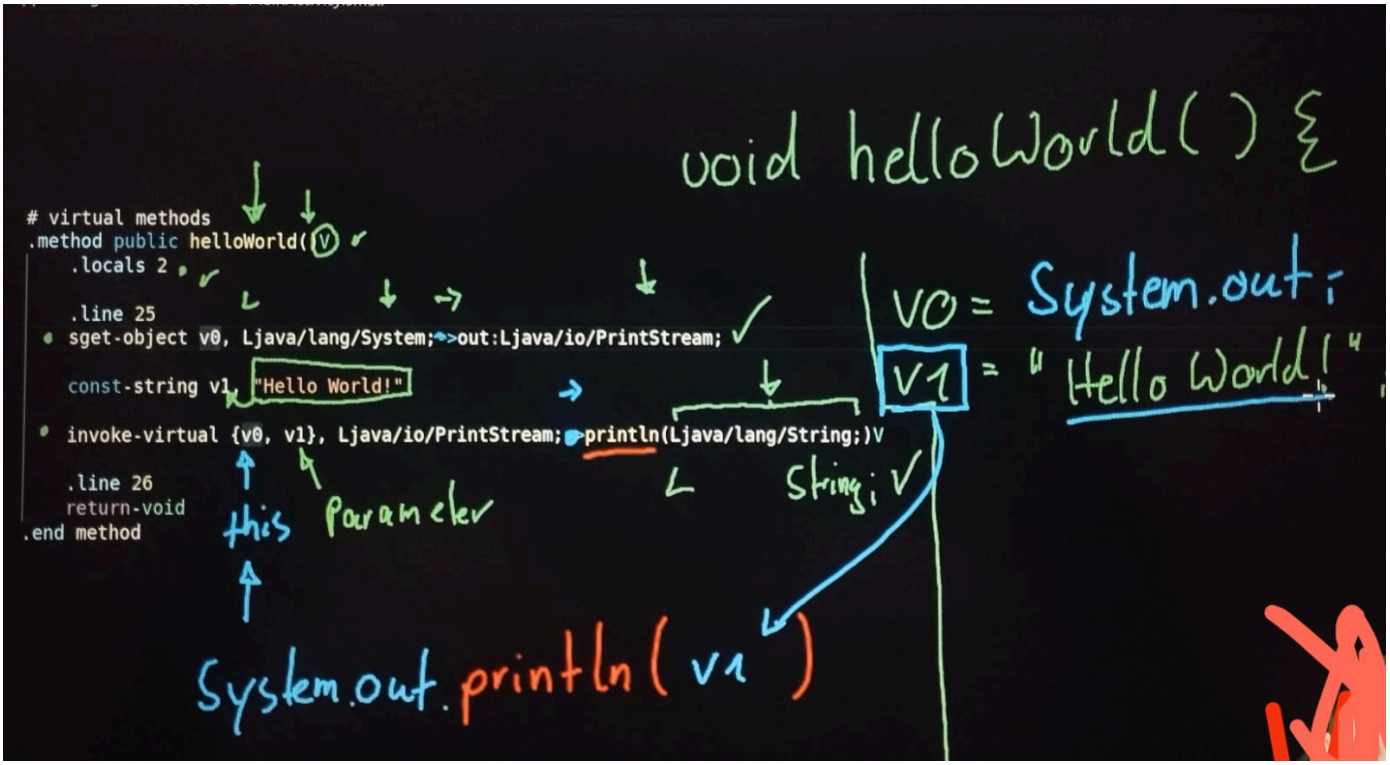
---

75- Hello world printed

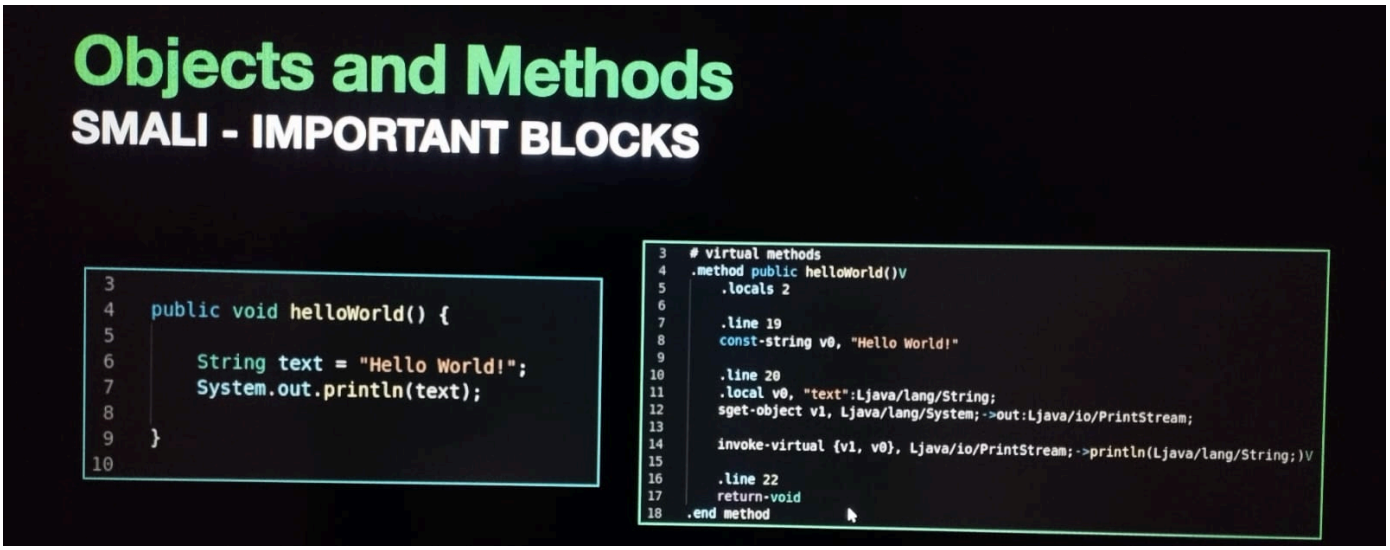
ازاي بقي لو معملناش class وعاوزين نطبع حاجة زي مثلا كود java ده

هنا اه مش موجود ان هو ينشئ object ولكن لو حللنا الكود ده هنلاقي ان sget-object بتجيب object اللي هو system وبتستدعي object اللي جواه اللي هو PrintStream وتخزنها في v0=System.PrintSystem --> v0

بعد كده بخزن string في v1="hello world" --> v1 ويستدعي method اللي هي Println علشان تطبع وبياخد parmater 2 اللي هما this and v1("hello world")

