

# Testing Coroutines and Asynchronous Calls

## We'll cover the following ^

- Making asynchronous calls
- Finding the class name
- Mocking the class

The `getAirportStatus()` function is making synchronous calls, one at a time, to the `getAirportData()` function. If we receive a large number of airport codes, then making blocking calls, one by one, won't be efficient. If we make the calls to `getAirportData()` asynchronous, using coroutines, then we can get a better throughput. Of course, we'll have to test first and then write the code for asynchronous execution.

## Making asynchronous calls #

For `getAirportStatus()` to make asynchronous calls, we have to do three things. First, mark the function with the `suspend` keyword. Then execute the body of the function in the context of a `Dispatchers.IO` thread pool. Finally, embed the calls to `getAirportData()` within `async` calls and `await` for the results by applying the techniques we saw in [Chapter 17, Asynchronous Programming](#). We'll write a test for each of these three steps and then implement the code.

The instant we mark the `getAirportStatus()` function with `suspend`, the previous test will fail compilation with the error “Suspension functions can be called only within the coroutine body.” To address this, let's go back to the previous test and embed the call to `getAirportStatus()` within a call to `runBlocking()`, like so:

```
// AirportStatusTest.kt
runBlocking { getAirportStatus(input) shouldBe result }
```

Remember to import the necessary coroutines library for this code to compile:

```
// AirportStatusTest.kt
```

```
// AirportStatus.kt
import kotlinx.coroutines.*
```

Now we can mark the function `getAirportStatus()` with the `suspend` keyword.

```
// AirportStatus.kt
suspend fun getAirportStatus(airportCodes: List<String>): List<Airport> =
    Airport.sort(
        airportCodes.map { code -> Airport.getAirportData(code) })
```

Before we can make an asynchronous call, we have to decide which thread pool to run the request in. Since `getAirportData()` will be making a call to a remote web service, it's appropriate to run the request in an IO thread pool. For this reason, we'll embed the body of `getAirportStatus()` within a call to `withContext()` so the code will run in the coroutine context we provide to that function. In short, our test needs to verify that `getAirportStatus()` makes a call to `withContext()`. How in the world can we do that?

We can write an interaction test by mocking the `withContext()` function. But that requires a bit of digging in. The `withContext()` function is a top-level function defined in the `kotlinx.coroutines` package. Sadly, we can't simply tell the Mockk library to mock the `withContext` function, as it won't know which function we're referring to. Even though Kotlin has top-level functions, at the bytecode level they don't reside directly in the package, but within a class. We have to track down which class, in the bytecode, the top-level function `withContext()` has been compiled into and ask Mockk to mock that class—yikes, that's going to take some effort.

## Finding the class name #

Visit the [documentation](#) for the `withContext()` function, and click the source link that is next to the return type `T` of the function. You'll notice that the file that has this function is `Builders.common.kt`, and in that file `withContext()` is defined as a top-level function. Now we have to find what class this code is compiled into. For this, you'll need the super-investigative skills of Sherlock Holmes and the `jar` tool that is part of the JDK. Download the jar file `kotlinx-coroutines-core-1.2.2.jar` by clicking the `jar` link in the [Maven repository page](#), and use the `jar -tf` command, on the command line, to find the class name. If you're on a Unix-like system, use this command:

If you're on Windows, then use the following command:

```
jar -tf kotlinx-coroutines-core-1.2.2.jar | Find "Builders" | Find "common"
```

The fruit of the that effort is the output:

```
kotlinx/coroutines/BuildersKt__Builders_commonKt.class
```

This tells us the code in the file `Builders.common.kt` is compiled into the class named `BuildersKt__Builders_commonKt`. That's the class we should mock to replace the `withContext()` method with the fake implementation in test.

Let's write the test for verifying that `getAirportStatus()` is calling `withContext()` with the appropriate arguments.

```
"getAirportStatus runs in the Dispatcher.IO context" {  
  
    mockkStatic("kotlinx.coroutines.BuildersKt__Builders_commonKt")  
  
    coEvery {  
        withContext<List<Airport>>(  
            context = Dispatchers.IO, block = captureCoroutine()  
        ) answers {  
            listOf(iah)  
        }  
    }  
  
    getAirportStatus(listOf("IAH")) shouldBe listOf(iah)  
  
    coVerify {  
        withContext<List<Airport>>(Dispatchers.IO, block = any())  
    }  
}
```

AirportStatusTest.kt

## Mocking the class #

Since the top-level function `withContext()` is compiled as a static method in the bytecode within the class `BuildersKt__Builders_commonKt`, we ask Mockk to mock that class. Then, using Mockk's `coEvery` function, which is like the `every()` function but for mocking functions that use coroutines, we mock the `withContext()` function. The `withContext()` function takes two parameters: `CoroutineContext` and a lambda to be executed as a coroutine. We pass `Dispatchers.IO` for the first argument context and Mockk's special

`captureCoroutine()` as the second argument block. As the name alludes to, `captureCoroutine()` serves as a place holder for the coroutine that is passed to `withContext()` by the caller—that is, the code being tested. In response to the call to `withContext()`, our fake implementation will return a canned response of a list with `iah` Airport.

After the `coEvery()` call, we call the code under test and verify that the result of the call is the expected list. Finally, we verify, using `coVerify()`, that the `withContext()` function was in fact called by the code being tested.

Running the tests now will fail because our `getAirportStatus()` function isn't calling the `withContext()` function. Let's change the function to make the test pass.

```
package com.agiledeveloper.airportstatus

import kotlinx.coroutines.*

suspend fun getAirportStatus(airportCodes: List<String>): List<Airport> =
    withContext(Dispatchers.IO) {
        Airport.sort(
            airportCodes.map { code -> Airport.getAirportData(code) })
    }
```

AirportStatus.kt

With this change, the test will pass. We're now ready to make asynchronous the call to `getAirportData()` within `getAirportStatus()`.

```
"getAirportStatus calls getAirportData asynchronously" {

    mockkStatic("kotlinx.coroutines.BuildersKt__Builders_commonKt")

    coEvery {
        any<CoroutineScope>().async<Airport>(
            context = any(), block = captureCoroutine())
    } answers {
        CompletableDeferred(iad)
    }

    getAirportStatus(listOf("IAD")) shouldBe listOf(iad)

    coVerify {
        any<CoroutineScope>().async<Airport>(context = any(), block = any())
    }
}
```

AirportStatusTest.kt

In this test we again mock the `BuildersKt__Builders_commonKt` class, but this time

we mock the `async()` function to receive any object for context and a coroutine for the second argument named `block`. In response to the call, we return a `Deferred<Airport>` object that holds the `iad` mock instance. After the mock is arranged, we call the function under test and verify the result is what we expected. Finally, we verify that the `async()` function was called by the code under test.

To make the above test pass, let's modify the `getAirportStatus()` function.

```
package com.agiledeveloper.airportstatus

import kotlinx.coroutines.*

suspend fun getAirportStatus(airportCodes: List<String>): List<Airport> =
    withContext(Dispatchers.IO) {
        Airport.sort(
            airportCodes.map { code -> async { Airport.getAirportData(code) } }
                .map { response -> response.await() } )
    }
```

AirportStatus.kt

Instead of making a synchronous call to `getAirportData()`, the `getAirportStatus()` now wraps the call within `async()`. The call will now run in the `Dispatchers.IO` thread pool. After dispatching the calls to `getAirportData()` for each airport code in the given list `airportCodes`, the `getAirportStatus()` awaits the response using the `await()` function call on the response returned by `async()`, the `Deferred<Airport>` object.

