

**KOTLIN /
Everywhere**
Turkey KUG

WELCOME

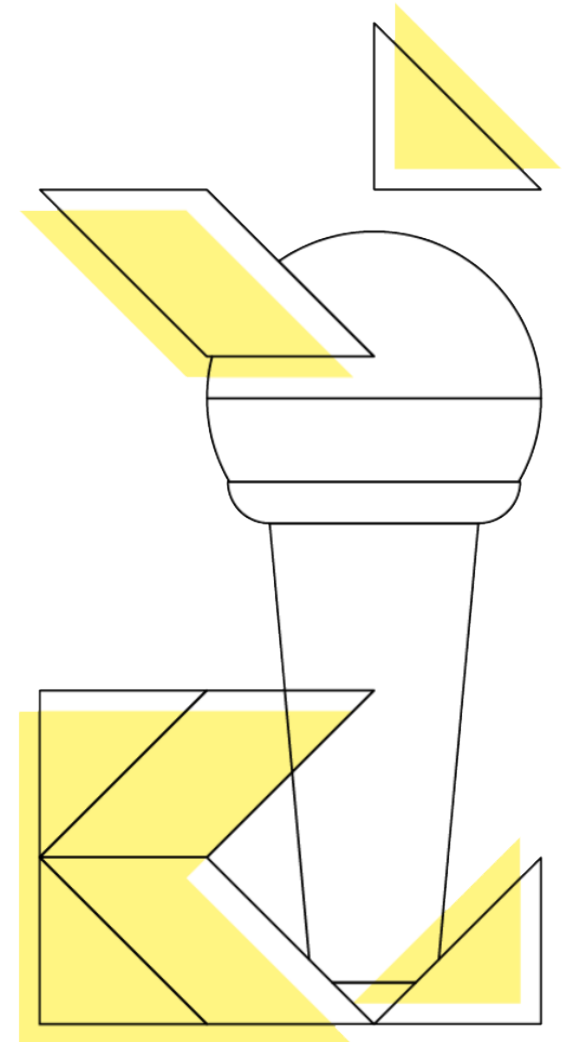


How to setup a multi-platform project

SPEAKER NAME