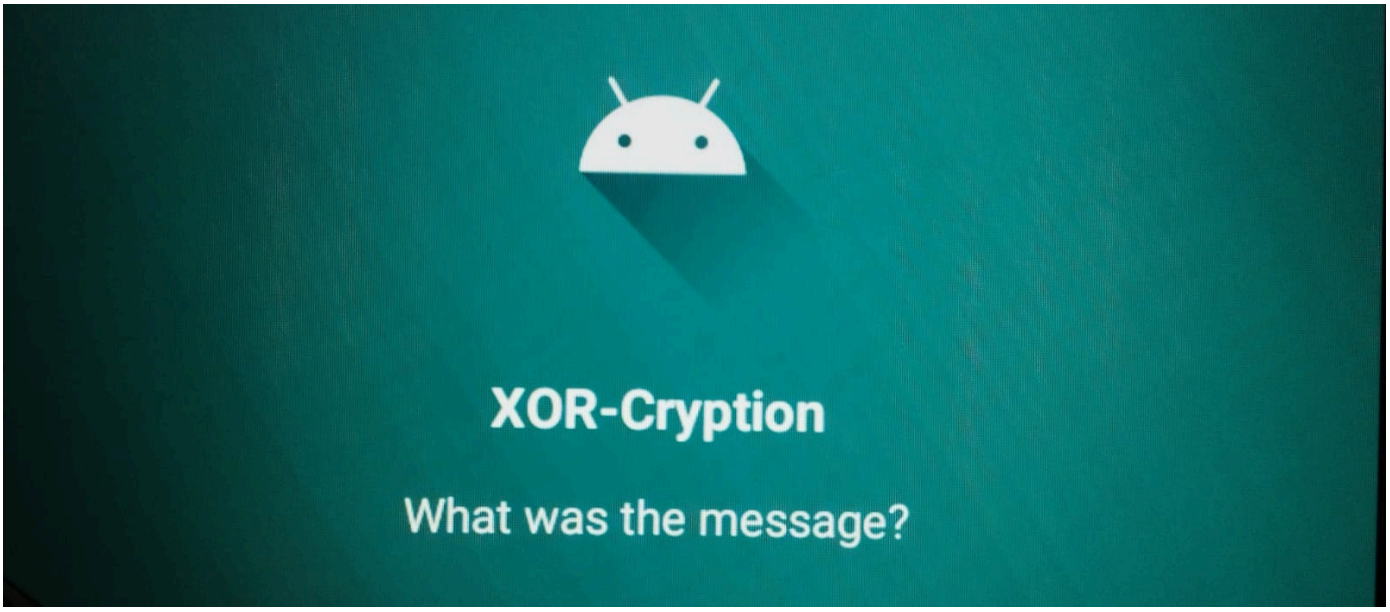


another example

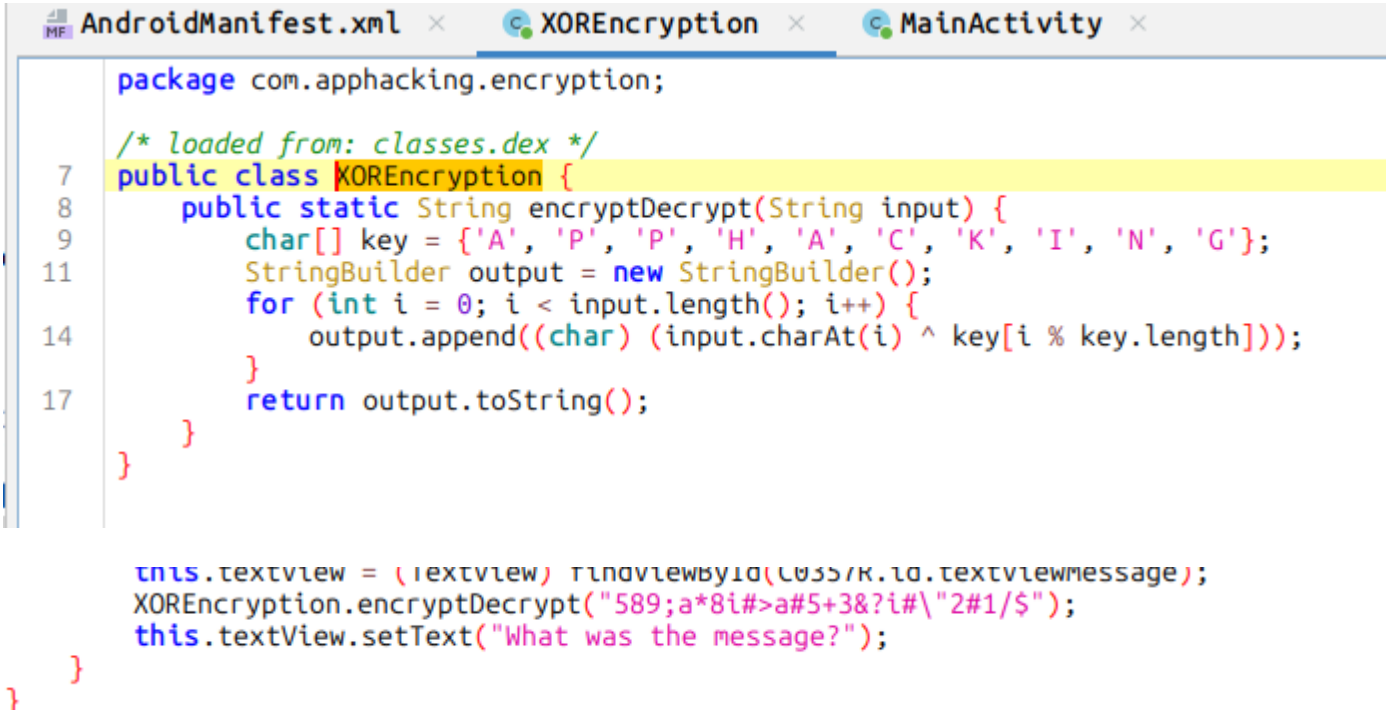


76-Printing out the Message Real World Example (Encryption) : Important

هنا اه في عبارة عن CTF وعاوزبن نعرف ايه هي message



لو شوفنا بقي الكود دلوقتي بتاع function اللي بتعمل encryption : هنلاقي ان هي بتاخد string وبتعمل عملية XOR مع string ودلوقتي هنشوف APPHACKING



ايه هي اصلا XOR : هي عبارة عن operation بتقارن بين 2 bit ولو مختلفين بطلع 1 ولو متشابهين بتطلع 0 زي كده مثلا

```
    01010111 01101001 01101011 01101001
⊕
    11110011 11110011 11110011 11110011

=   10100100 10011010 10011000 10011010
```

زي بقي بنعمله مع string هو ان احنا بنشوف رقم char في ASCII ونحوله ل binary وبعد كده نعمل عملية XOR زي كده مثلا

1. Convert Characters to ASCII (Decimal):

Each character in the string is converted to its ASCII (decimal) value. For example:

- `"h"` → `104`
- `"w"` → `119`

2. Convert ASCII to Binary:

Each ASCII value is represented as an 8-bit binary number:

- `104` → `01101000`
- `119` → `01110111`

3. Apply XOR (Bit by Bit):

Perform the XOR operation between corresponding bits of the two binary numbers:

- `01101000` (104, "h")

- `01110111` (119, "w")

```
    01101000
XOR 01110111
------------
    00011111 (31 in decimal)
```

4. (Optional) Convert Back to ASCII:

You can convert the XOR result back to ASCII for storage or transmission, though it might result in non-printable characters.

5. Reverse the Process (Decryption):

To decrypt, you XOR the encrypted value with the same key. For example:

- `31` → `00011111` (binary)
- Key: `119` → `01110111` (binary)

```
    00011111
XOR 01110111
------------
    01101000 (104 in decimal, "h")
```

code python for this operation

```
#!/usr/bin/python3
def encrypt_decrypt(input_string, key):
    # Initialize the output as a list of characters
    output = []

    # Perform XOR operation
    for i in range(len(input_string)):
        # XOR each character in input_string with the corresponding
character in the key
        output.append(chr(ord(input_string[i]) ^ ord(key[i % len(key)])))

    # Join the list to form the resulting string
    return ''.join(output)

# Example Usage
input_string = input("Enter the string to encrypt/decrypt: ")
key = "APPHACKING"  # Example key

# Encrypt/Decrypt
result = encrypt_decrypt(input_string, key)
```

```
# Display Results
print("Result (raw):", result)
print("Result (hex):", result.encode().hex())
```

he will convert the message to hex and we will convert from hex to ascii to know the secert message

بس دلوقتي تعالي نحلل smali code

```
        XOREncryption.encryptDecrypt("589;a*8i#>a#5+3&?i#\"2#1/$");
        this.textView.setText("What was the message?");
    }
```

```
    .line 21
    const-string v0, "589;a*8i#>a#5+3&?i#\"2#1/$"

    invoke-static {v0}, Lcom/apphacking/encryption/XOREncryption;->encryptDecrypt(Ljava/lang/String;)Ljava/lang/String;

    move-result-object v0

    .line 22
    .local v0, "encrypted":Ljava/lang/String;
    iget-object v1, p0, Lcom/apphacking/encryption/MainActivity;->textView:Landroid/widget/TextView;

    const-string v2, "What was the message?"

    invoke-virtual {v1, v2}, Landroid/widget/TextView;->setText(Ljava/lang/CharSequence;)V

    .line 24
    return-void
.end method
```

هنا اه من line 21. -- > هو هنا بيخزن string في v0 وبعج كده بيعمل invoke-static ان هو بيستدعي XOREncription.encryptDecrypt(v0)

1- First Soulation

وبع كده واهم حاجو بيخزن القمية اللي بترجع في move-result-object v0 --> v0 يبقي كده عرفنا ان ناتج هيبقي في v0 من خلال كود smali

```
    .line 21
    const-string v0, "589;a*8i#>a#5+3&?i#\"2#1/$"

    invoke-static {v0}, Lcom/apphacking/encryption/XOREncryption;-
>encryptDecrypt(Ljava/lang/String;)Ljava/lang/String;

    move-result-object v0
```

كده بقي عرفنا ان القيمة هترجع في v0 عاوزين بقي نتسخدم smali علشان نحصل علي القمية منه

to print the value of V0 we need to use System.out.println(v0) so we need to create object and to create those objects from Delvik Operation Codes :

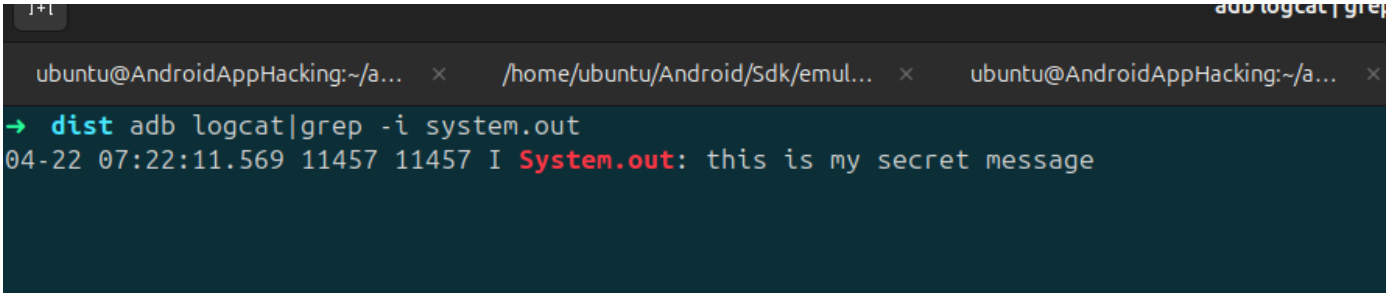
1. **sget-object System.out**
2. **println() --> public method so we need to invoke-virutal**
3. **System.out.println**

