# Espresso

By:

Deepika,
Nikhil &
Rohit

# What is Espresso?

# Automated UI Testing

# Agenda...

- What’s Espresso
- Similar Frameworks for Automation
- Why Expresso?
- Advantages of Espresso
- Set-Up
- Hands-on with API discussion
- API cheat sheet

# Espresso

- A testing framework for Android to make it easy to write reliable UI tests for a single application

- Developed internally at Google and open sourced in 2013

- Espresso 2.0 released in December 2015 and part of the Android Support Repository.

# Similar Frameworks for Automation

- Robotium
- Appium
- Calabash
- Selendroid

# Why ? and Benefits

## Problems Faced during automation?

Tests are flaky!

Testing framework is complex!

Not easy to write tests!

Tests are slow and lasts hours!

# Why ? and Benefits(contd.)

- Easy
- Fast and Reliable
- Backward Compatibility(Froyo is there)
- Automatic Synchronization
- Supports testing activities outside apps
- Test Coverage

# HOW ??

# Writing Espresso Tests

# Revealing The Formula !!

Do Something

```
onView(ViewMatcher)
                    .perform(ViewAction)
                                        .check(ViewAssertion);
```

Find Something

Check Something

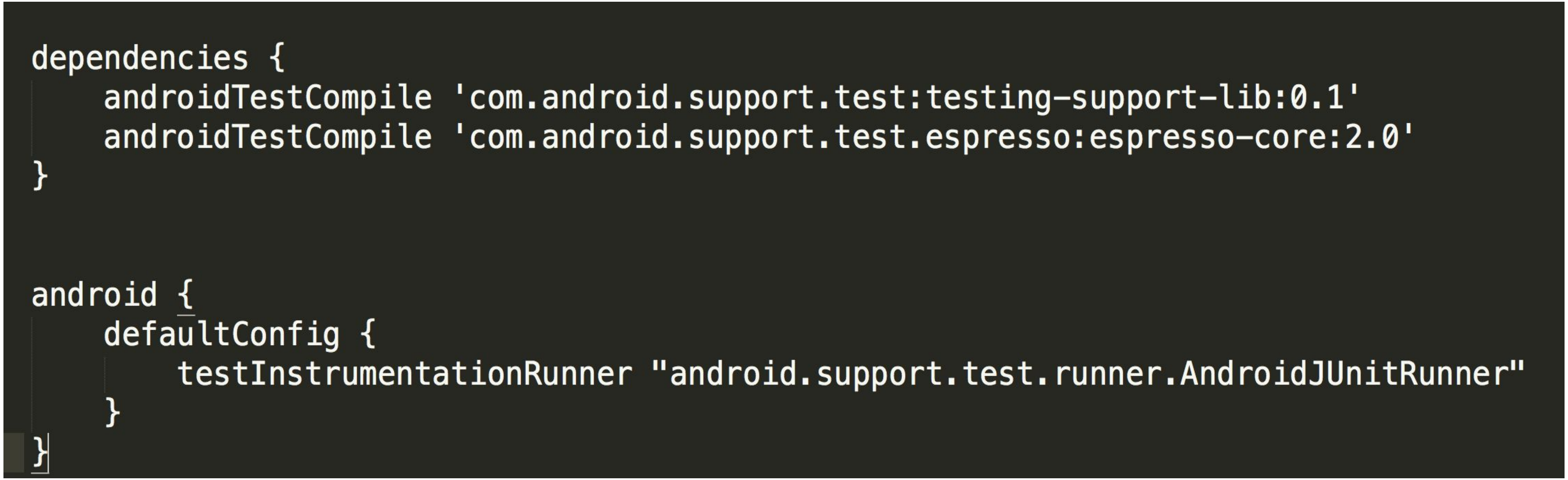
# Example

```
onView(withId(R.id.add_expense))                              //ViewMatcher
          .perform(click())                                  // ViewAction
          .check(matches(withText("Add Expense")));   //Assertion
```

# Setup- Tools

- Android Studio with Android SDK
- Gradle
- Espresso
- Repo : git clone https://**github.com/ajitsing/ExpenseManager**

# Setup- Espresso with Gradle



```
dependencies {
    androidTestCompile 'com.android.support.test:testing-support-lib:0.1'
    androidTestCompile 'com.android.support.test.espresso:espresso-core:2.0'
}


android {
    defaultConfig {
        testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
    }
}
```