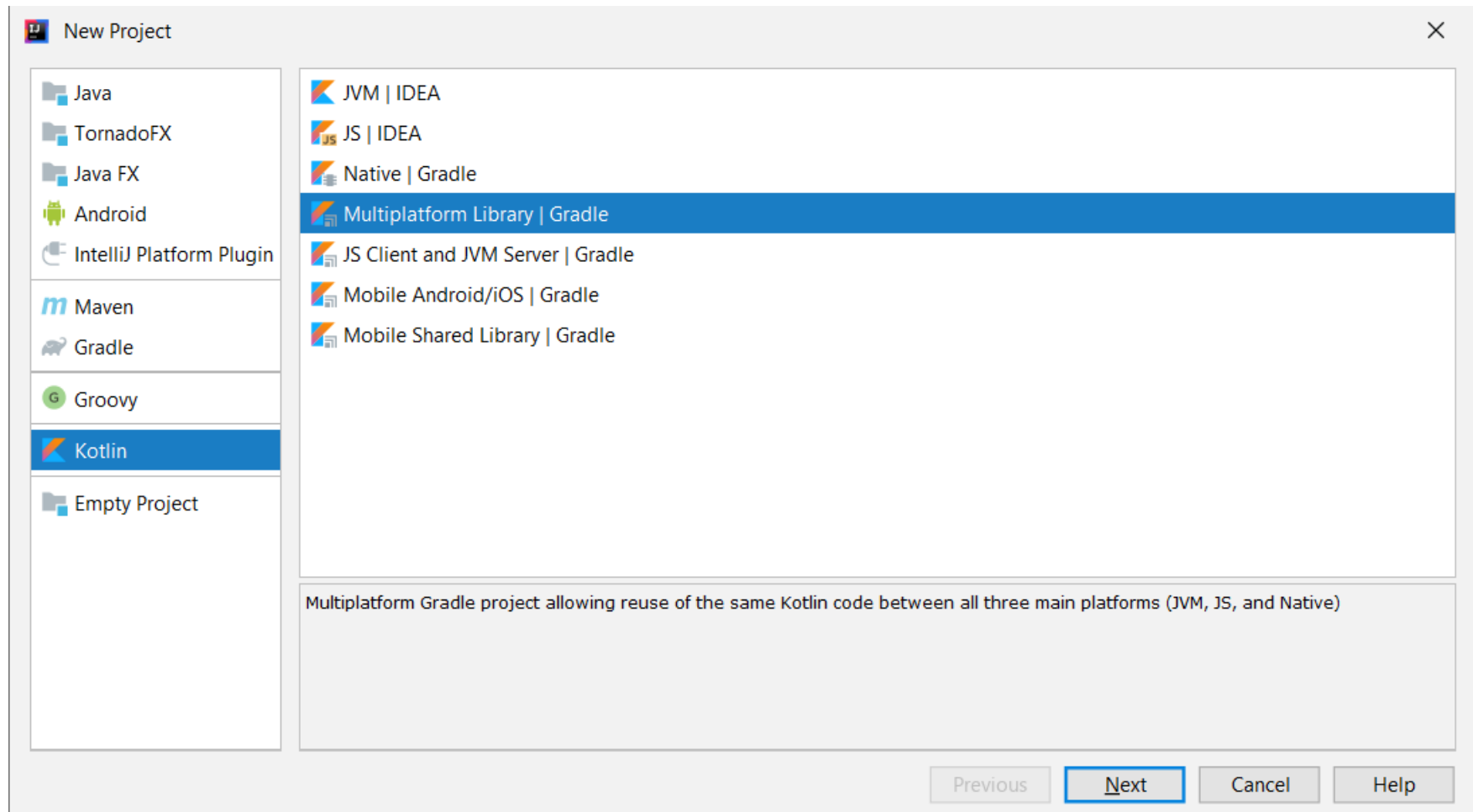
Mehmet Ali SICAK

CS Teacher, MEB

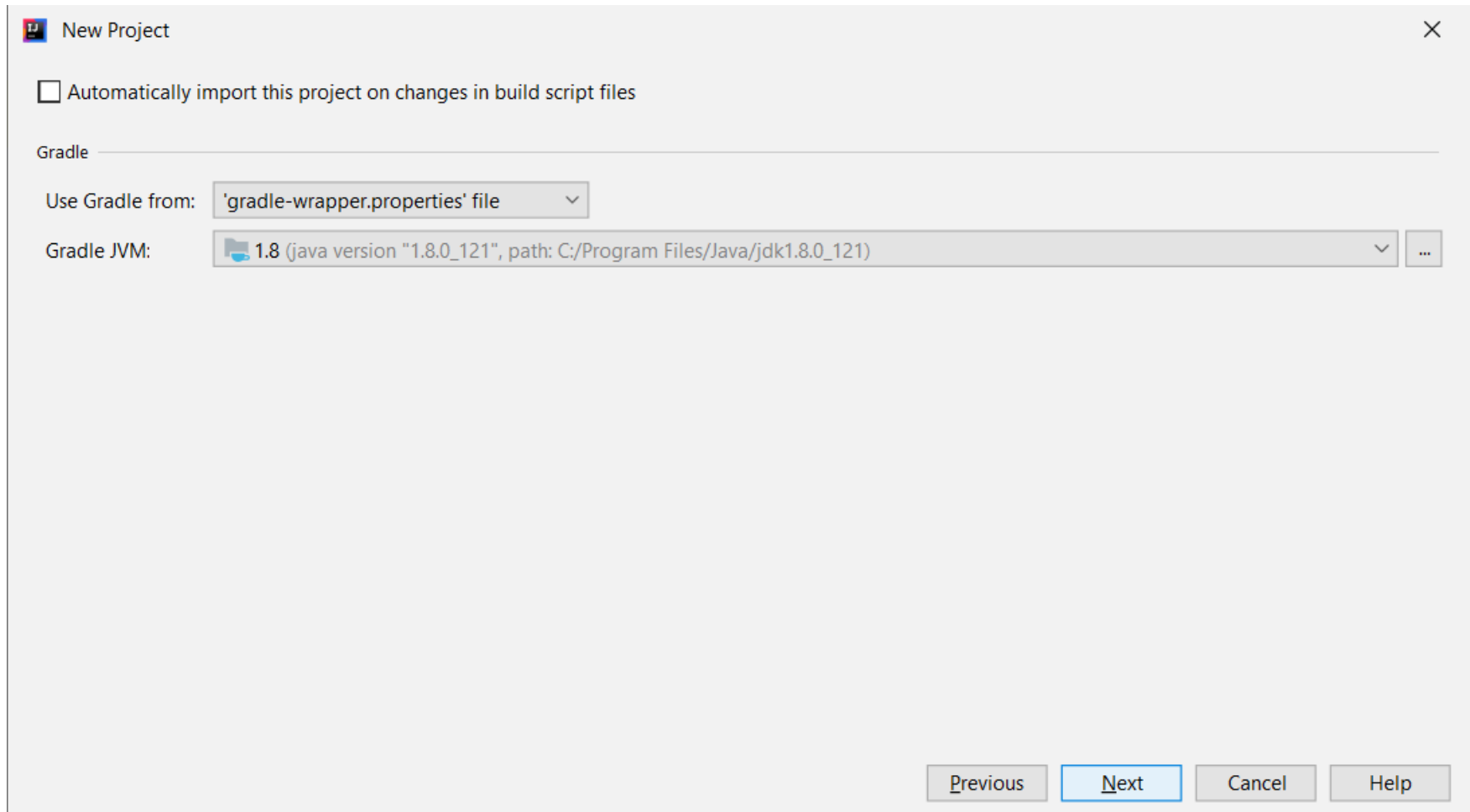


Your first multi-platform with Kotlin (Setting up a project)

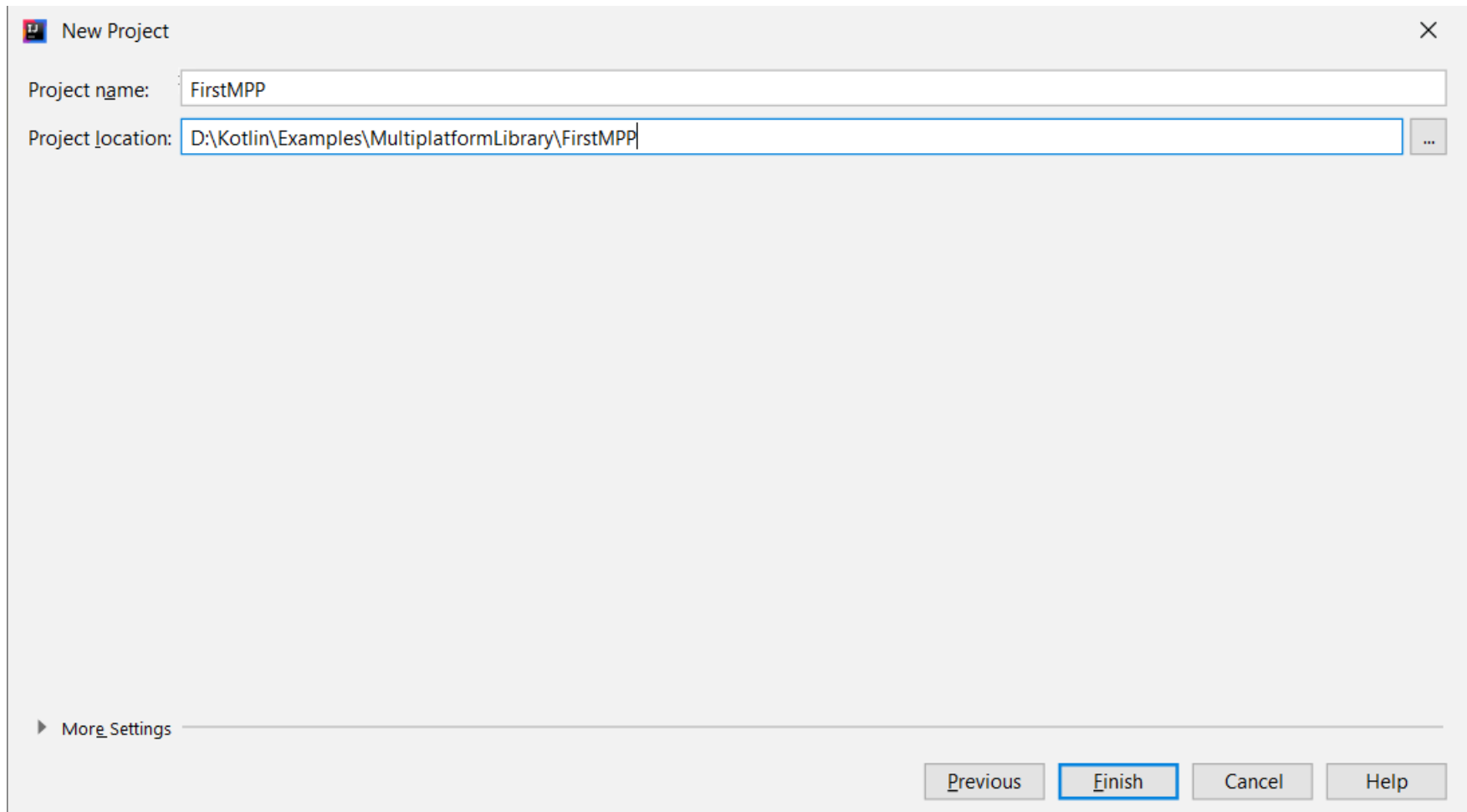
We select File | New | Project, select Kotlin | Kotlin (Multiplatform Library)