Run the test and make sure it passes.

```
PK
  A  META-INF/ PK
  A m+>=@ ? META-INF/MANIFEST.MFóMìĚŁK-.ŃK-*îĭî³R0ô3ààòì-ÈÍÍMÍ+I,
êdāăZ),. %!ă* % %ăñrñr PK
  A  org/ PK
  A  org/gradle/ PK
  A  org/gradle/wrapper/ PK
  A ãùf0 è - org/gradle/wrapper/BootstrapMainStarter.classV[W0Wp d c@
?(âLÊxKÆ%,.à2®(ĀU±L+!£`³b¹ÖëBòvædde(èÂ%g±(ãĀNĀð\ n
wKb¹UxwW{x pbe±Ī+8Ī9gX;ĀJ$XI»@0°añÛj;óú²IhĀîæφîâ%. ŰŰĒĀİØN1]tôÉÓ÷%\æNú
ŰāñÊ`JĀ·øNø=Āāÿ%]û·èC°A GÉv-µ6JØµă! ŰD²ñzL7{ª°Űφ!Ír r í+r7û°âðRóE<´Zx·»xu[43Āİ"m
vâ«pç%ĀŰR|»mĒĒ_ăá:MŰxçĀñĀ0aµ` ZP70Ø,m±I%hØDë}°ú1Ē/ÿ^4ùª [ET0%,1G[ J Djj\q. |d"¹¥
D000"-2f0x' 0000-aÇi00r00r;ñĀ5D²ëkèXCç`'%. Ō°3.EŌv00āāèñîĒēφŸŰö0EāF0h.4Lî{oĀ¥;{,
&ĒĀñ-fxûĒÉè÷0æeİ0NèèSgýgĒ*Bİé'èİG÷añ0$ 3RtèG0éN 0jè00`âàcDââ_İiŞçÿ<Āj\`?àj00x ý
  A hQp}ç ò # org/gradle/wrapper/Download$1.class}M
Ā00ßh5µ*v/00x0%?àB\X0000T{700ĀC)âŌæ%÷^iÇĀC0Ē0 $g{sŰ0Z0Ē[ŰmE%,Ā]0&;iè0£²ú0
  A Çâî00 L 4 org/gradle/wrapper/Download$ProxyAuthenticator.classS]0Ā=³-1]ŸXWmEä
ó0nb!Ā00İ00eŰv00ocĀĒ00ñga0êç ?;Ā0,Ē0`X cñº%#0%QĀ«Ē*0Ÿ,
Ī¹LĒF0φŰ00{ŸµŸnX0U¥bp_Ē±Pă·Z0Ā0Tō *X-0AGñfs`àĪĀ0ăµ¥³%00°0DİHUĒĒ02fXóó÷ü00b;ă 00»
  A 5Ÿ<0Ā 00 ! org/gradle/wrapper/Download.classY |\ÇŸÿœĪi%ŞŌŌá0¹²NıXŌi;00Ē00K%dĒ
```