# First Test

- Simple test

# Second Test

- Workflow Test

# ViewMatcher

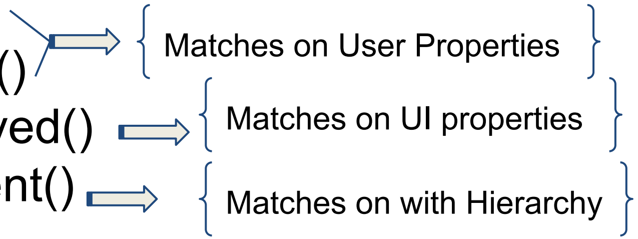
Entry point to interactions with views is via :-

- **onView**(Matcher<View>)
- **onData**(Matcher<? extends Object>)

ViewMatcher - allows you to locate a view in the current view hierarchy

Matcher:

- withId()
- withText()
- isDisplayed()
- withParent()
- etc



Matches on User Properties

Matches on UI properties

Matches on with Hierarchy

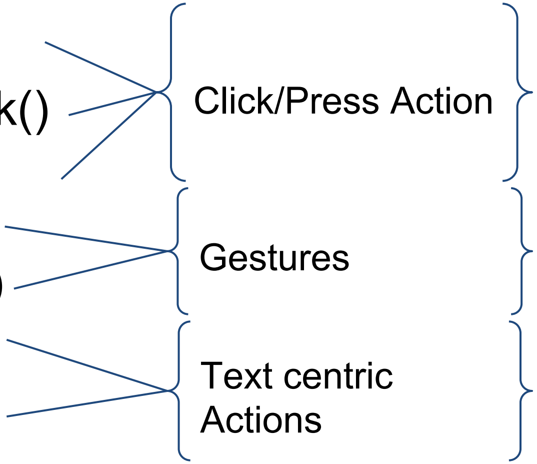
# ViewActions

<u>ViewActions</u> - allows you to interact with views

- perform(ViewAction… viewActions)

<u>ViewAction</u>:

- click()
- doubleClick()
- longClick()
- scrollTo()
- swipeLeft()
- clearText()
- typeText()
- etc



Click/Press Action

Gestures

Text centric Actions

# ViewAssertions

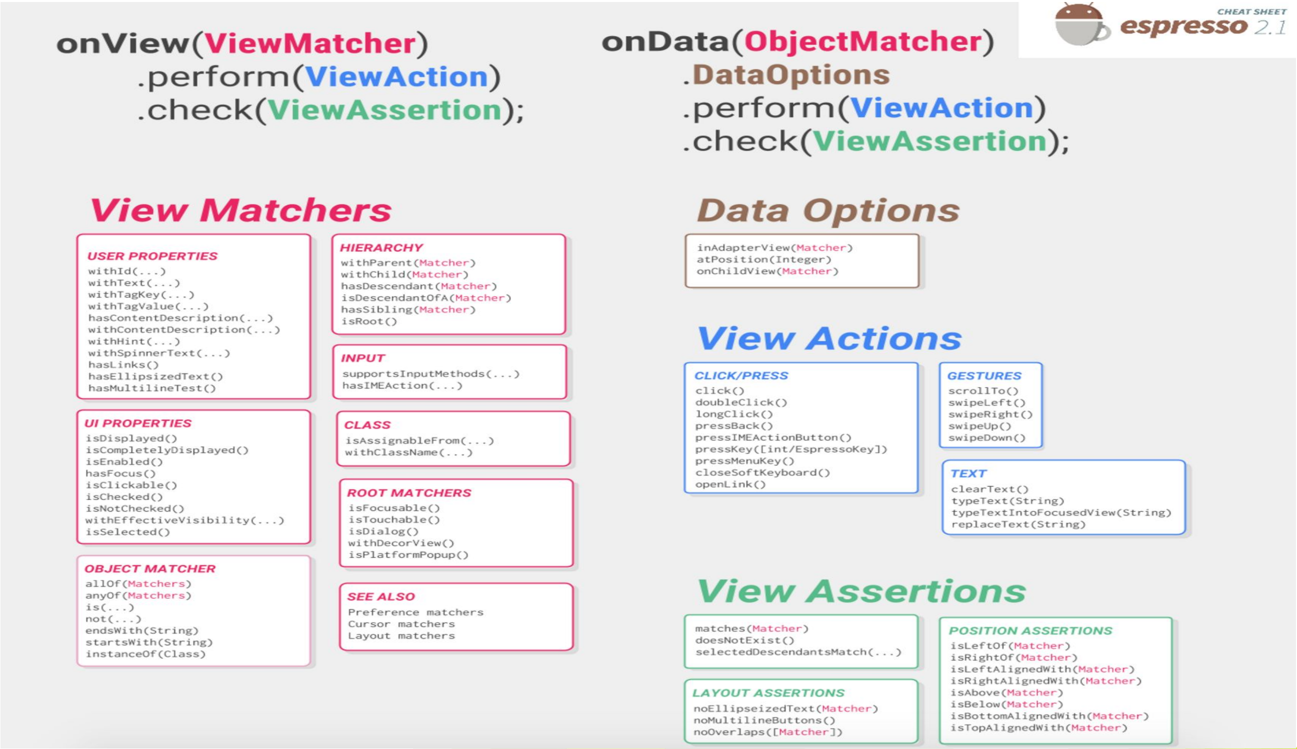
ViewAssertions – allows you to assert the state of a view.