Your first multi-platform with Kotlin (Setting up a project)



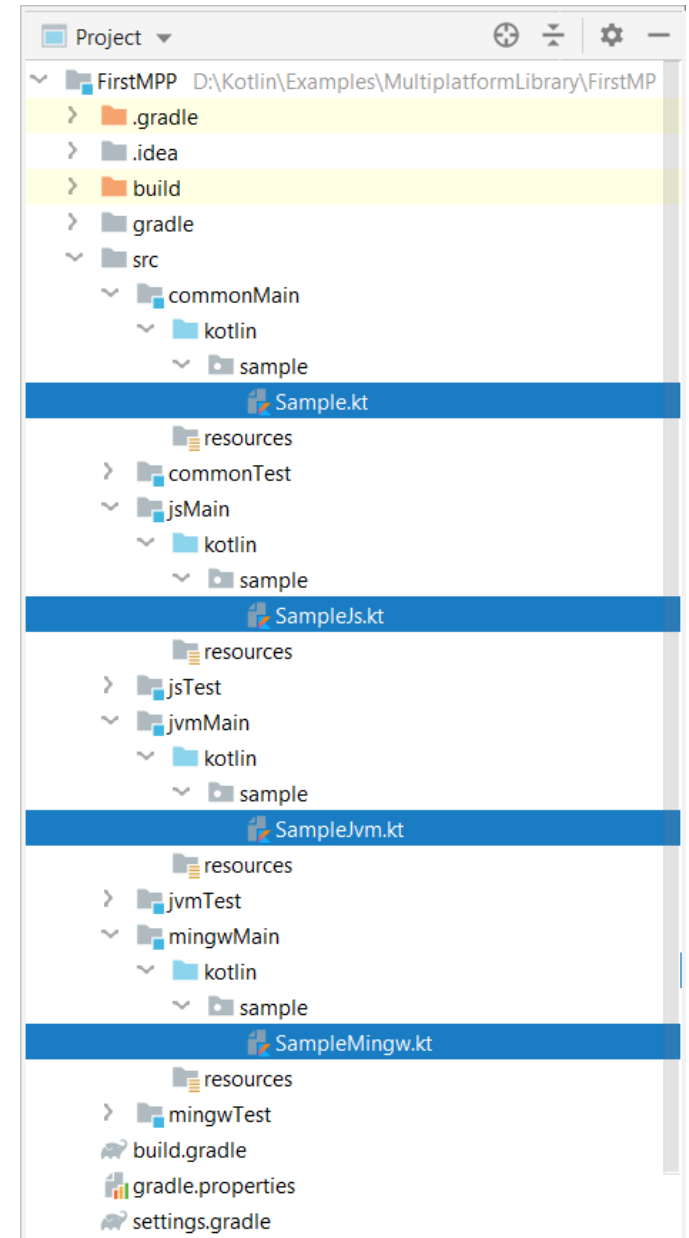
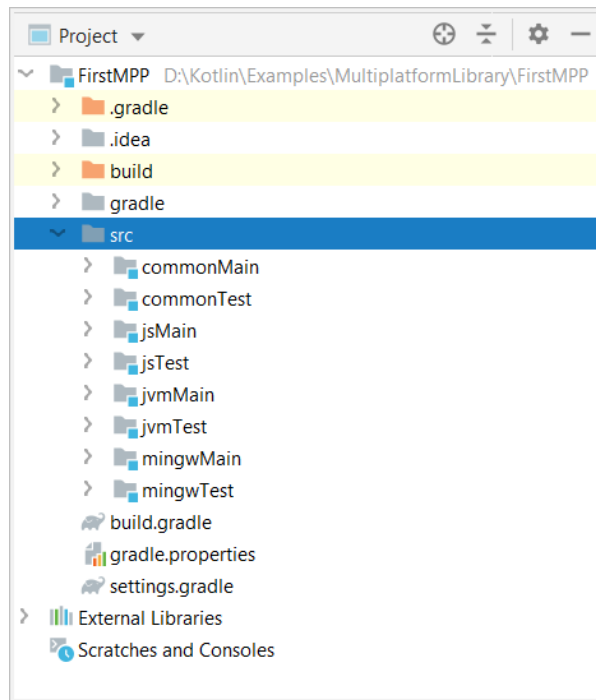
Your first multi-platform with Kotlin (Setting up a project)