62	sget-object vx,field_id	Reads the object reference field identified by the field_id into vx.	6201 0C00 - sget-object v1, Test3.os1:Ljava/lang/Object; // field@000c
			Reads field@000c (entry #CH in the field id table) into v1.

```
sget-object v3,Ljava/lang/System;->out: Ljava/io/PrintStream;
invoke-virtual {v3,v0},Ljava/io/PrintStream;->println(Ljava/lang/String;)V
```

دلوقتي بقي لو علمنا build وشوفنا logcat

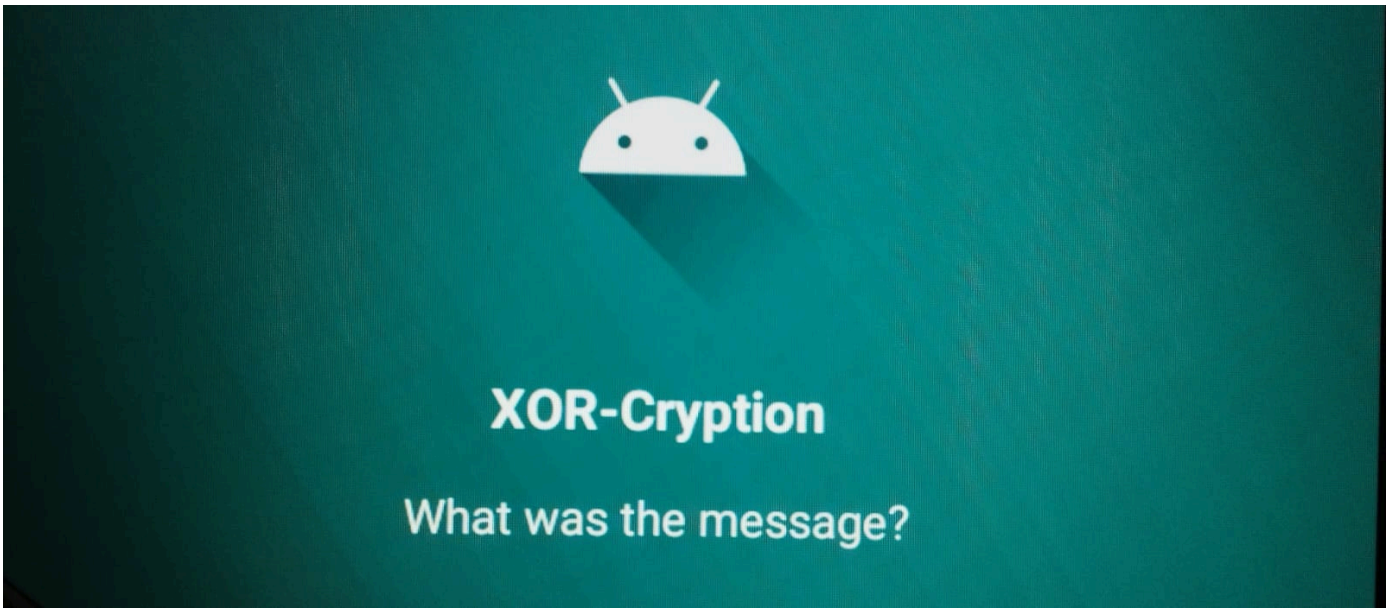
```
adb logcat | grep -i system.out
```



the decrypt message : this is my secert message

2-Second Soulation

هنا بقي بدل ما نخليه يظهر في logcat نخليه يطبع في الشاشة بدل v2 : يعني هنا هو بيجيب علي الشاشة قيمة v2 اللي هي "What was the message" احنا بدل دي بقي نحط القيمة اللي هتيجي بعد ما يروح لل function اللي هي encryptDecrupt بدل ما يعمل move-result- نخليها object v0 و بكد هي هيطبع الناتج move-result-object v2

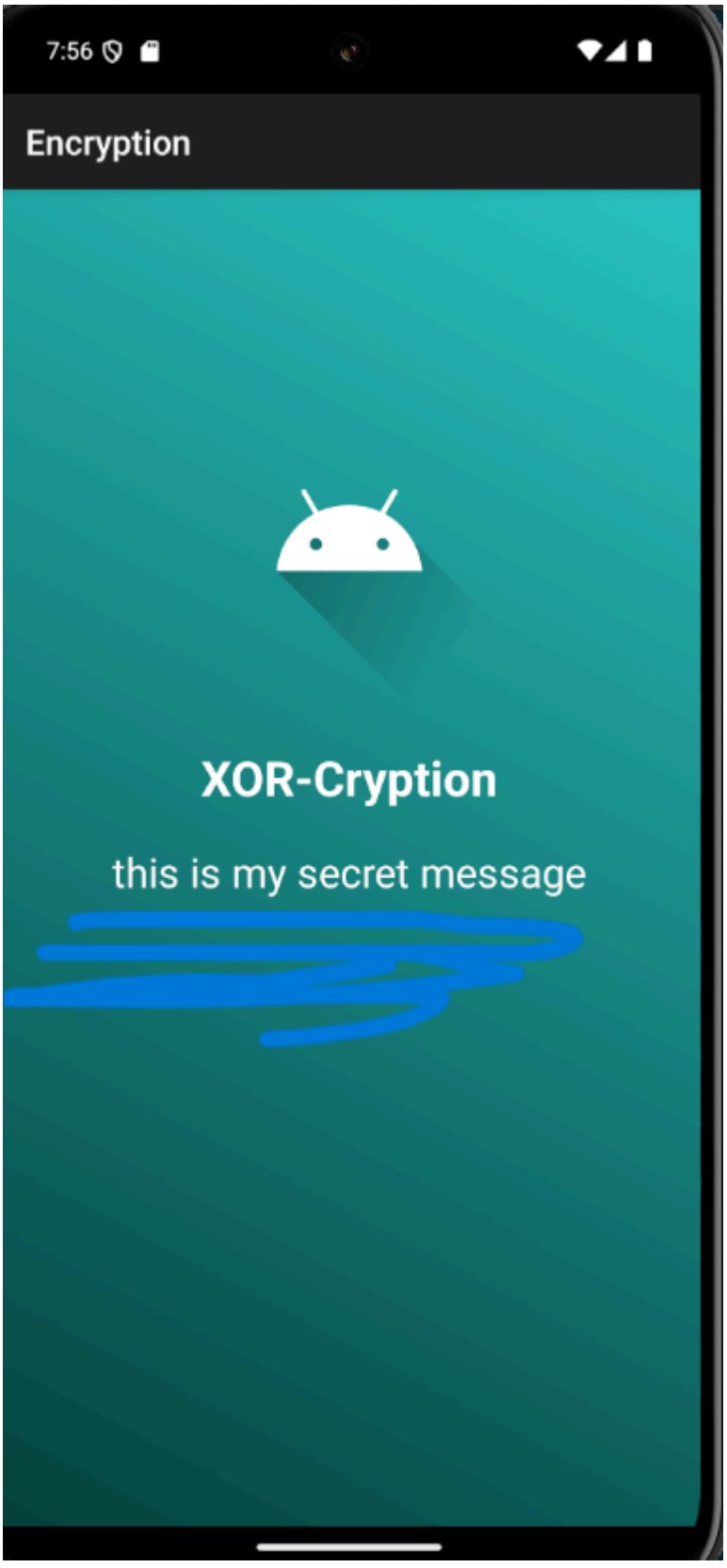



```
    .line 21
    const-string v0, "589;a*8i#>a#5+3&?i#\"2#1/$"

    invoke-static {v0}, Lcom/apphacking/encryption/XOREncryption;->encryptDecrypt(Ljava/lang/String;)Ljava/lang/String;

    move-result-object v0

    .line 22
    .local v0, "encrypted":Ljava/lang/String;
    iget-object v1, p0, Lcom/apphacking/encryption/MainActivity;->textView:Landroid/widget/TextView;

    const-string v2, "What was the message?"

    invoke-virtual {v1, v2}, Landroid/widget/TextView;->setText(Ljava/lang/CharSequence;)V

    .line 24
    return-void
.end method
```

هنا بقي الكود اللي هنعمله علشان نغير ل v2

```
    .line 21
    const-string v0, "589;a*8i#>a#5+3&?i#\"2#1/$"

    invoke-static {v0}, Lcom/apphacking/encryption/XOREncryption;->encryptDecrypt(Ljava/lang/String;)Ljava/lang/String;

    move-result-object v2


    .line 22
    .local v0, "encrypted":Ljava/lang/String;
    iget-object v1, p0, Lcom/apphacking/encryption/MainActivity;->textView:Landroid/widget/TextView;

    invoke-virtual {v1, v2}, Landroid/widget/TextView;->setText(Ljava/lang/CharSequence;)V

    .line 24
    return-void
.end method
```

وبكده هيظهر علي الشاشة



78- One Challenge To Recep All : part 1

try and catch on smali

1. **`try` Block**:
    - You specify the range of instructions that belong to the `try` block using `.try` and `.catch` directives.