ÈrYaÊJµè\*CV²Æ□²)kYgĖú□d□)□M¹Eí□;²ÊÉ□[M¹Í□Í|11âvSŊ□Đ 6øfHE!M□«+□pŪĬ,□J9)A``ĬM^□9□ö&□□□□µ½¤+  
Ä□□□□äëN°□m□°=□wŪ.¹öô´ǎŋuëI□18î±\$□ú□%|ÇMǺ □ Áý´□□p□Tª;□ð□□Ū\$ÿŪ>□□□□ñ¤=Ö.□□;□î□%b□~AGǺ□□,6  
0%#ç¹HEJŊîiü□Ūn<|,V□q□□)Ĭ,\$□1 #Êîêƒ□□2»ð□£²K°□E=□□Ū□3XðĀâ□RjŊö\B#□9ÍM/%]Z□\*ŪNg«`ECŪ□□□6ă\$Æ□  
Ê□}|7t.□+□}óĬ□ĀxĐ□²%□&□ñT□□r\□ð□ðē□Ā□□0b8Ĭp □□c-|□(□□□O□□□BH%□0□□□□p|s\_>#ð□÷2fð>İŊ üðñ[Ö  
Ŋ□`âV%□%Æ,½□□oÓ?ëUiaƒ{JT□i)□a□ZĀ£9□□iä□ú!pcǎ,\*□□°□;²±ǎ□|□aB5ð□□`!À □ c^îF□1|´-ŪivĐb+v □□ð□ÔBé²  
úf±□.\_Ōn□}|pðĚ`m□ðĀ°vÿ²□|Ê17ŌhĀî`ð□ēª=5a□ðĬ`ǎðîª□ç1u□.d□y  
+?□Mj[3,m□□□C[ƒqŮÊ«jVV]±÷Ō¬ªÖBêUĐ+°uªê=Ĭj|0ñ□`mĚ ñ□&:{ªI □ Ū@H}□V□¬ªù\$□□h □□+«2h~ǎǎ»  
ðĚù}|ǎ,□îBĬ□¹□ŌŭĐ□□ð□□Ĭt²=□.□F□□ĐáG1ǺŊ□□±ú□3°□p=K/□ ÇÓ8□□ƒ□□%çÇ1□Wq□²0ǎ□8□/á,%□□ð□□£□\$ñ  
«□s□ƒb=Ǻ½y□p\_À□ý□□Ū□□□` : wác46%]cvǺ□`f□V□□0L&%î³P)q  
pæq°ǎ` □□N4□□gp\$t´ðŌēÊǎX0sè,□wŪ8□°□□□ðĭk□î÷dĐŪî□ŪĀ~NöW6□ú28I□PWB□□+□ÔÊÀŌ□a□□ǎñ÷Íah8tJ□□ð[^|□8  
×TJGoð\ñ□H\$¤\*□d0□;¤□ǎH□sGh□□Ō□ĐiŊ=Í7p□□n|. \_xǺ□`oĬǺLK6\_°2»Ĭ^~dĬ\*}R%□□|□°;□eð□aî□ƒLŪQĐ~□Ĭe^a`ñ  
³Uð<öü Ō`□!□t□°GX}|Y}ÊŊ£Y□rŊ□□Ê□Đ□±□□0kŪB□□ĬhŪĀ□ú□ô!Ā\*z□=¿÷Ū□\_a/@ĬWµǎ\_Ǻ□%FU` \ÿ\_M□ēÿEN`ømPK□  
□□ A y□Lƒj Ê 1 org/gradle/wrapper/DownloadProgressListener.classu□Ǻ  
□@□Eî+ój□DĚöEĐ@Ū□E□□ERŪI□|□#3cp[□> □□4Ūv□□»9□×ŭñ□°ǺǺçjO□Ǻª.r%ǎðJ[□íM □ Ç□ñ,ěǎ]□□8□ökB□\_  
□□ A !9|□Ŋ□ □ 3 org/gradle/wrapper/ExclusiveFileManager.class□WŪs□W□ÿ□YŌŪ«u|iªVÍM  
t□\_2/□YŪtǎ³S%g9óy□C°SñMÇ□1íª□hñ□t«pDE@ǎv□]ŪĬ;%ð.□Ŋ"Æ□,Çð □ Ā2ù|÷□@tĚ-R°mŪrǎøjinzŌæ□-□=.´Ū3|g  
□ǎMn,Êr¹Q□□iîŌĀôtÿ1□□)B,□N□ǎ@LǺ×5|CǺ75|K@gùĐŪ. □`ĀhÊ□wL;ēQ÷ǺĀtnk□□}ƒ2□□ts¥□óĐ□&J',ŪVXjĚt³□□%s  
úð²□Wðª□i□ôâŭx□Ǻ¹cŌLŌQ°V\$□ǺCnŌ. |□×O□<iú2]6UĐŌǺ□□ó□]/ □%HŪi□`□□~`´□\$□□%□,□"p□□□ǎG□~□□□x□g4%  
□²□^2-□É L`µ\lþ´Ĭ□□ǎ(ö3%eà:°jð¹□.qZǺ/ŭ□WYĐ;A`□Ŋ}3□□öŪ□r□ŭ»µS#»Ū□□^L/xîR`□X□ □|  
Ū□F[F  
ý□f□° >#□&¤□A7□zx□ □N5@.iT3□Ĭ(íú□ǎz□\$Yð40□)Ĭ□□] □ ð³û□4%□□?□□E¹vqŸ□□□Ǻ□<İ□□ □□ǎóðLð¥□□□Ǻ  
□Wŷ□□BǺ°T□□U4]»□Ÿî~`Tô°7%□6□□□öviǎpà□7□YBZ \$ŊHè  
□Ō«@'□%c ¥;□K1Ĭ□,□3Ō1□.□ÿ;ǎcŊĀ□Ŋ□□öDQDDwp>`□ǎPöz□□p}□ŌÊ□@8B □gmĬ³\*&□\_ \$W□Ō  
ëö□+ð□kð□□rŌ}|□:F□;A?~□□Ō(µ□AZ□ÆÇ8□□ðǺñ□:ü□Æ°□q□Ǻð8Ō□□"□%ð□Q±□cǎ ĭ? □ z5)□□çq□j□□sDP□.  
□□ A □i□,y□ □□ - org/gradle/wrapper/GradleUserHomeLookup.class□SŪŪŌ@□=□\$V□□Ō□K  
□J)\$`ǺŪŌĐ□Eƒ`@`"\□@¤&EY□□±#çjƒ\_Ō□□`H}|iç;@ª:ē□K □ǎaföäiİ9ëŸ?ÿ~p□0□□□Q□Ix&c\Ǻ□ǎ□□  
B□□ñ\¬\_DĐ□) I□  
Ædè"¿□1-ð□□Y□pceq7µ□\_M/. \$□ünÆHç×Ŋ6□-u` □□°m:%=ǎ{□SúǺŌt□□BNom□é□\~ŸÊ1t,¹NǺ7□?kŪUNýið□í1□□□.□  
qhN□aē^~ëN□3B%□ŭðpđ6□ðǺĀ□Ō□jEL□□U1tS&>Ō-□psc-□ǺǺ¥ð0□N□Q□M□□}□(1□ǎ ĭǺ#□j□□□R□`□□ē¿□í□i□ið□  
□□ A À!1ǎ7 "□ \* org/gradle/wrapper/GradleWrapperMain.class□X□\□W□ÿ□,İ2□□□m□"1\*□□<ĚŌ`  
^Ěµ\ŌJ%j°×ǎ°^ŪǎǎŌē%×{ĭ½z%ær÷\_DEL3Ě²e□ŭi÷□.ð%÷%ÿ×¿÷□□=rǎ□□R¥□')ð□'+Pp□□□"□\$Ěx□□□A.ÁŌe<CÆ%`^<Ō□g  
□:ǎ□+x ^\*`\_&□□□ǎ□  
.ǎ~□`ôâU^%Z□□×Ěx□5x□□j%`Æ□□-□ □ñF□□ôǎĬ  
□x□□□ð ðz«□p&□.□{ñ□/pŌ □i□ów+,□Ě  
p□÷\*x□p` □□□I□□P°□W□L`Ê□□□W%,&ǎ°ðŭ□Xŭ°Xŭ□@ǎ×Gñ11|\Ǻ'ðI□□ǺŊe<¤ □□□°~½\$¹5ðv¬¿.çXG÷ǺŊcŸ□ú:»»\$ð  
Ō□□ǎi1□)"□□é]□ðē ŸiŌNDt;¤□Ō"G43,ǎiƒç□C%-□□ǎHǺ□@Gð□qS□□ŌÍ□v{ :ǎİ□jǎ□□ðDù+ǎi`´Đd[°ƒ□□`ÊeY  
Ŋ`□ü,KŌ1?□□□□□ǎÿ+Ō1Ōv÷t/\_□`□Ĭ□Rb¤Aw°s³@tO°¥Y□Ŋ;^6ªǺŌÍ□DT□Ys@□ǺgŮÊðHĀð+hy□□Æ□%]1□□óµēxĚŪf  
Ĭ²»æ□ðð□÷b□□Iiǎ`ǺBǎu□□\ŊYz8□ðǺŌēǎ":èC?□Kq□□ǎǎfé>\*UFŭŌI□□7LǺs□Cq×□J□S□□=~Ō□p□¿\$ð\*J□□|e%□□□Ū  
Ō□pð)H□}«}5)Ū<□5□~o  
·¤°v  
ë□`>□Ū□³□ǺÇq=½\Kc□(G=□a□Va#Ŋc□:è`Ǻð□Ça+□□ƒc□&°□÷`7îE□ĬǎV\` □°a□i°1Ç=ǎ□´c7ǺÇÈ□ŸY□\$iǎ×□J,İgÉ□□  
□ǎ³Ō-fwZLù□Ū□^AQ)tǎ;ýd□\y□NǺŌ1°`p□PK□□  
□□ A úǎ□æª Ū " org/gradle/wrapper/IDownload.classE□Ǻ  
ǺŌ□ÿİēæ□□%  
^ðbǺē°\*□AŊ{ŸÊŌ(í`Ōù1□| □JìTf \$\_ðçİŌu\_DEL X ĩ;è! ð□°□Ō<L|□□\_9Sƒd□]□~)Ō1□I□Ĭ□□`u□EQfZ□=ð,iðǺǺƒp□  
□□ A p@ǎðc□ □ " org/gradle/wrapper/Install\$1.class□Wù□□e□~pì1Ōí□|□□□Ŋ□¥ĐĬ□d{ĐŌ|  
Ê`p□~>□ēŌíñ6□m1ýaBó{<iŭ}½Bwöµ□¿ð"□Mxk□□C6Ge□□í¤1LŪ□□a3|□X1`□V`Ç□Ǻ□□8r□Wñ□Ū»r¥ □Ō6EŪ□□!□□1  
□[□,o}yŪ`□Ǻǎð□ǎ□Ō`d□ŪŌ2,£°kĚyy1,O) ðŪ6Í[zj`pzí□'Y»□□□Ō□£Ō□è~□nĬ□İWC□ÊÊ□□□Vvǎü□i;□1ðǎ°Ǻ|uwĚp  
İhx□İ□D°ó°ëið□Ū>□Yz□□Æ;¤Ǻ□□s□æp^ǺŌ. à□□~□□5%□□5ü□;ñð  
~«ǎwŌĀ«,□Ūè3,ǺŌēǎ□`ŌqŌŊĐiñ□□□ðð□pǎáuŭYǺ\_ðw  
7 □Ōè□1□íx□EŪ^3BǺæt×□;Ŋ- □□zǺ±{hðyƒ7ñ7□□po□w`zŌ, j íŭ□□¬Y2æ□H□+□«ǎB□2□ç□ú=□uǺq+'óýŌ□i□□□#□2  
¬ŭ□XŭŌ□]Ŋ□□cþ°V506\_ì□YēF\□Ō7±ðAē¬□□£YÉ{c□íF□Ŋq□+h□%□i'□ó`uA□Êði□ŌǺ/j□w19□İÊF□□□□%IBpyù□ŭ+7  
Ū&Nǎ□□H□é\$M□□¥ñŊðǺ4T□p`Ǻa□Z□DD□ŸU□x¤#ǺĬ□yē□`xd□)□ŸŌ□□□□□%bH□ðĭYŌg□□Up□İ□Y°□,M□□GĚĚ,99IV  
□ƒ\□#ðŌ(□i□ðǺŌ□Wn□ú□ð  
Ō□=ǎ]xBǺ□iB¥□Ǻ  
Æÿ□pç  
:B&°»(□□İZ÷2%î□R□)?EǺŭªaǎ□□□¥□□eŌU31?ǎĚ□□÷ð%□ \_DEL/Ǻ□pİQ□R□ü\Ǻ\_DEL□PK□□  
□□ A T`,□-□ t+ org/gradle/wrapper/Install.class□Y |TŌŌ?`³ǺÉðð□!è□□²□□ □□\$H4 HX  
Owq%□□à□  
İtŌ□.pŌ□.□□"i□□.qŌTİus)□¹°□Ěǎs□4³¥□3□□ð\□ç¹x%Ǻ□dÆB7ÍǎS¤□□□+ǎ¹XŌ]"c\$°)□□°ð4y.ĚæJ^îæ\*°vǎFǎ□nZ  
g□□3]|`=ð  
7(%ĚǺ"|□t¬qŌ2n□f□^ǎu

