

Añadir dependencias en bibliotecas multiplataforma

Cada programa requiere un conjunto de bibliotecas para funcionar con éxito. Un proyecto multiplataforma de Kotlin puede depender de bibliotecas multiplataforma que funcionen para todas las plataformas de destino, bibliotecas específicas de la plataforma y otros proyectos multiplataforma.

Para añadir una dependencia de una biblioteca, actualice su archivo `build.gradle(.kts)` en el directorio de su proyecto que contiene código compartido. Establezca una dependencia del tipo requerido (por ejemplo, implementación) en el bloque de dependencias:

Kotlin

Genial

Kotlin:

```
kotlin {
    sourceSets {
        commonMain.dependencies {
            implementation("com.example:my-library:1.0") // library
        }
    }
}
```

Groovy:

```
kotlin {
    sourceSets {
        commonMain.dependencies {
            implementation("com.example:my-library:1.0") // library
        }
    }
}
```

Dependencia de una biblioteca Kotlin

Biblioteca estándar

Se añade automáticamente una dependencia de una biblioteca estándar (`stdlib`) en cada conjunto de fuentes. La versión de la biblioteca estándar es la misma que la versión del complemento multiplataforma kotlin.

Para los conjuntos de fuentes específicas de la plataforma, se utiliza la variante correspondiente específica de la plataforma de la biblioteca, mientras que se añade una biblioteca estándar común al resto. El complemento Kotlin Gradle seleccionará la biblioteca estándar JVM adecuada en función de la opción de compilador `compilerOptions.jvmTarget` de su script de compilación de Gradle.

Aprende a cambiar el comportamiento predeterminado.

Bibliotecas de pruebas

Para las pruebas multiplataforma, la API de `kotlin.test` está disponible. Cuando crea un proyecto multiplataforma, puede agregar dependencias de prueba a todos los conjuntos de origen utilizando una sola dependencia en `commonTest`:

Kotlin:

```
kotlin {
    sourceSets {
        commonTest.dependencies {
            implementation(kotlin("test")) // Brings all the platform
dependencies automatically
        }
    }
}
```

Groovy:

```
kotlin {
    sourceSets {
        commonTest {
            dependencies {
                implementation kotlin("test") // Brings all the
platform dependencies automatically
            }
        }
    }
}
```

Si utiliza una biblioteca kotlinx y necesita una dependencia específica de la plataforma, puede utilizar variantes específicas de la plataforma de bibliotecas con sufijos como -jvm o -js, por ejemplo, kotlinx-coroutines-core-jvm.

Kotlin:

```
kotlin {
    sourceSets {
        jvmMain.dependencies {
            implementation("org.jetbrains.kotlinx:kotlinx-coroutines-
core-jvm:1.7.3")
        }
    }
}
```

Groovy:

```
kotlin {
    sourceSets {
        jvmMain {
            dependencies {
                implementation 'org.jetbrains.kotlinx:kotlinx-
coroutines-core-jvm:1.7.3'
            }
        }
    }
}
```

Dependencia de las bibliotecas multiplataforma de Kotlin

Puede agregar dependencias a las bibliotecas que han adoptado la tecnología multiplataforma Kotlin, como SQLDelight. Los autores de estas bibliotecas suelen proporcionar guías para añadir sus dependencias a su proyecto.

Echa un vistazo a esta lista de bibliotecas multiplataforma Kotlin mantenida por la comunidad.

Biblioteca compartida para todos los conjuntos de fuentes

Si desea utilizar una biblioteca de todos los conjuntos de fuentes, puede agregarla solo al conjunto de fuentes común. El complemento móvil multiplataforma de Kotlin añadirá automáticamente las partes correspondientes a cualquier otro conjunto de fuentes.

No se pueden establecer dependencias en las bibliotecas específicas de la plataforma en el conjunto de fuentes común.

Kotlin:

```
kotlin {
    sourceSets {
        commonMain.dependencies {
            implementation("io.ktor:ktor-client-core:2.3.5")
        }
        androidMain.dependencies {
            // dependency to a platform part of ktor-client will be
            added automatically
        }
    }
}
```

Groovy:

```
kotlin {
    sourceSets {
        commonMain {
            dependencies {
                implementation 'io.ktor:ktor-client-core:2.3.5'
            }
        }
        androidMain {
            dependencies {
                // dependency to platform part of ktor-client will be
                added automatically
            }
        }
    }
}
```

Biblioteca utilizada en conjuntos de fuentes específicos

Si desea utilizar una biblioteca multiplataforma solo para conjuntos de fuentes específicos, puede agregarla exclusivamente a ellos. Las declaraciones de biblioteca especificadas solo estarán disponibles en esos conjuntos de fuentes.

Utilice un nombre de biblioteca común en tales casos, no uno específico de la plataforma. Al igual que con SQLDelight en el siguiente ejemplo, use `native-driver`, no `native-driver-iosx64`. Encuentra el nombre exacto en la documentación de la biblioteca.

Kotlin:

```
kotlin {
    sourceSets {
        commonMain.dependencies {
            implementation(project(":some-other-multiplatform-
module"))
        }
        androidMain.dependencies {
            // platform part of :some-other-multiplatform-module will
            be added automatically
        }
    }
}
```

Groovy:

```
kotlin {
    sourceSets {
        commonMain {
            dependencies {
                implementation project(':some-other-multiplatform-
module')
            }
        }
        androidMain {
            dependencies {
                // platform part of :some-other-multiplatform-module
                will be added automatically
            }
        }
    }
}
```