2. **`catch` Block**:
    - You define the exception type to catch and the label where the exception should be handled.

Example in Smali

```
.method public exampleTryCatch()V
    .locals 1

    :try_start
    # Code in try block
    const v0, 0x0              # Assign 0 to v0
    div-int v0, v0, v0         # Divide by zero (causes ArithmeticException)
    goto :try_end              # If no exception, skip the catch block

    .catch Ljava/lang/ArithmeticException; {:try_start .. :try_end}
:catch_block

    :try_end
    return-void

    :catch_block
    # Code to handle exception
    const-string v0, "Exception caught!"
    invoke-static {v0}, Landroid/util/Log;->d(Ljava/lang/String;)I
    return-void
    .end method
```

Explanation

1. **`try` Block (`:try_start .. :try_end`)**:
    - The `try` block starts at `:try_start` and ends at `:try_end`.
    - Any exception that occurs within this range will trigger the `catch` block.
2. **`catch` Directive**:
    - The `.catch` directive specifies:
        - The type of exception to catch: `Ljava/lang/ArithmeticException;`.
        - The range of instructions to monitor for exceptions: `:try_start .. :try_end`.
        - The label to jump to if the exception is caught: `:catch_block`.
3. **Exception Handling (`:catch_block`)**:
    - This block is executed if the specified exception is thrown.

- In this example, it logs the message "Exception caught!".

4. **Flow Control**:
   - If no exception occurs, the execution continues normally after `:try_end`.

دلوقتي بقي فيالتطبيق ده هو بيستخدم AES encryption هنا في الكود ده لو لاقينا زي كده تشفير بيبقي اهم حاجة لازم نهتم ان نعرفهم هو secert key and Initialization vector

وعلشان نجيب secert key لازم نستدعي method --> اللي هي getEncoded()

getEncoded

public byte[] getEncoded ()

Returns the key material of this secret key. --> return the value of secert key

AESCrypto

```java
package com.apphacking.aes;

import java.security.InvalidKeyException;
import java.security.NoSuchAlgorithmException;
import javax.crypto.BadPaddingException;
import javax.crypto.Cipher;
import javax.crypto.IllegalBlockSizeException;
import javax.crypto.KeyGenerator;
import javax.crypto.NoSuchPaddingException;
import javax.crypto.SecretKey;

/* loaded from: classes2.dex */
public class AESCrypto {
    public static byte[] encryptMessage(byte[] plaintext) {
        try {
            KeyGenerator keygen = KeyGenerator.getInstance("AES");
            keygen.init(256);
            SecretKey key = keygen.generateKey();
            Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5PADDING");
            cipher.init(1, key);
            byte[] ciphertext = cipher.doFinal(plaintext);
            cipher.getIV();
            return ciphertext;
        } catch (InvalidKeyException e) {
            e.printStackTrace();
            return null;
        } catch (NoSuchAlgorithmException e2) {
            e2.printStackTrace();
            return null;
        } catch (BadPaddingException e3) {
            e3.printStackTrace();
            return null;
        } catch (IllegalBlockSizeException e4) {
            e4.printStackTrace();
            return null;
        } catch (NoSuchPaddingException e5) {
            e5.printStackTrace();
            return null;
        }
    }
}
```

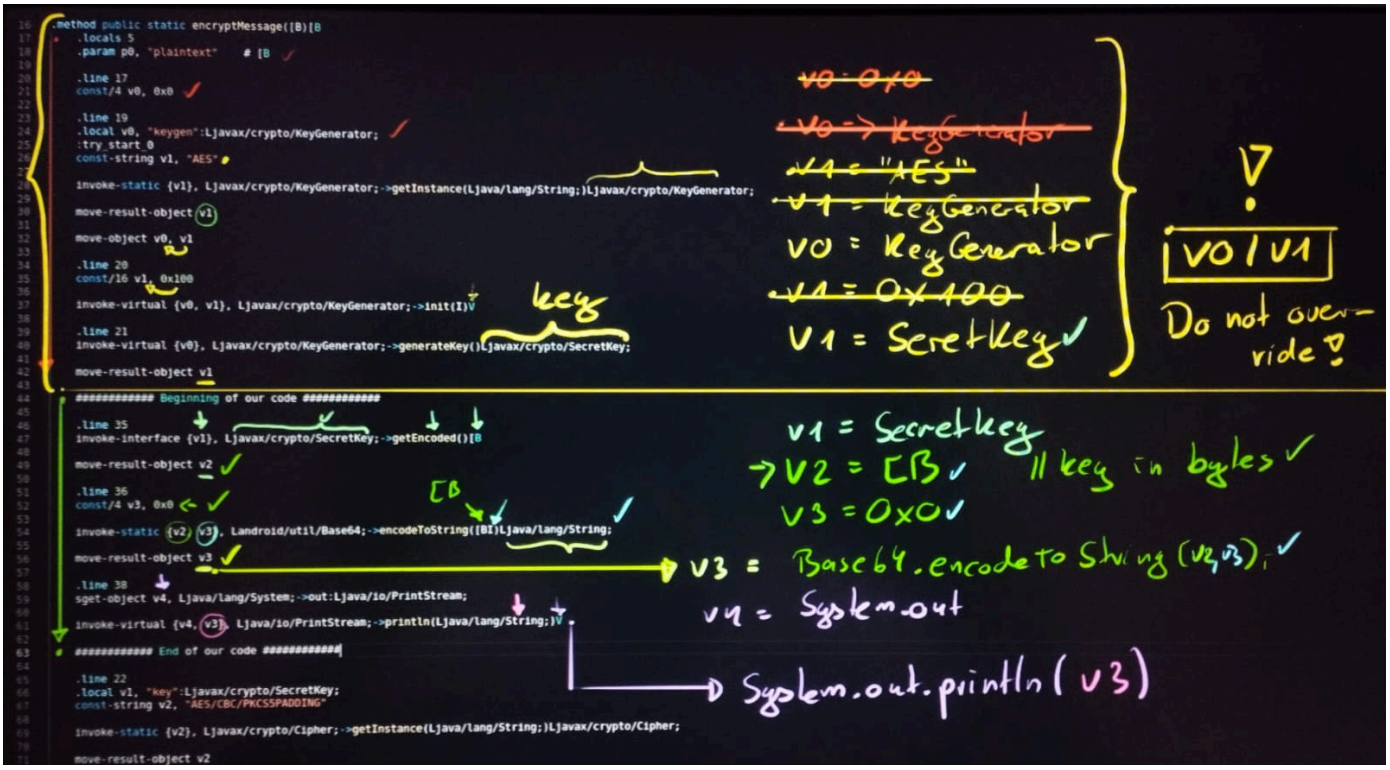
فلازم نضيفها بقي في الكود اما java او smali وتعمل print لها وبكده يبقي حصلنا علي secert key : هنا بقي اللي بحتاجين نضيفه علشان نجيب secert key

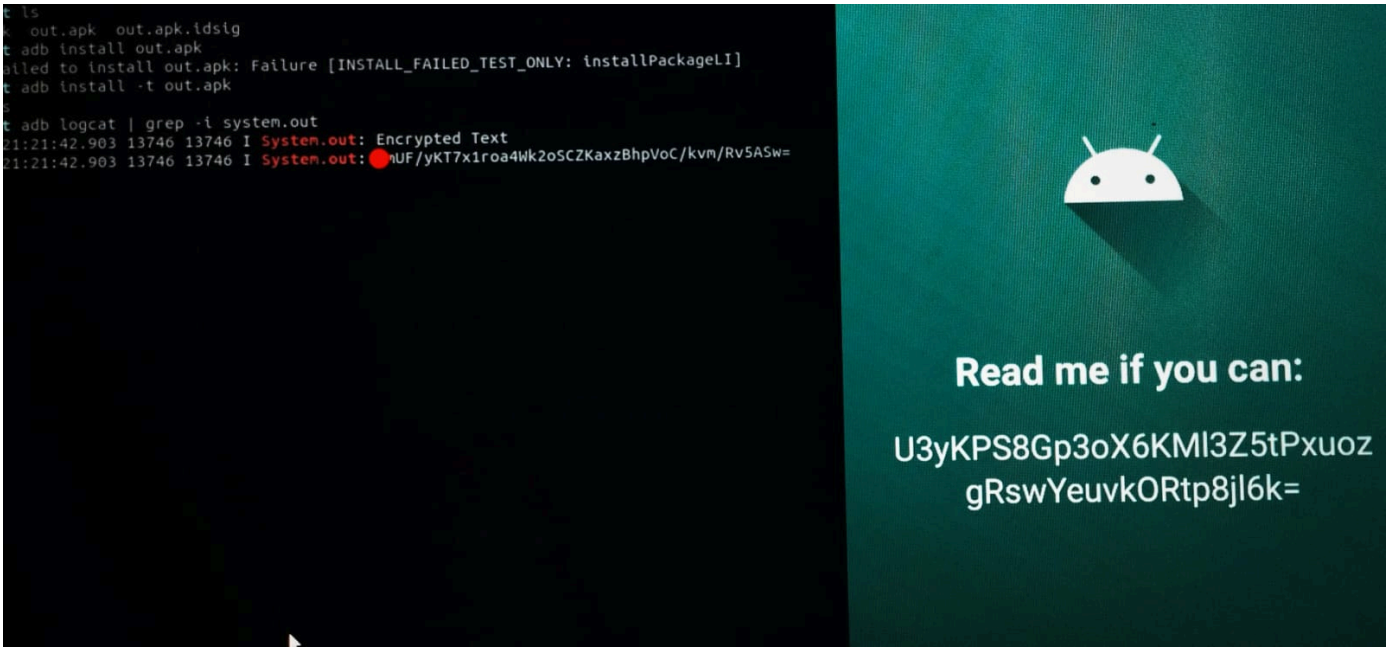
```
myArray=key.getEncoded();
secertkey=Base64.encodeToString(myArray,0);
System.out.println(secertkey);
```