~Wx%❶iJZnr0J>Gá□n^□ê+uk.P\$□çJ3GáInjá-  
knnf□4~7□□YMM□êâ□yJJ3ÇIm□pñy.P\*1□³9Āa□.âavy□/◁□□¹@U□ÚÊÆ0ú□ã01S.ĀMA®□!S&v)%)já`»é|□ÚÎ□È□o□%□  
;+²su □bª\_f0Ŭü□@»|C□À□P.Ēb  
i□Y0è□□;P4□\$□MR□É{1mz'□□ðW□K¶□ä³Bp□É0ð□i□G□FWy=°□ V□Zð□!+[Āè¹@=jĀÍq□tu=□Q□□□İñ□□□z0Āãð¶æ.  
\_□ãŬ,ö□□Y0|ú¶ÆĀnó□□2|L□à`JÈ²□ô«ù□.□4Q□□□Ā£>YĀú0%³}÷□°ß□q□-.A:of`ôK«ðā Ō□hkF<ª□Ōñà□Cd□¥c°Q0  
iaø□□□□hĀ«□p□ô□□m□□□□,ð□È□□5h²8Ŭ`□u □ÑÊĀD%□7□□XÆ□Ē×□Ùϕ:à;fcF`□`□z□É□□o□@ŭ0□]&Y□□=  
æHĀ ñ□\$□Uiý□ç□T@□}^□±□p<¤\$□μİdĒi×%[(Ŭ□□/oèi&□  
\_jðw%❶!P□iİJ□FLe□□□Tfö,èAs6□\¶ hñŬŬ□>k0,|°□¹+□èN8øduÎ□f¶4sú.□-[eN□É2&%Tu□\$4G□-9qBé00□ŬŬcÓ:F  
□□P0)□%\$0-é2□v□²i÷Up□\_ð5|Ā×ÉÇð\*ÿ□oPùG|□ðñÑ□□>[□İTùF% "¤K□.n\YY□jÊã□óİðİðÑ4G□□ĒÊ·ð\*ßÆ·  
Ō□A 1·K,÷¶D#|`æ5ϕñ□èÈ\*□ĒD0øzAéÉYtáÍ□ā□□ñð+U□YÝ)□ŭ□ð[□i(□ñV□E0□□□{Āö«|□□\*Ŭ«ðĀb0G0²u□`~□Fμ.  
İ□%□Ji0;\*,÷□□ðÿ5i□!u□ēİIŬŭ□àc0`è□P □O□,éCb,÷Ā□□Wé□ð/□?à«Tp□oGð<z-G□□i□□Fy7çĀ□ĀI□□æK□Ē□;ä0  
[□=e)§~>â|1□□ävrB·□æ□äV6ç"Ā, j□+L, J`²à□6ä°ð□□Ōà|ÿ<&C70Ñ.\_A¹□0;□|A-□āĒ²Ŭr□#X4/R  
□Ā,a□Xu²ø□□□□¶;ù0Wd0b□dª0Ā0ĒĀ %0,□□  
f`□□£|&0Ŭ□□ϕth□□ù□aŬD□Z□ÍK□□□o□Çg>¹Ē~cú□□WB0è□;³3i}□□□f□Ŭ□I□ªO`~¥□□1Pt%á WO~□%Vib¹0  
ÔeY§9C[ {J0?□□\□□§Ē:à°ēhxu0□□ñ□£□Q[§h;□□□□<t□}□□.ĀW□]D9rBÇ{□□×Ī`□Ŭæ□\$n<Ç`îāð□`WāēbP°ā9%`□²  
i□Pa□¤q?Ā!□□Āh ĀĒ□ā0ĀĀ¶□T□E İŬ00×□¶bv□ç\_□â. \*xh;04¤□Ō`b□Ī·}Pb□V®¤nē□rö£W5□`=□ð#jĒ|~#4|j□S  
ĀM□ðð□`ϕ□V6ðRm□|□ñCgðP□D□z(°jB□jªp=My□%gμ□w□«±□İŬ□x\$□dMĀ.□d□]WDæý`%□6HÇŬ□\!b³0KM xNĪÆnú□ □  
Ñūalü<□ē□°Ā□p²□□□□§9ð□°8□Ŭāc0□é0□b□Āªā□□Ōİ 7é-S□□);,¤?□v□□□□İs□jðŬ<ù3□É□□6□ùùel.nuM/μJfnª  
1 0□□qC5£  
□□1□P□ðP;S!é2z□□P□□□°?□¥□□□>DFú□ô>F,ÿŬ0>□â>Aðó□SŬk □ét6ý□p  
□mBað□ý□□□%ö>ý□;Cg)e\_□□V□»□t□ôWāĒ□ç□S~Dÿ□iĀ□□)□0(□>V`ō□□+□İ£Ŭ¤¤á\Fð□□dö|FŬY}°ª#5□□æP□,ç  
»2□3+06\_nİ□0İ -p6»3)□30>□2(□□É|VQ0J□ĒB□¶pYĀ□éGĀÿ□PK□□  
□□ A ð;:öo□ 4□ □ org/gradle/wrapper/Logger.class□□ko0`□Çÿ□□Unc□á@0&s\*□]7%□11\$KĤ0°°²w  
î%q□□□<Lâ□rI□ðX□`0<%N`×□□İ0`□□□F \□□ā\□c□ñi~¹b@□0□ϕr□J[ }C#qi0□□`□İ□Ç4İjŬ□□□zŬ|æw/İ  
□g□□PPϕð□pz\*□|□Ŭ□İ{QLæ|Ā5□ðó □ !1D80>□c□□±I! [3□İ□□j□Jg□`ÇXó&0□jovĀHIFiİ□□ó□  
Qª¤V÷(μI0□□ð)rm□/□,°0°ŭ□□ñĒēāē□\$Pª¹\*qw(¤PÝ&nŭ\_Ÿ□7Ŭ□□□PK□□  
□□ A on□□¶□ `□ 8 org/gradle/wrapper/PathAssembler\$LocalDistribution.class□R[K□A□pİf³«éV  
Wdel.i<□Ŭvt0n□□&< }□İP□\_> vðİŬç-□-□,ð□ßDdel PK□□  
□□ A □J0â□□ [□ & org/gradle/wrapper/PathAssembler.class□Vic□W□=cK□Y□Ç□b'μc□Ā1,Ŭ□ 4@  
ϕ0×ð□z\*`[¹0<X¹Q□m□□«äè□³¤çĒ□□#□çf□□□X`□:□³ð1pX,□ð¥Ā□hxfİBēa3oðDèA02□□  
□ā□K  
□□□,ŬŬ\$MĒ□)o-□ð□%□7□\_È/é¶)k□3à-□dY? ]°s)□\*uĒ0□EĀN]Ŭ□ð1zμAM□@Gāu;□=g8ŭā±□odē1o!·×İ\Ŭ0Ewt0mQ□P□  
□□i□` %ª□□□]□□İ0-'¶Ŭ□□Y°%□\*pāKŬİϕ□Ŭ0[³÷.ŭl□yμlç/é¥uJ□j`Y× ¥āá¶£T0sÆ¤3J□□>00d-□m0ŬJēöbM¥āē  
»>□>ð□□□Zðİ□Ŭ□ÇÑ\_x□Æ□vëý□;□ĀP7@Ŭ□□□`ÿ0+òŬŬŬ}ð0Ēāİā□□ŬĒĀ0□%v□□□Jc□□□YÝj¹`üē,0  
e}´P<İİtĀ03□h0×è;□ēvā□4@×ñV>ð\$□56g`□É□Ŭé□K0%°ŬE□Ç0□q¥çŬŬ<J0ð¥□0%²□□%²;ŗ·J□±á□ñ²]`ª□C□,á□;□İ  
Wñ²□Ēr□\i□`E□pè□p` %□□óĀ|YĒùt£ùtji`~(´cĒēİ`w`□İİ;=□q00□A`#□□□ē□□°axμK□é□k·1Āð°Vð.2□M<u□  
°□ÉÇý□Ŭ0μ%ß□>İ³<)6{□§ŭ2C□□□□}□□Ŭ□ē□<ŬĀýuðāð□ð1Hÿ4z0A□ϕ0;□□□□<□□b□ñ□§ŬJ0x¹□K³Ē¥□Ēē;□Ŭj□□~  
□□ A □□İēĀ□ | 0 org/gradle/wrapper/SystemPropertiesHandler.class□VÝ□ŬŬÿ]□°, -ð□0□  
