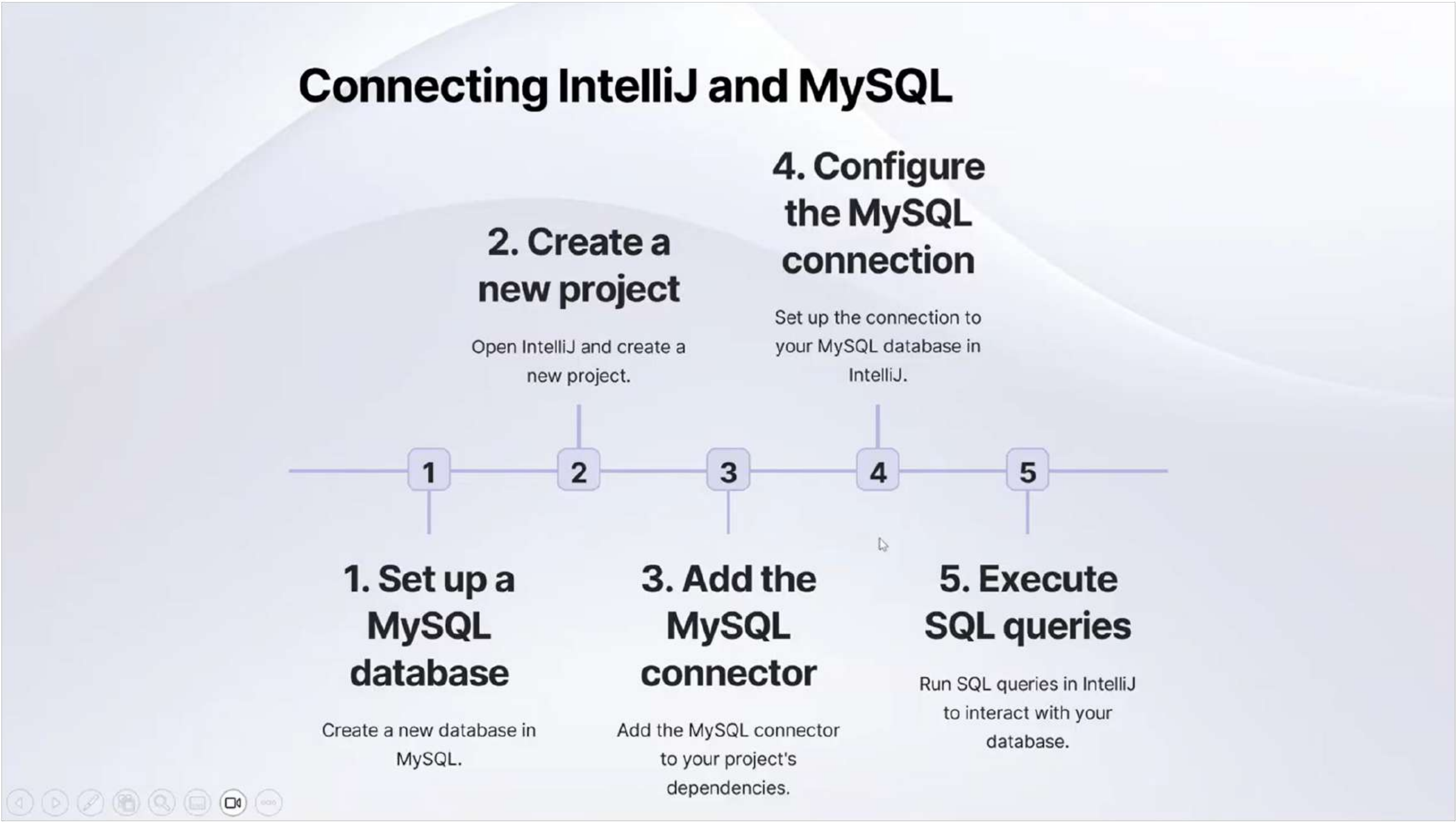
How to Connect IntelliJ with MySQL

IntelliJ IDEA is an integrated development environment that supports MySQL databases. Learn how to connect them today!

gradle-intellij-plugin – IntelliJPlugin.kt [gradle-intellij-plugin.main]
Plugin
apply
RunPluginVerifierTaskSpec.do not use Li...pty array passed to ideVersion
RunPluginVerifierTaskSpec.kt
RunPluginVerifierTask.kt
IntelliJPlugin.kt
build.gradle.kts (gradle-intellij-
IntelliJPlugin
apply()

```
package org.jetbrains.intellij

import ...

Jakub Chrzanowski +5
@Suppress( ...names: "UnstableApiUsage", "unused")
open class IntelliJPlugin : Plugin<Project> {

    private lateinit var archiveUtils: ArchiveUtils
    private lateinit var dependenciesDownloader: DependenciesDownloader
    private lateinit var context: String

    Jakub Chrzanowski +1
    override fun apply(project: Project) {
        archiveUtils = project.objects.newInstance(ArchiveUtils::class.java)
        dependenciesDownloader = project.objects.newInstance(DependenciesDown
        context = project.logCategory()

        checkGradleVersion(project)
        project.plugins.apply(JavaPlugin::class.java)

        val intellijExtension : IntelliJPluginExtension! = project.extensions.create
            IntelliJPluginConstants.EXTENSION_NAME,
            IntelliJPluginExtension::class.java,
        )

        intellijExtension.apply {  this: IntelliJPluginExtension!
            pluginName.convention(project.provider {
                project.name
            })
            updateSinceUntilBuild.convention( value: true)
            sameSinceUntilBuild.convention( value: false)
            instrumentCode.convention( value: true)
            sandboxDir.convention(project.provider {
                File(project.buildDir, IntelliJPluginConstants.DEFAULT_SANDBO
            })
            intellijRepository.convention(IntelliJPluginConstants.DEFAULT_INT
            downloadSources.convention(!System.getenv().containsKey("CI"))
            configureDefaultDependencies.convention( value: true)
```



Connecting IntelliJ and MySQL

1. Set up a MySQL database

Create a new database in MySQL.

2. Create a new project

Open IntelliJ and create a new project.

3. Add the MySQL connector

Add the MySQL connector to your project's dependencies.

4. Configure the MySQL connection

Set up the connection to your MySQL database in IntelliJ.

5. Execute SQL queries

Run SQL queries in IntelliJ to interact with your database.