smali

```
invoke-interface {v1},Ljava/crypto/SecertKey;->getEncoded();[B
move-result-object v2
.line 36
const/4 v3,0x0
invoke-static {v2,v3},Landroid/util/Base64;-
```

```
>encodeToString([BI)Ljava/lang/String;
move-result-object v3;
.line 38
sget-object v4,Ljava/lang/System;->out:Ljava/io/PrintStream;
invoke-virtual {v4,v3},Ljava/io/PrintStream;->println(Ljava/lang/String;)V
.line 42
return-void
```





وبكده عرفنا نجيب Secert Key اللي بيتنشأ من AES encryption

part 2

هنا بقي هنشرح ازاي بدل ما نخلي مثلا الحاجة اللي عايزنها تظهر تكون علي screen بتاعت app نفسه يعني متظهرش في logcat وعلشان تظهر لازم تربطها ب actvity لان activity هو page احنا هنا هنربطها بMainActivity لان هوactivity اللي موجود فقط

لو عاوزين ان اول ما نفتح التطبيق يظهر text

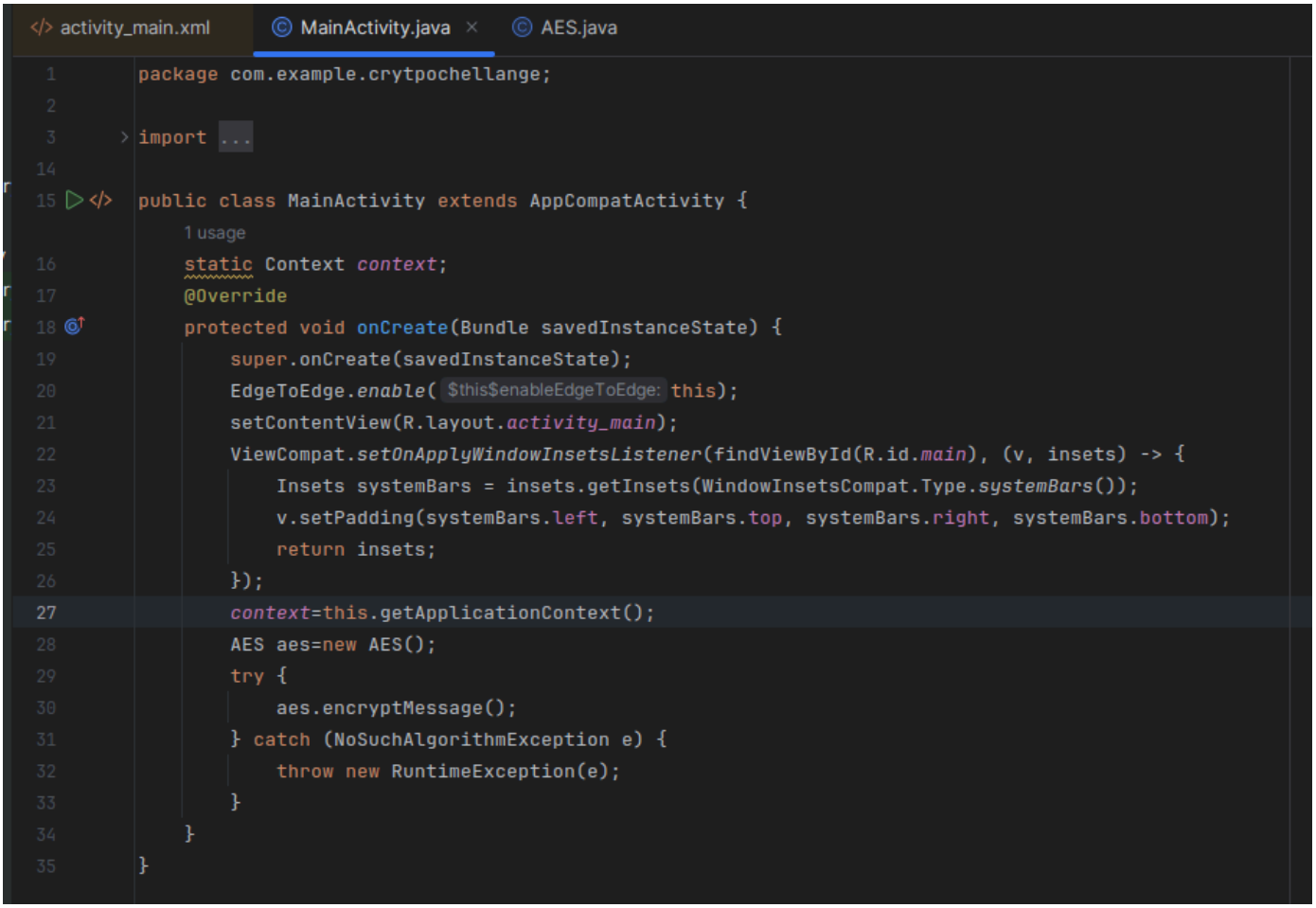
```
Toast.makeText(this.getApplicationContext(), "Hacking Your Applicatoin",
Toast.LENGTH_SHORT).show();
```

علشان نخلي text يظهر علي الشاشة بنسخدم Context

```
in MainActivity
static Context context;
context=this.getApplicationContext();
in AES
Toast.makeText(MainActivity.context,"this pap",Toast.LENGTH_LONG).show();
```