f□h@0□99%-Ā%□ñ T,H0;0XĀR=.â□□Ēr□,□«~%□Ā{x\_□□¤,&ñŬ0c□□□□□æ0r|!:}mn~v□":□»Bā0□vC□\$53□□;¶a&#□  
-nyμ|/â°#·;ÿ□ð□)āŬĀ0□yeðç¤2FIiŬŬÿ&ñ°0Ēē«,=A□□5ð²ñŬ¶i□āð%´Tā□mªMēfR□0μ□/}6·□-Æ·□bU ñ\$  
F□1μWĀ□Ŭ□>□Ē  
□`=□ÿ□□ú□□`Spñ□L<`|×Y□\_b□qW»ŭ□PK□□  
□□ A ª=0□°° ?□ - org/gradle/wrapper/WrapperConfiguration.class□□m0□A□Çg;ðŬāJK□□|□□i□ā  
"  
HUL□C□□ß□-□āH¹□»«&~\*M\$8%ð□0j□3w□□e□□&3³36□ýíiðİß\_□;□ □Ē:¤Ā0àa□æ`@E□G xLá<□□20Ē<  
÷ni00□÷□Z□d,8t|□āZŬkZM□□`lè□ÇONİŬŬŭ□Ŭ0Éiv<N,□□nŬĀú¥□gJeùİi\_n□.Jİ!004\*:Zp1ŬākĒŬªæBéj`jttWò□(;  
□)%Ē 2`□)0A□e□E0,ð É □²□\$!□<W□□j□□²□□μ□ y□2□İĒ U¥Ē5HR□Ŭ@İ□□ □□2□jfy□PK□□  
□□ A Gü`□□ □□ ( org/gradle/wrapper/WrapperExecutor.class□W÷□□ç□ÿ□-ùāX2□1□Æ□□1□□+¤0  
□8\$@ĀŬüāq□İ□ā□2P¥āP □0(□□s2P£`!İ0□ç{yx□□İç□□āāI|~0□□ñŭ□□|□W0□ðİçyx6□□ðŭQ&□□W□□á□<□□□0□ð□  
8%g□9Y²=/a0£zĀ62İ□ā90; ĀuĒē□p□□NĀ1ç=q0¤İi□1.aC□i□.□[Z:£·□²`□Ÿk?İİ□□\$`i¶aféRÝ□i□□āzĒPoX4□Ŭ□□  
h08Y[ç#□00□H M·□(aÉ`□p-ĒĀW□  
□Āvð□\$□ç□ŬŬ`Pm-□P-Ŭ□?}U·b§ú4FŬásiN□0ú□VĒŬē\;İ□é□/TĒ0□,Pj§0(□H□□dH□W)`(±MĒİ5ðyb-Ŭ«\$E□;μ,¤ϕ"6²  
o±J□Vð;Ŭ6w»±0İtĀ□\$Ŭ`°0-□□T|□ßj;Ē}²y □μ□I%Xj□Z]□"%ðĒ□)□Y;6; >□6i□5i□~□ªg#□M «del□;□□İā{,İ□5□;-  
□u=□£\*°āŭ\$§İ)0□-`«□0^Q□ÇI□?dİe□ İ□Pñ#æ01□ē%Āā□F/Q²00ā`□Æİ□?e6ð!#□¤(è□□1ðçe□□0□L0%□ē|  
Y:bā□qJp»q(□,Kq6±Ŭ·ßb¤□□j &□□ðŬ□□İİ□};dm\$□□?□JßpðE0¤□□)°@aŬ|`□Ā□□fŭhr%□□Qß□□æ□|2&Ŭ0xēg  
C±□p!İ£□BμhĀ^ŬG»×ð°  
□h;İ³\_Fŭ□İ~□İ;=ŭU`ßİŬ70%Ē³\_□0·kZ□¹c□□z8Í0Iv□z□³□v□`j□ð0□ix□ŬĀ±pð,□s□□İϕf□ð`□ē§ñNa\□□□ªs o□  
ç-I□tİV9gw□□`3Xİ□6;□k□c□s¶09[ `İ0\_@c□pĀ□ÇqvCŬ,bW/□p□<□g□u□ð□E4n`k0ðÿĀÍXBÿ□□è□e□Ŭ(ŬŬr□Ā})□İÇ  
ā.<□7□^~M□ð«h□ēİ7ñ.@§#Ā})~`É□ā"□āð|□1%□%□P4□-□çp÷°Xn□Ā¶Ēİē%□úPG  
□2kμ«□□Yu□□  
delē,2□&"[²r□ŬŬÉ□÷ðÆ[g°□~;/@ēā¹M□`□d0□ā°ϕ0āT¥P, Ia□ŬOs#ÍMä\0#`t[éf8`80□`□£HyCTİ!ā0ĒHs□0ð□ϕ°cð£/□  
ª EXÆ□`-q□-z3Aİİ·□D#»7è6²á x□p□¶^á□C=□n\*ēðāē`á-□ÿāÿ□PK□□



```

    A āD_   # gradle-wrapper-classpath.properties+(ĒİJM.)᠑M/JLÉİÖMÎÊä**î+ÉİMμ PK
    A      org/gradle/cli/ PK
    A Ŭü?®< S 1 org/gradle/cli/AbstractCommandLineConverter.classT]oA=□□□ēÜ"᠓μ~C?
    A xμ³X i
        ; org/gradle/cli/AbstractPropertiesCommandLineConverter.classV[WUOpN2ÉÀ0â Hı°ÔōxÄK
ðøHÁÇÈô` È=Å
V»ð□□□>İB᠑□xAİϕ°&c□ë.cCÆ□Á□i□P□□?j c□□i%ÁĐ1-c¹%.i8 í"qB[×k□crºÆ%mqóe, □±□Bçàhùφ□Đ
F]èp¥ă᠑᠗0□□fv´}-QŎ-B"è9!U Ñ□□Ñ»□§□E#İ0ÖfİÖ³f±[-³Pv□òTä4ØðV5C%Û□ÁX3L%X□xíÚQgIW□□´-W]ÂpS
Ò*q,|□ÊÆāAÉ1\W@
μó|\□i᠑AU□□A)5"Éãä4jæä8,Õ□äF³hªJJ+g□^êz□%K□Â□□□añ7ª e<¢ fí□¥y""ß%´□;D□='pû□□□□+□;□i%E,
50ÊÄÖxE;i1g□÷Yu[së°¨ª-D:ZÜKMÍu6ò?A1©□□T'᠑[ô□'\íLz %□φ□□B□□E□1]^□%Í□□.³□õj&□z6ð□K®Z6□<ñ
□#ð0,_;□Êð□□iq□á9|.x□)ô[Aß1 □□□q"û□0à¿ú□□É□□T□□Áéq□ˆÔ᠑bù9p□N□□ēÊäR□7|Ê□3äwÈ□ó□o□□='
□□ A }İyG K 1 org/gradle/cli/CommandLineArgumentException.class□□İJ□1□Oú3£μ᠓Zm+êÂi´U
"
ÀàÄ□iÓi□Ff□Éİ¨¨âªàÂ□đ;Ă$-U´□fq□{rīwOÈÙÇë□□34K(`Ėm□u□□ç□□□öē□p=}x⁂DEèðSÂEØ=□□□.â□□T}.Øm□
□□ A ³Baú□□ g ) org/gradle/cli/CommandLineConverter.class□QMKA@□)ÖŎÆÔ¨ª'I"4
F<6¥ EQ($xB|²è²YĔvSúŬ<ø□üQâ6□□L
]XvaÍ%÷□□□βˆo ÷8wDupJp□L-,6□~/□²□ós|□yb,fā□□iu<â %*Dè<.□pid|æ□İ□n$□b&x□pw'<(a□ÑR□`□ÄqàÜK
□□ A Sf
Ö□□ g & org/gradle/cli/CommandLineOption.classV[wUUY□|□4□^□´i;j\ä"iB□μ´ ÜZ.Â⁂ĐÖ□□□ódL□|