- checks(ViewAssertion viewAssert)

ViewAssertion:

- matches(Matcher)
- doesNotExist()

# API Cheatsheet



CHEAT SHEET espresso 2.1

```
onView(ViewMatcher)
        .perform(ViewAction)
        .check(ViewAssertion);
```

```
onData(ObjectMatcher)
        .DataOptions
        .perform(ViewAction)
        .check(ViewAssertion);
```

## View Matchers

### USER PROPERTIES

```
withId(...)
withText(...)
withTagKey(...)
withTagValue(...)
hasContentDescription(...)
withContentDescription(...)
withHint(...)
withSpinnerText(...)
hasLinks()
hasEllipsizedText()
hasMultilineTest()
```

### UI PROPERTIES

```
isDisplayed()
isCompletelyDisplayed()
isEnabled()
hasFocus()
isClickable()
isChecked()
isNotChecked()
withEffectiveVisibility(...)
isSelected()
```

### OBJECT MATCHER

```
allOf(Matchers)
anyOf(Matchers)
is(...)
not(...)
endsWith(String)
startsWith(String)
instanceOf(Class)
```

### HIERARCHY

```
withParent(Matcher)
withChild(Matcher)
hasDescendant(Matcher)
isDescendantOfA(Matcher)
hasSibling(Matcher)
isRoot()
```

### INPUT

```
supportsInputMethods(...)
hasIMEAction(...)
```

### CLASS

```
isAssignableFrom(...)
withClassName(...)
```

### ROOT MATCHERS

```
isFocusable()
isTouchable()
isDialog()
withDecorView()
isPlatformPopup()
```

### SEE ALSO

```
Preference matchers
Cursor matchers
Layout matchers
```

## Data Options

```
inAdapterView(Matcher)
atPosition(Integer)
onChildView(Matcher)
```

## View Actions

### CLICK/PRESS

```
click()
doubleClick()
longClick()
pressBack()
pressIMEActionButton()
pressKey([int/EspressoKey])
pressMenuKey()
closeSoftKeyboard()
openLink()
```

### GESTURES

```
scrollTo()
swipeLeft()
swipeRight()
swipeUp()
swipeDown()
```

### TEXT

```
clearText()
typeText(String)
typeTextIntoFocusedView(String)
replaceText(String)
```

## View Assertions

```
matches(Matcher)
doesNotExist()
selectedDescendantsMatch(...)
```

### LAYOUT ASSERTIONS

```
noEllipseizedText(Matcher)
noMultilineButtons()
noOverlaps([Matcher])
```

### POSITION ASSERTIONS

```
isLeftOf(Matcher)
isRightOf(Matcher)
isLeftAlignedWith(Matcher)
isRightAlignedWith(Matcher)
isAbove(Matcher)
isBelow(Matcher)
isBottomAlignedWith(Matcher)
isTopAlignedWith(Matcher)
```

VODQA™

ThoughtWorks®

FOLLOW US ON twitter

Like us on facebook

Like us on facebook

FOLLOW US ON twitter

VODQA™

ThoughtWorks®