هنا بقي مثال علي اللي عملناه

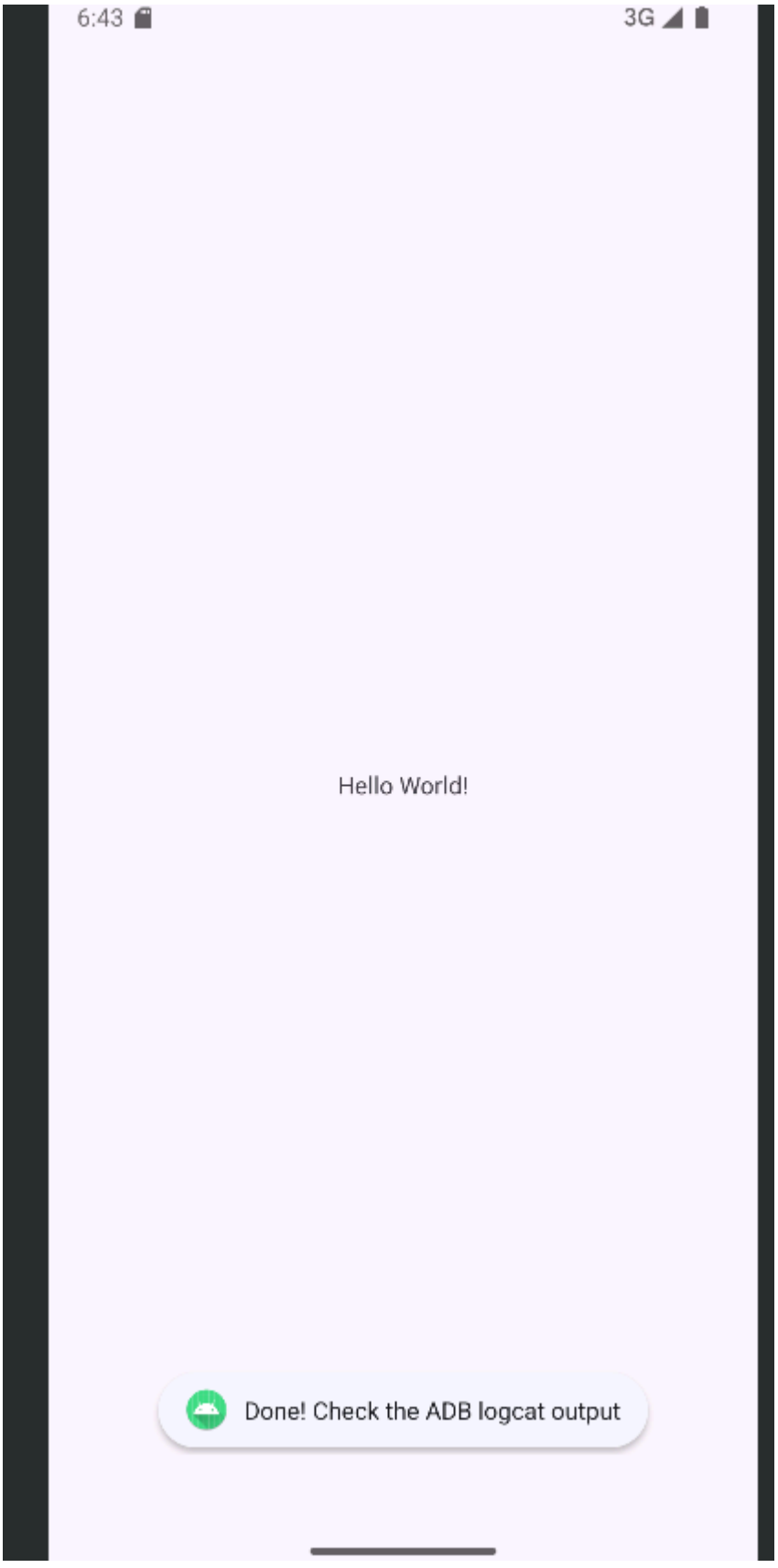
1-MainActivity code

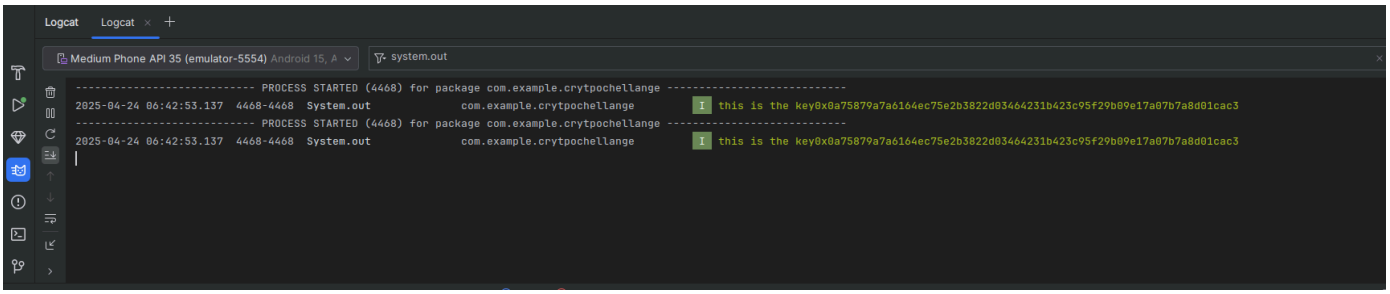


2-AES code

activity_main.xml | MainActivity.java | AES.java

```java
package com.example.crytpochellange;

import android.widget.Toast;

import java.security.NoSuchAlgorithmException;

import javax.crypto.KeyGenerator;
import javax.crypto.SecretKey;

2 usages
public class AES {
    1 usage
    public void encryptMessage() throws NoSuchAlgorithmException {
        KeyGenerator keygen= KeyGenerator.getInstance( algorithm: "AES");
        keygen.init( keysize: 256);
        SecretKey key=keygen.generateKey();
        Toast.makeText(MainActivity.context, text: "Hacking into your Application",Toast.LENGTH_LONG).show();
        byte[] byteArray;
        byteArray=key.getEncoded();
        String keyString="0x";
        for(int i=0;i<byteArray.length;i++){
            keyString+=String.format("%02x",byteArray[i]);
        }
        System.out.println(keyString);
        Toast.makeText(MainActivity.context, text: "Done! Check the ADB logcat output",Toast.LENGTH_LONG).show();
    }
}
```


6:43
3G
Hello World!
Done! Check the ADB logcat output


Logcat
Logcat
Medium Phone API 35 (emulator-5554) Android 15, A
system.out
---------- PROCESS STARTED (4468) for package com.example.crytpochellange ----------
2025-04-24 06:42:53.137 4468-4468 System.out com.example.crytpochellange I this is the key0x0a75879a7a6164ec75e2b3822d03464231b423c95f29b09e17a07b7a8d01cac3
---------- PROCESS STARTED (4468) for package com.example.crytpochellange ----------
2025-04-24 06:42:53.137 4468-4468 System.out com.example.crytpochellange I this is the key0x0a75879a7a6164ec75e2b3822d03464231b423c95f29b09e17a07b7a8d01cac3

دلوقتي بقي الكود اللي عملناه ده هناخد بقي كود smali ونحطه في التطبيق بتاعنا اللي هنعدل عليه: هنا اه java code for app دلوقتي بقي اللي محتاجين نعجل عليها من بعد سطر 21 هناخد كود smali من التطبيق اللي عملناه ونحطه في كود smali للتطبيق ده

AESCrypto

```java
package com.apphacking.aes;

import java.security.InvalidKeyException;
import java.security.NoSuchAlgorithmException;
import javax.crypto.BadPaddingException;
import javax.crypto.Cipher;
import javax.crypto.IllegalBlockSizeException;
import javax.crypto.KeyGenerator;
import javax.crypto.NoSuchPaddingException;
import javax.crypto.SecretKey;

/* loaded from: classes2.dex */
public class AESCrypto {
    public static byte[] encryptMessage(byte[] plaintext) {
        try {
            KeyGenerator keygen = KeyGenerator.getInstance("AES");
            keygen.init(256);
            SecretKey key = keygen.generateKey();
            Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5PADDING");
            cipher.init(1, key);
            byte[] ciphertext = cipher.doFinal(plaintext);
            cipher.getIV();
            return ciphertext;
        } catch (InvalidKeyException e) {
            e.printStackTrace();
            return null;
        } catch (NoSuchAlgorithmException e2) {
            e2.printStackTrace();
            return null;
        } catch (BadPaddingException e3) {
            e3.printStackTrace();
            return null;
        } catch (IllegalBlockSizeException e4) {
            e4.printStackTrace();
            return null;
        } catch (NoSuchPaddingException e5) {
            e5.printStackTrace();
            return null;
        }
    }
}
```

هنا اه انا عامل comment للكود اللي المفوض هروح نحطه في app

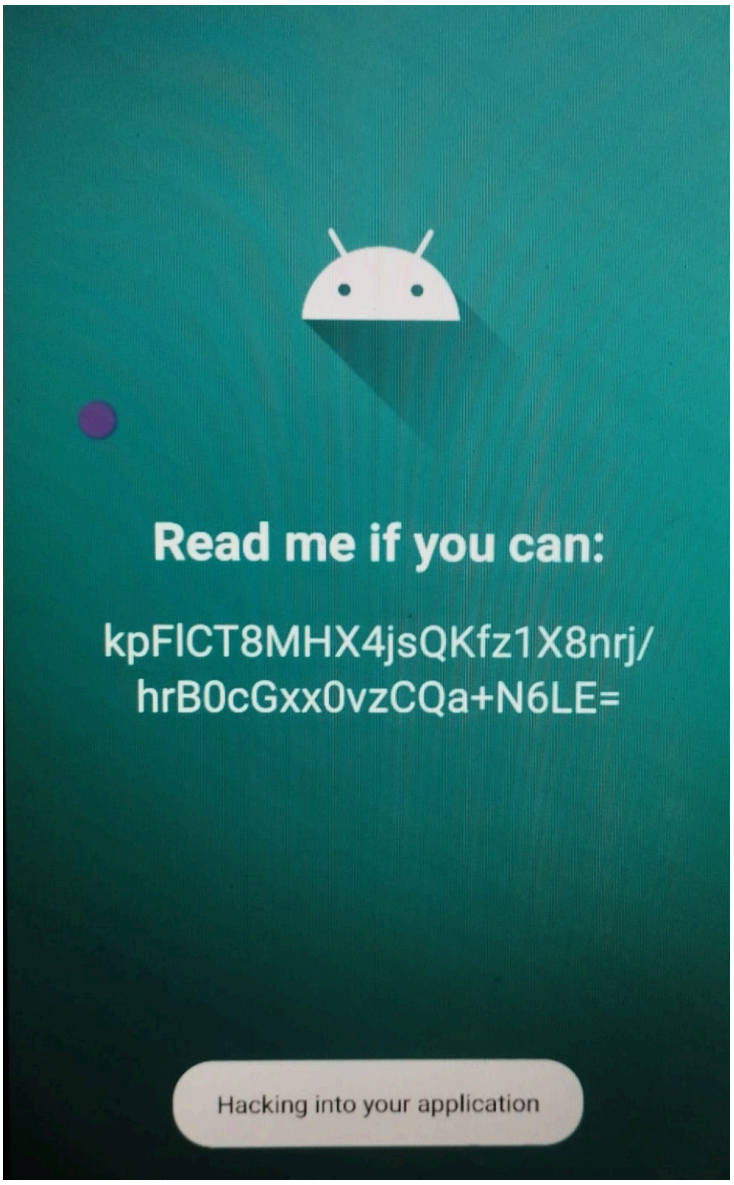
```
###### Class com.example.crytpochellange.AES
(com.example.crytpochellange.AES)
.class public Lcom/example/crytpochellange/AES;
.super Ljava/lang/Object;
.source "AES.java"


# direct methods
.method public constructor <init>()V
    .registers 1
```