.□A□±□A□□□□wĀh/YĔ
iEHZ□WĀ□¨□□□Âª8É□Á□C^Ü%&□KFA□-A6ó᠓na
□:|™ˆxÔTÑŎ□Ŏ□f□Jh□Ó³9Ŏ.Z□□Ý□šc¥¥;æ²©9ŬòsŬÑÃ%;¨V᠑,ªâìù[y^Ŭäó70B□*□÷Æ□âf$£□ò□i8#!Z□!/¬çŎÂēǺ
᠑c□□□
na.□|bèC□□iá᠑□÷1£à□|(j³º2>Rđ1>aÑÊ Sz□´□5 □□%xAÁ$>U□D□é%P~Ā.*8□ó
ö□□Êi%□>Ā□□İñ□,ũ¥□¨ðm□;Âôİg¥¨ĀñŬâ+Z□pEkB□ōm'
gŎ6-□θ□ë□:ú□|NV□çİ□=!!x{□Ý□LX□ˆ°e2zf}I«Á□3UöĚÉ%ÑN□μše5CC]ñZ]Aá□⁂mÑ,YC᠑YŬb×□Z.+èĔØ&%ôĀčyēb□Y´
a
¿"2□ø Á5(□,P{úc□φ□□â,□!□Ŭ¹□;□=èÆVú%□£óé$¥İj%H□□ôa□□%Dy□□/ü□□□□i;□ĐBaÊ
çŎİ□¥□□□xÉ□□dôÊ8$ó□□□□¬□đ«T□Ā□□f□+q%Yøú□£□r²*6÷:^□đ□|Ŭ%□x,w□G\□78□8Ĕ□xH□□inˆĀİv?/'□¨i□□
ÈY□rĀ□i□□Ñ᠑i□ .ø□;*ŬŬĀxÇªa□QN□□"Z□ñ□i\đ;É□©ðŎēª□,D□Eu!İzqu{9z¿ª□3uzYbcŎ»ŬU}y¥İe□òzˆ□\ý□□
□□ A üi□¨¥ â ( org/gradle/cli/CommandLineParser$1.class□□A
Ā0□Eyhμμ
vª□k□^j(□□□'□iP[ð□□ēá\x □%|,táó□□iÿxûñ°⁂8D□bD□İæfsPv □³İ4□ŎĀ%ò|□Ŏ±|Ŏð. □Ŏ□İ□q□.□Ŭ
□□ A □ĐÆN □ â ; org/gradle/cli/CommandLineParser$AfterFirstSubCommand.classĀVÈN□Q□p
AIðß²-u;j□i□ı"%Ŏİă`φA%1p3ñQ|□c□³[ X`□!ñİ9sfæ|sŬŎŬóóİxo ð□□Đ□D□-I □HHH!□@/iqôqŬ□□□ □□
â*® Çf¨□]L□□Br□ »Ĕ□s
□Z«ü(Ŭ□□K□;iL=VML□Āİ□1ª□c:N□>}ÓPıWÑ□Eà(Q□É□φ□□□□$Ŭio□□Ñâ´é|¥□ŎŬφĀ□1¨□óªĀĐQ>1rĔ«ªbíY-gqİÑŬ
(6^f□i□âe¨pfSQ;□□□!nc,ŬÆèÁeUGÉl□=ÆZ□s□ö□> w□DÈ07Rè□x□¥ŎĐð í□!po)□ú□□|=è□p□PK□□
□□ A ¥□D£φ□ & 3 org/gradle/cli/CommandLineParser$AfterOptions.class□mŎ□A□Çy{-%r-¨´ >W
wĔ\J□□I□Ĕiðdi%ômüă□ixBŎEBØrÉâ□□Mªuør$f□□□□i§-ôup~Ŏ⁂đ¹ð□F□p±ð+p□[.□5kŎ□□¨-□o0$,¬©ăŬ□□z<□Aè
q~□:5ú□h□h□pk⁂mqéóăŎĔē□□_ð2w.íc^□2
ò\|é¥G□ðŎI]ĐT□□¥□´Gxi4gL᠑□ı,~Ab□T¼äö"A;□£□Ā`□Ĕm é□<N□□x□~§Đ□BchV⁂Zâc□□"□ø□□Ck14²□□□□8□2
□□ A xÇÇà□□ □□ < org/gradle/cli/CommandLineParser$BeforeFirstSubCommand.classĀVèRŬF□pŎ□
®r|$□Ĕ□+□ö±□x8®p□ă|□[φDÆ Ŏ8ò2ĀXç(pŬaè.¿3□±□□±|Yªx+□⁂ŬŎŎZŎHˆŎk5ŎŬĔē|~Gμ□⁂□#ð□a□ö□□□□pðx□□y
6ð9C□
Jðd□zéòà>%dÈxF⁂gn□ðÇf□İ<sð□□s□□¬%aH□□ˆŬª4k⁂i_{fé<8x□mâĤJ,¨□□Ñ□ñÆĀ4Ch?Y□â□⁂fs|«@EYĐŎ.y□i□i
□D□m□.@X□FİèÇðöē□ŎŬ□□8İBŏ□sj0;□9æ□□Ā1□□□/tÑñq □Đ,Bj□Q. |□,1□Ç%ÑMg□ |yF,
□ ⁂éU□^9□Ŭ□□ĐøăĀW4~‑Éÿ□PK□□
□□ A ´*«ZM □ İ F org/gradle/cli/CommandLineParser$CaseInsensitiveStringComparator.class¥S
□□ A □è2 □□ Ā□ = org/gradle/cli/CommandLineParser$KnownOptionParserState.classYXūw□Ŏ□y
Ŭ:□□M□U□)‰Eä□:ª᠑§□=ŬŎÑm□İFüð□;:)□□xÍ;pİi□ª□h÷İYĀâ□!Y°ë,pİ=□□z³nēN□Ā□□□É□□□Ā□/İē9³i@ ĐcÆ
t.ˆ°çŎ-İ□k]T□□hYLu□¬[X[£□□²^G³TÇ'hJ□®j□□ĀmĐİ!Ŏ□i-¥X□«Ŏ«7â□:{#ó□□□Q□aR□ob±ðxƒ$□!WW/Í□¹ˆ□B«ñø
âG⁂u¨ªĀÊ%□ĔX:X□p□Rÿ□,□FĀß□°<}<\□{ŁpióV□a□ Zy□$□».¨e□□S□Āi²p}¬□gd%>@ˆøĐ%Am§=s□(᠑j□lŬNp Ekx
&pă□Ĕ¢⁂H□□psKKM |đ□âu%□%â□⁂İfĔ$□□□Pū
□□ð□HÆ□%Y2óú□$ă!.□□z□□,âp¨«a□Ŭ□□R□□£□!ŬFBŬ□ŭ□Ĕ8~D□□¬¨"□U´□□ŭ□İ8^□□%□PK□□
□□ A $Ā%¢¥□ ô □ < org/gradle/cli/CommandLineParser$MissingOptionArgState.class□mŎŎP□ÇyW
7ŎŬ□□□q□ă□eĐ□-İrD□.□□_ðk.]S⁂zü3□□§B□²js×5]□¹□#□Ā□äuŬ,nŬV±ê4\Ŏk□θ□¿Ĕ#□%□rx□□w□;□i□□[c□U
SĔx[t¿□)|<□L/gdēi□x÷□=Ij□$i□¨ ¨Ñē0iOH□Đ□*İö□É□~¨⁂Cj□İAæQðŬĀ {□□I□;³ÑĀ□gI□/),£ß9□□ĐĀ□;
□□ A T□K>ª□ Ä□ = org/gradle/cli/CommandLineParser$OptionAwareParserState.classUŬnŎ□=□x
□J| □□MY□*A.□/p□□³ç□gfı□¿%ÿ PĀ%8□ĔēĐ0*w×Tz□©Ĕc□□□ □□*□L+(`□cVAQĀu□□á4□Ŭ®,ŬdÈ□□Q/Ŏ□ˆ5İ□Ñ
0¹pİŬē□İ□CŎ{i¹¹□□ē□\ŭ □ēēĔ᠑%;□óĀđˆũ
CdðŎ0²İéøh5«|xÆ«ð□□□7*XŎð□□□.CúĔ᠑g9ŎĔē□□%ZŬă□ĀmuŬ6Āj□»I.□Ā,s%□è]SMb᠑᠑j□%‰}ĀP%ð□;æ□□.è□²',¨%4
```



BE□□\*^□Üvð#8«Ni\$èy6□φ□□+□ùÿÑ1E□83\$re]□φoPK□□  
□□ A □□àİ»□ a□ 7 org/gradle/cli/CommandLineParser\$OptionComparator.class□Tm0ÓP□~iÖÑQ:□  
%□Ý  
□2@□□□.Ñ,YÀd□Ä□e4³mkIU□□□□□?□/~□#□á³ÿÆ"□\_□ç¶N6).]İĖ}İsi=cð~ýðu□□E□%dθ#;□;úθ+bn□ UÂ□J"□E,IHs;□  
ùä□#fÂdyPä"□5□□□">-{ }FCĀâNS□Æ#ZOâq□İf□H□.INQİ2(RØ8éôĖ|ÉSĀ<Üi□Ė□ÿPK□□  
□□ A äfC□\$□ £□ 8 org/gradle/cli/CommandLineParser\$OptionParserState.class□□ßJ□A□Æ;³Y³e|  
}□"□/ú }"Ö3□□□□4»□ß□³ßü□İ□ýýćá□□#ÖB□°□ĀÇF□İ UBñXie?□  
ð%□Āo| }IX1)-O□□®4çφ□pf¹□öDð□F¹ó8éÜK□□□İ¹Jð□a2iUVXpTú®µ4İDd□dĖÇVjâ(6φ□Ė"□""□□□B÷]□QJí□mAXH5□  
□□ A φÆ÷E«□ ¶□ 3 org/gradle/cli/CommandLineParser\$OptionString.class□TÿN□A□pözâÊq□Z□DQ□  
□□ A gAq²□□ x□ = org/gradle/cli/CommandLineParser\$OptionStringComparator.class□TİO□A□pf  
U□□.ç°¥²□□;^`Ėý□□;zn7]□¶□□è□bó  
mÇü)Pz÷i□TJ`{-;amÛr`LW»yyĖ□□□□%,kxİ°Ü□cE□VÖ,7°CûĀĖ%□□=□²ăm□Qd□ò;ófXýİ□t¥XG,□□ēðb□ēýOð%q,·□□  
□.□¥h□"□,##□□% □G□Ç□I□ă  
ðă□□ n`□,□òĀ□?Cj<°U<□φ%□|%□a1ÿy□`ÙSèfē□□%□=□i)□jû□ô□i□□"j□m\*φD□Çé³□□Ā O"Ė5Ló>ă¹□□□Ā□OE□□  
□□ A `M~U□□ ,□ 2 org/gradle/cli/CommandLineParser\$ParserState.class□SBoðP□pN)□□s□ûÖ9□□  
□□ A pÍXŸ□ k□ ? org/gradle/cli/CommandLineParser\$UnknownOptionParserState.class%UĖR□Q□=w  
BY°»\$Sqt2CM&Ė/ø/.#□G□□?Ā□□\_7nÜè/XvO|B□P! .ÜtßÜ·İĖ%\$o&i□;□`  
·5Ā□Öð□±6Z□³Ė±□à□I□R|f,□Ā5□µiİİ□â□@û#é□w□□□ú□û6U%K□ĖFY`|ûq□¹φ+□□□Ė[fnĖ)□m]X6mf□□□®İ □m  
□□ A dZ□Ô□□ 1) & org/gradle/cli/CommandLineParser.class□Yi`Ā□pJ□□7zz¶eU²=`lá□ē□1¶°âS>  
zE\*ÔJ□\_aÖ.3Us°.³ă5□Ė□|~□□;□ĖÉ\5□\$¶-ðĖ|¥÷□;T;□i²XG3□Ė□□âHêx□Ė-f□ĖR; ,Sv-x±UV`ÑY~9[yç□]Ėē¥.çĖ\*  
Æ□□hL□u□ĖĖ□q³>?7#ñP"t□□m\$(°¹ôüİ□L8?²=□Y□±□×□cq3Ö□□LMÖsÇsO°¶A0Ůeú□AQC(□çÖøj]-ŸZç-um 3f  
Āâ□|imôkG0ðJ□Kİ¶ghİi□xdcĖ□u;□13cßM□ü□□İÜŸY[□1«x~\_Ÿ,µ±®AĀÖ5uēðXŮ(□-ē^□-īN□ĀÖ1³YŮU%&□ V□ü□QQ[Ů¥  
µG□□i□- \□¹°Āp□□FŮ«m□óWecİŌİ±`kð~n □a□ÜV□" |iüE□9:□□Gw6□v□□»ēÚİÖ|Æyqeİ4□□1ē×ðP<,-lò□%□;H  
kŮİ□°X5¹P°jðİĀ:ĀĖ□)B□p□(^□□H□i□□¹□`n Zİük□n»□□Y□□Ė\_□>N¶\$φ□  
□□□□İ□□&3A□32g□□R4iÑd«□Z□8?□lcy□ðO¶«□□m'□áĀ□f□çðh~Ō□%sÆ□UĖUĀİ6ē□□3□iĖ  
Ā□%¹Ůİ□φĀ%Āp77ð□□DLý^a÷E□□9İ□V\Ñi□□EW□p% vX8□)â~ă4iFf(Mâ%au□)i□à.ĀŌ1?°Fò` ,□N@6ßİ7üwâ□3ô³ăĀŚĖĖ  
□□ĀzφXç>`£Nd1iV□f□Y□72³pLbĖ h1s}Ā.%6=£Jq□®ĀĀ`İ\*£9j;¶ðDuXâ□U aēÇ»?□¹²}□eĖĖð³;□□\$[»□□Ö□eN(ăóq  
ç4□6°□ð□<[ ];k□t6SN¶%oİ□Æ;:ē□□□□`□»ð□ÜC□ēðQð□□| [ăç¥;`©ŮŊ|Y%óŮŮ(~°zvú□V□□7\ē□ □£□□"□zü□, a>²Ÿ  
9□Q□â} (à□Ā□ðôa`~.%°F□□;Ā□ZoeĀŮŮ±»â±□.°`□ăvĖ □ă□□ē□Æ}r□\$□%8i7N"eĀKâ□{1±Ō□ðôâ□□\N□I□AUĀ[°°□□  
ŌO~ð□ü%`ŮŮĀ□?i.UĀMj°aÑ[□w□\$Vý æð\*prv□×ē□"uG¶□□Yðxm□i;pâ>,|.□®³%□Ō□ðK□éŌ□İ1Ůw1fe/N`ðoE□□\$`Q  
İórĖögj;ðTp¶4□ð□3□+®"d□□; (ân□□W16□±E□□¶Jk□Ů~ðæ f5□7V□°ñăðx□%E□□□ĖEPYµ%□dN□□□>İİĀ□k%□ûßUðâ□ :  
Ÿñ□|□Q³û□c"□□á□û%¹İp;Ā°üa□s□Ė□`□q□×ç;□ð¹à □6V|□¹□□ē□~,|i□Ōz\*-wŌ□ü□□%°F□G|¶□(:%f)ĀX□J`X□□²  
Ōd^pòK¶Æİ□□ñ7x□Y□□\$ñ+òçb□ŮŮ7X:y`ñôx□Yçµ²Fâ□ĖÉ`S□ü-1fPáİ8GĖē0Ėð{zSex□y□µNv%¥a□` ;aZ□x%□úâ□5Ōð  
□eİc%,3w□□Çă}}□s□Ůēē²S□}ÆbÇ~e?□□2□□<□Ė\_□`pâa²>1J`~ðŸY□□j%\$iA^□Āiô□  
u□oQEŮĖŮ□;Āð¹p Āa□8D~?Pâ□oSÑw□/;ă-yŌĀ³w□g;Ā□b¶Kñ/`Ÿy□□U°□üGðy□^¶}×B±□□«□,İZJ%÷,;□□ó□ô;Ėφ□□%  
cÑĀY\*^®□°ðHY\*□R·Ė!çV¹□g□ð~-y□hÇà·.□#`f□N>□@\_Ā{□EV43\$§□ðñİð□çf□f`□N¹8÷Ñ~·Ā□` ;ŮĖ1çİdp0ü□□sNöĀĀŮŮ  
;{±□□□g□<¹Oajað v<□sð[©□Gò,%.CX"□üð`ù□Q□kG□Oa□XÇg\□□p□³□±□İ>İ>Ā%□â.a;□□»ð÷xtòvâ2İ□g□9ü+ð□□  
□□ A □□â>Ė□ □ & org/gradle/cli/ParsedCommandLine.class□Wiw□g□~FŮĖðĀ□ē%v□Tmc[□â□□ ñR·  
□İ%□1s□#+3¶□Ė¥©□% /ē□E;□□□ŌóĀ  
ēf2é□n□-CĀĖĖ\$#İ2«çð□□Āð¶i¥□Ō□□Ė□□ŮİĀ□°U0RăæŌ□□KMfrÆ)yâð(□□ð□Q0r¶Ė\*T□>c04y¶Zăq □²□%sŮH□□□□  
`%ăİ FA|»□ŌWŸY³wó□/ē□%□)xðŌàŮV5KŌ¶x± :Ō□|)ŮĀ□ð#÷wOP;çĖŌ□□c□I(□ßVñ□□Ā^yð□φĖUð6K¥p□ið¶ÿbÑ`□  
`Rİ□|{□çY□#ŮŸðô-nQ/□a~5^kÑK□□u·□O;=F□,m,·RŮŮŮ`h¶q3□5□\$ßbwcŌLO□ĖF¶ŮŮm¶Æ  
□□□¶%ßŮLİðİñ`a□Yðûâ9mðG7M□h0z\*gJw□r□ vè©□D□Ō³Ec□úY°Āeçy□İK□?ēa{ÿðÿ?ĀŌ□qZ:~«f□y Ó×Ė□Y`□□c□Ñ  
Ÿ é,^Xd□kēçp□□□ü□p□□□□á□□□6□□Y□%`□Ōð□|PĀ|x[Āwñ=ßÇ□□Āiç1sµ=>4ü□?Rñç¶ĀfŮUüDĀŌñ3□□Ÿñ8  
|Ė□□\_à□□□□=□|θaç4ü  
WU\Ō0□³□~□ßp`WL□ôHŽİ□%:zyÁpyðēð□ēē\*ðĀr|□K□ç□}j;Ė□cŮ□ē, aá□□Ñð□®Ō□[Ā□»Ėv.□□2}Ri□□Æ□□c9SV`UðP  
ŮŮ`ðQ¥`Z³□y□□Ů□ămàµ□O□BçMkI'Æ GĀ>yP □İ□=□7r□□W=Ā¶~©□□İ□gDð□1□□[%f(□O□:°%6y□ŮP=ð,□³J8□J□e³  
Zç3%□]□^á□□□φkéó>□□1□Jp;İ k>ŮŮ□eP`\*ù6zp□□Eă□□Ge¶)ð¶Aù<□],;[İφĀS□; \ð-O°□□r□=18>Y □Āß□ă□□□^Ů  
□dĖ¹ÿ□PK□□  
□□ A yß□tŮ□ E□ , org/gradle/cli/ParsedCommandLineOption.class□S]ŌŌ@□=Ÿ`İG□eùP□a□□eQa`φ  
□,□|â□□Ė\$ð4□,□Ā-L%q[%İ` ,+ßŌr¹\$â%□□  
□[İlpWĀ¶`Ė¶pðð©□□ēİ(H□□a1`áp□#~□g□³Ė,C/{□°□□9\*İİ  
Kxs  
φ□ñU□±E»JĀ□%añ□□Ů:w^³u□"¹□]aæ\*s□<□Ā□·!□Ijd;□n8~jr%b  
ý□s^`]`k5fU%ŌJŸ□¶Ėİ□□÷V¥ □Ÿ□ñ³□□`□=Vy÷□ŌĀ□©Ė  
j□;\*i□=z&x□Ō«V□□£Qă□\$ šp6K:□dç)¹□ñKn07T@□%Ů□.Ů\$Ā □ Ėvâ□ē□□IC□z□t□°]p□¶-)kèDNĀ(Æ□□t]6Mn0□,İ  
÷±( # İWmİæ~ŮĖ°~□gŌvb;□□²Ñ°□□□f0« ?L`æuZĀðDŸĀA|K□@Ōİ¶\*Ōð□ó□□\$;Ñÿ□İ)□□ðM`!ŮŮ□□İ□+v□İ□i¥áĖQ  
×\$ĖĀŮŮ"xŌjĖŮ2D~İ\$İç□xr7□k>°%|ăó□V.pĖ]Ā.ŌðG²ăó±ðăD□ç□Ō8"Ůy Rð□□ÿ□PK□□  
□□ A \vÆB|□ □□ : org/gradle/cli/ProjectPropertiesCommandLineConverter.class□□KŌĀ@□□İ□□  