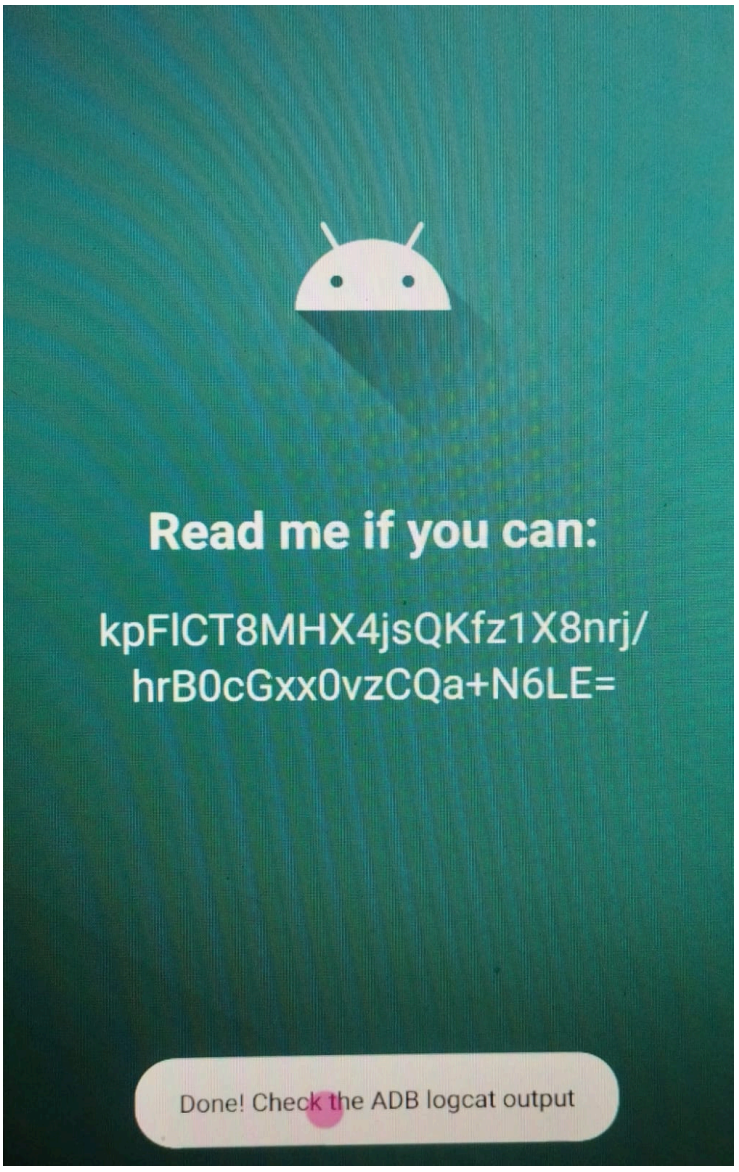
```
    .line 10
    invoke-direct {p0}, Ljava/lang/Object;-><init>()V

    return-void
.end method


# virtual methods
.method public encryptMessage()V
    .registers 10
    .annotation system Ldalvik/annotation/Throws;
        value = {
            Ljava/security/NoSuchAlgorithmException;
        }
    .end annotation

    .line 12
    const-string v0, "AES"

    invoke-static {v0}, Ljavax/crypto/KeyGenerator;-
>getInstance(Ljava/lang/String;)Ljavax/crypto/KeyGenerator;

    move-result-object v0

    .line 13
    .local v0, "keygen":Ljavax/crypto/KeyGenerator;
    const/16 v1, 0x100

    invoke-virtual {v0, v1}, Ljavax/crypto/KeyGenerator;->init(I)V

    .line 14
    invoke-virtual {v0}, Ljavax/crypto/KeyGenerator;-
>generateKey()Ljavax/crypto/SecretKey;

    move-result-object v1
    ########   form here begin of out  code   ########
    .line 15
    .local v1, "key":Ljavax/crypto/SecretKey;
    sget-object v2, Lcom/example/crytpochellange/MainActivity;-
>context:Landroid/content/Context;

    const-string v3, "Hacking into your Application"
```

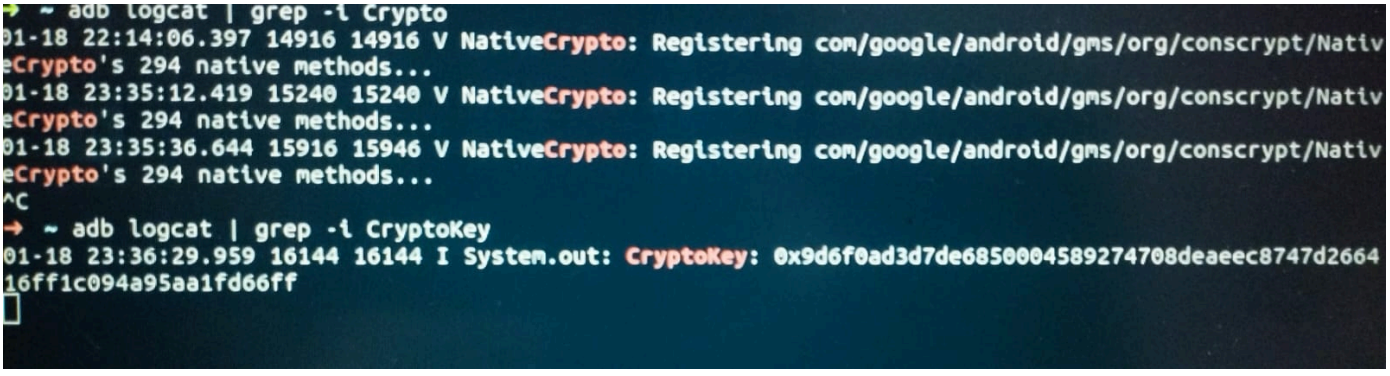
```
    const/4 v4, 0x1

    invoke-static {v2, v3, v4}, Landroid/widget/Toast;-
>makeText(Landroid/content/Context;Ljava/lang/CharSequence;I)Landroid/widget
/Toast;

    move-result-object v2

    invoke-virtual {v2}, Landroid/widget/Toast;->show()V

    .line 17
    invoke-interface {v1}, Ljavax/crypto/SecretKey;->getEncoded()[B

    move-result-object v2

    .line 18
    .local v2, "byteArray":[B
    const-string v3, "0x"

    .line 19
    .local v3, "keyString":Ljava/lang/String;
    const/4 v5, 0x0

    .local v5, "i":I
    :goto_22
    array-length v6, v2

    if-ge v5, v6, :cond_49

    .line 20
    new-instance v6, Ljava/lang/StringBuilder;

    invoke-direct {v6}, Ljava/lang/StringBuilder;-><init>()V

    invoke-virtual {v6, v3}, Ljava/lang/StringBuilder;-
>append(Ljava/lang/String;)Ljava/lang/StringBuilder;

    move-result-object v6

    aget-byte v7, v2, v5

    invoke-static {v7}, Ljava/lang/Byte;->valueOf(B)Ljava/lang/Byte;
```

```
    move-result-object v7

    filled-new-array {v7}, [Ljava/lang/Object;

    move-result-object v7

    const-string v8, "%02x"

    invoke-static {v8, v7}, Ljava/lang/String;->format(Ljava/lang/String;
[Ljava/lang/Object;)Ljava/lang/String;

    move-result-object v7

    invoke-virtual {v6, v7}, Ljava/lang/StringBuilder;-
>append(Ljava/lang/String;)Ljava/lang/StringBuilder;

    move-result-object v6

    invoke-virtual {v6}, Ljava/lang/StringBuilder;-
>toString()Ljava/lang/String;

    move-result-object v3

    .line 19
    add-int/lit8 v5, v5, 0x1

    goto :goto_22

    .line 22
    .end local v5    # "i":I
    :cond_49
    sget-object v5, Ljava/lang/System;->out:Ljava/io/PrintStream;

    new-instance v6, Ljava/lang/StringBuilder;

    invoke-direct {v6}, Ljava/lang/StringBuilder;-><init>()V

    const-string v7, "this is the key"

    invoke-virtual {v6, v7}, Ljava/lang/StringBuilder;-
>append(Ljava/lang/String;)Ljava/lang/StringBuilder;
```

```
    move-result-object v6

    invoke-virtual {v6, v3}, Ljava/lang/StringBuilder;-
>append(Ljava/lang/String;)Ljava/lang/StringBuilder;

    move-result-object v6

    invoke-virtual {v6}, Ljava/lang/StringBuilder;-
>toString()Ljava/lang/String;

    move-result-object v6

    invoke-virtual {v5, v6}, Ljava/io/PrintStream;-
>println(Ljava/lang/String;)V

    .line 23
    sget-object v5, Lcom/example/crytpochellange/MainActivity;-
>context:Landroid/content/Context;

    const-string v6, "Done! Check the ADB logcat output"

    invoke-static {v5, v6, v4}, Landroid/widget/Toast;-
>makeText(Landroid/content/Context;Ljava/lang/CharSequence;I)Landroid/widget
/Toast;

    move-result-object v4

    invoke-virtual {v4}, Landroid/widget/Toast;->show()V
    ###### to here  end of our code  #######
    .line 24
    return-void
.end method
```


Read me if you can:
kpFlCT8MHX4jsQKfz1X8nrj/
hrB0cGxx0vzCQa+N6LE=
Hacking into your application





---

part 3

هنا هنشوف ازاي هنجيب IV :اللي المفرو ض نعمله ان نعرف ByteArray ونحط فيه القيمة بتاعت Cipher.iv وبعد ككده محتاجين نطبعه ب System.out.println

```
            System.out.println("IV: " +
Base64.encodeToString(cipher.getIV(), 0));
            System.out.println("Return Value: " +
Base64.encodeToString(ciphertext, 0));
```

Java code

AESCrypto

```java
package com.apphacking.aes;

import android.util.Base64;
import java.security.InvalidKeyException;
import java.security.NoSuchAlgorithmException;
import javax.crypto.BadPaddingException;
import javax.crypto.Cipher;
import javax.crypto.IllegalBlockSizeException;
import javax.crypto.KeyGenerator;
import javax.crypto.NoSuchPaddingException;
import javax.crypto.SecretKey;

/* loaded from: classes2.dex */
public class AESCrypto {
    public static byte[] encryptMessage(byte[] plaintext) {
        System.out.println("Param: " + Base64.encodeToString(plaintext, 0));
        try {
            KeyGenerator keygen = KeyGenerator.getInstance("AES");
            keygen.init(256);
            SecretKey key = keygen.generateKey();
            byte[] myArray = key.getEncoded();
            System.out.println("Key: " + Base64.encodeToString(myArray, 0));
            Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5PADDING");
            cipher.init(1, key);
            byte[] ciphertext = cipher.doFinal(plaintext);
            System.out.println("IV: " + Base64.encodeToString(cipher.getIV(), 0));
            System.out.println("Return Value: " + Base64.encodeToString(ciphertext, 0));
            return ciphertext;
        } catch (InvalidKeyException e) {
            e.printStackTrace();
            return null;
        } catch (NoSuchAlgorithmException e2) {
            e2.printStackTrace();
            return null;
        } catch (BadPaddingException e3) {
            e3.printStackTrace();
            return null;
        } catch (IllegalBlockSizeException e4) {
            e4.printStackTrace();
            return null;
        } catch (NoSuchPaddingException e5) {
            e5.printStackTrace();
            return null;
        }
    }
}
```

Smali code

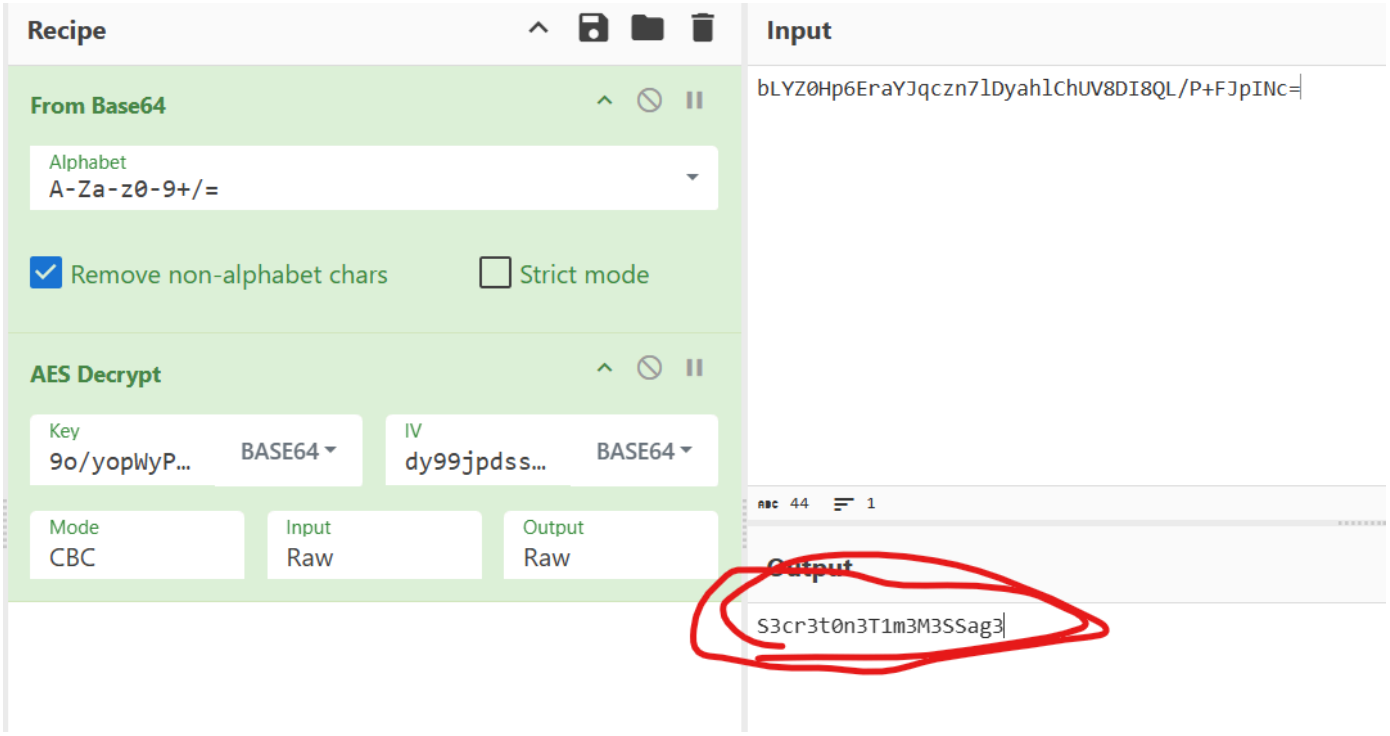
AESCrypto

```
103     invoke-virtual {v4}, Ljava/lang/StringBuilder; ->toString()Ljava/lang/String;
104
105     move-result-object v4
106
107     invoke-virtual {v3, v4}, Ljava/io/PrintStream; ->println(Ljava/lang/String; )V
108
109     .line 22
110     .local v1, "key":Ljavax/crypto/SecretKey;
111     const-string v2, "AES/CBC/PKCS5PADDING"
112
113     invoke-static {v2}, Ljavax/crypto/Cipher; ->getInstance(Ljava/lang/String; )Ljavax/crypto/Cipher;
114
115     move-result-object v2
116
117     .line 23
118     .local v2, "cipher":Ljavax/crypto/Cipher;
119     const/4 v3, 0x1
120
121     invoke-virtual {v2, v3, v1}, Ljavax/crypto/Cipher; ->init(ILjava/security/Key; )V
122
123     .line 24
124     invoke-virtual {v2, p0}, Ljavax/crypto/Cipher; ->doFinal([B)[B
125
126     move-result-object v3
127
128     .line 25
129     .local v3, "ciphertext":[B
130     invoke-virtual {v2}, Ljavax/crypto/Cipher; ->getIV()[B
131
132     move-result-object v2
133
134     sget-object v6, Ljava/lang/System; ->out:Ljava/io/PrintStream;
135
136     new-instance v4, Ljava/lang/StringBuilder;
137
138     invoke-direct {v4}, Ljava/lang/StringBuilder; -><init>()V
139
140     const-string v5, "IV: "
141
142     invoke-virtual {v4, v5}, Ljava/lang/StringBuilder; ->append(Ljava/lang/String; )Ljava/lang/StringBuilder;
143
144     move-result-object v4
145
146     const/4 v5, 0x0
147
148     invoke-static {v2, v5}, Landroid/util/Base64; ->encodeToString([BI)Ljava/lang/String;
149
150     move-result-object v5
151
152     invoke-virtual {v4, v5}, Ljava/lang/StringBuilder; ->append(Ljava/lang/String; )Ljava/lang/StringBuilder;
153
154     move-result-object v4
155
156     invoke-virtual {v4}, Ljava/lang/StringBuilder; ->toString()Ljava/lang/String;
157
```

the result we got IV and Secert key

```
→  Downloads adb logcat | grep -i system.out
04-24 10:12:42.901  4702  4702 I System.out: Encrypted Text
04-24 10:12:42.902  4702  4702 I System.out: Param: UzNjcjN0MG4zVDFtM00zU1NhZzM=
04-24 10:12:42.903  4702  4702 I System.out: Key: 9o/yopWyPzAh9pU6lg81rCF7C4+8bOO0I8QeyqypsKY=
04-24 10:12:42.910  4702  4702 I System.out: IV: dy99jpdssNYFA76mXAGvXg==
04-24 10:12:42.910  4702  4702 I System.out: Return Value: bLYZ0Hp6EraYJqczn7lDyahlChUV8DI8QL/P+FJpINc=
```

take those and go to Cipher Chef


Recipe
From Base64
Alphabet
A-Za-z0-9+/=
Remove non-alphabet chars
Strict mode
AES Decrypt
Key
9o/yopWyP...
BASE64
IV
dy99jpdss...
BASE64
Mode
CBC
Input
Raw
Output
Raw
Input
bLYZ0Hp6EraYJqczn7lDyahlChUV8DI8QL/P+FJpINc=
44
1
Output
S3cr3t0n3T1m3M3SSag3