}FHWkw□LS°□ŮŮ!;□□]°nY×7İZK:İ;cJDðŌİē□□□ZRY¶\$□ēsŮñ□ŸVð□;ŮH□+-φ )□□□n`kS□#cruLXðøgh|Ō×B□□□j  
ßµ|XUPðŮV%□□ĖŮİ0□1□ĖkĀĀ□□□p8□□w□Z±ß8T0gİ?□□Pôa|İ□□m□□İi=□□İC  
□S³□s□□|□\$□£□□| Ė1\□áoZēq~{-CŌ èJ□□□EĖē□ēj□□□E+ ``Ůw'Ōð□PK□□

```

    A
İ8=| ü 9 org/gradle/cli/SystemPropertiesCommandLineConverter.class
o}É,G ÑÖûTFMz99y²~X{()spl`7e.°KV,øTXxÉfö¿FD T E xG ÄPCW□āJm@h~²Æ49A1jxuÑ°ÓsÄhöİ
Öi$P|‰ô´6¿;□/t-İ,¢;@h-İ.¢Z
a>kizY□PK□□
    A ÄW□ □ gradle-cli-classpath.properties+(ÊİJM.)ª**İ+ÊİMµ PK□□□□
    A □ □ İA META-INF/PK□□□□
    A m±>=@ ? □ x) META-INF/MANIFEST.MFPK□□□□
    A □ □ İA org/PK□□□□
    A □ □ İA; org/gradle/PK□□□□
    A □ □ İAè org/gradle/wrapper/PK□□□□
    A ãÛfØ□ □ è - x□□ org/gradle/wrapper/BootstrapMainStarter.classPK□□□□
    A hQb}¢ Ø # xk□ org/gradle/wrapper/Download$1.classPK□□□□
    A Çâi@□ □ L 4 xN□ org/gradle/wrapper/Download$ProxyAuthenticator.classPK□□
    A 5¶<□À □ □ ! x» org/gradle/wrapper/Download.classPK□□□□
    A y□L¢; Ê 1 xº□ org/gradle/wrapper/DownloadProgressListener.classPK□□□□
    A !9|□¶ □ □ 3 xª□ org/gradle/wrapper/ExclusiveFileAccessManager.classPK□□□□
    A □i□,y□ □ □ - x±□ org/gradle/wrapper/GradleUserHomeLookup.classPK□□□□
    A À!lá7 "□ * xu# org/gradle/wrapper/GradleWrapperMain.classPK□□□□
    A úâ□æª Û " xô, org/gradle/wrapper/IDownload.classPK□□□□
    A p@áôc□ □ " xþ- org/gradle/wrapper/Install$1.classPK□□□□
    A T",□-□ t+ x5□ org/gradle/wrapper/Install.classPK□□□□
    A ð¿:ôo□ 4□ □ xiI org/gradle/wrapper/Logger.classPK□□□□
    A on□□¶ □ □ 8 x□L org/gradle/wrapper/PathAssembler$LocalDistribution.class
    A □JØâ□ □ □ & xN□ org/gradle/wrapper/PathAssembler.classPK□□□□
    A □□íêÄ □ | 0 xpU org/gradle/wrapper/SystemPropertiesHandler.classPK□□□□
    A ã=0□° □ ?□ - x□[ org/gradle/wrapper/WrapperConfiguration.classPK□□□□
    A Gü"□ □ □ ( x^□ org/gradle/wrapper/WrapperExecutor.classPK□□□□
    A ã□Ð_ □ □ # xdg gradle-wrapper-classpath.propertiesPK□□□□
    A □ □ □ İAÄg org/gradle/cli/PK□□□□
    A ÖÛ?®<□ S□ 1 xðg org/gradle/cli/AbstractCommandLineConverter.classPK□□□□
    A x□µ³X□ ì
; xj org/gradle/cli/AbstractPropertiesCommandLineConverter.classPK□□□□
    A }İyG□ K□ 1 x.o org/gradle/cli/CommandLineArgumentException.classPK□□□□
    A ³ßâú□ □ g□ ) xÄp org/gradle/cli/CommandLineConverter.classPK□□□□
    A Sf
Ö□□ g□ & x$r org/gradle/cli/CommandLineOption.classPK□□□□
    A üi□`¥ ä ( xjx org/gradle/cli/CommandLineParser$1.classPK□□□□
    A □ÐÆN□ ä ; xuy org/gradle/cli/CommandLineParser$AfterFirstSubCommand.
    A ¥□D£¢ □ &□ 3 xü| org/gradle/cli/CommandLineParser$AfterOptions.classPK□□□□
    A xÇÇà□ □ □ < xi□ org/gradle/cli/CommandLineParser$BeforeFirstSubCommand.c
    A ´*«ZM □ i□ F xJ□ org/gradle/cli/CommandLineParser$CaseInsensitiveStringCom
    A □è2 □ □ Á□ = xû□ org/gradle/cli/CommandLineParser$KnownOptionParserState
    A $Ä%¢¥ □ ô □ < xZ□ org/gradle/cli/CommandLineParser$MissingOptionArgState.cl
    A T□K>ª □ Ä□ = xY□ org/gradle/cli/CommandLineParser$OptionAwareParserState.c
    A □%àİ» □ ¢□ 7 x^□ org/gradle/cli/CommandLineParser$OptionComparator.classPK
    A äfC□$ □ £□ 8 xn□ org/gradle/cli/CommandLineParser$OptionParserState.classP
    A ¢Æ÷E« □ ¶ □ 3 xk□ org/gradle/cli/CommandLineParser$OptionString.classPK□□□□
    A gAq² □ □ x□ = xg□ org/gradle/cli/CommandLineParser$OptionStringComparator.c
    A `M~U□ □ □ 2 xV□ org/gradle/cli/CommandLineParser$ParserState.classPK□□□□
    A pÍXY□ k□ ? xa; org/gradle/cli/CommandLineParser$UnknownOptionParserState.c
    A dZ□Ô□ □ l) & xäx org/gradle/cli/CommandLineParser.classPK□□□□
    A □□â>É□ □ □ & x>· org/gradle/cli/ParsedCommandLine.classPK□□□□
    A yß□tÚ □ E □ , xK¿ org/gradle/cli/ParsedCommandLineOption.classPK□□□□
    A \vÆB| □ □ □ : xoÄ org/gradle/cli/ProjectPropertiesCommandLineConverter.clas
    A
İ8=| ü 9 xCÄ org/gradle/cli/SystemPropertiesCommandLineConverter.classPK□□□□
    A ÄW□ □ □ xÄ gradle-cli-classpath.propertiesPK□□ 1 1 □□ gÆ

```

Let's revisit the two tests we just wrote, to reflect on the efforts and the benefits.

The nice part of the tests is that by mocking the functions `withContext()` and `async()`, we were able to perform interaction tests and didn't have to worry about actually running asynchronous operations and waiting for results in the tests. This makes the tests fast and deterministic. The biggest downside to this approach is figuring out which class to mock in order to reach in and replace the functions like `withContext()`. To say the least, this isn't easy, and it can be frustrating to find the correct class to mock. But the fast feedback we get from the tests is worth the effort.

---

In the next lesson, we'll try to use a service with our application.