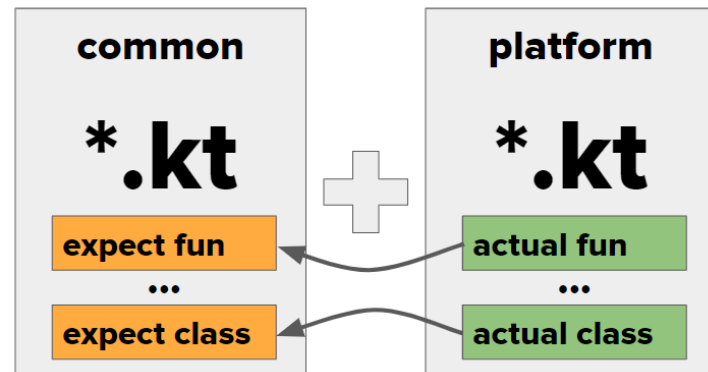
The screenshot shows a 'New Project' dialog box with the following fields and buttons:

- Project name:** FirstMPP
- Project location:** D:\Kotlin\Examples\MultiplatformLibrary\FirstMPP
- Buttons:** Previous, Finish (highlighted), Cancel, Help
- More Settings:** A link to expand more options.

Project Structure



expect & actual



Sample.kt

```
package sample
```

```
expect class Sample() {  
    fun checkMe(): Int  
}
```

```
expect object Platform {  
    val name: String  
}
```

```
fun hello(): String = "Hello from ${Platform.name}" +  
    " check is ${Sample().checkMe()}"
```

SampleJvm.kt

```
package sample
```

```
actual class Sample {  
    actual fun checkMe() = 42  
}
```

```
actual object Platform {  
    actual val name: String = "JVM"  
}
```

```
fun main(args: Array<String>) {  
    println(hello())  
}
```

expect & actual

Sample.kt

```
package sample

expect class Sample() {
    fun myAge(): Int
}

expect object Platform {
    val name: String
}

fun hello(): String = "Hello from ${Platform.name}" +
    " age is ${Sample().myAge()}"
```



SampleJs.kt

```
package sample

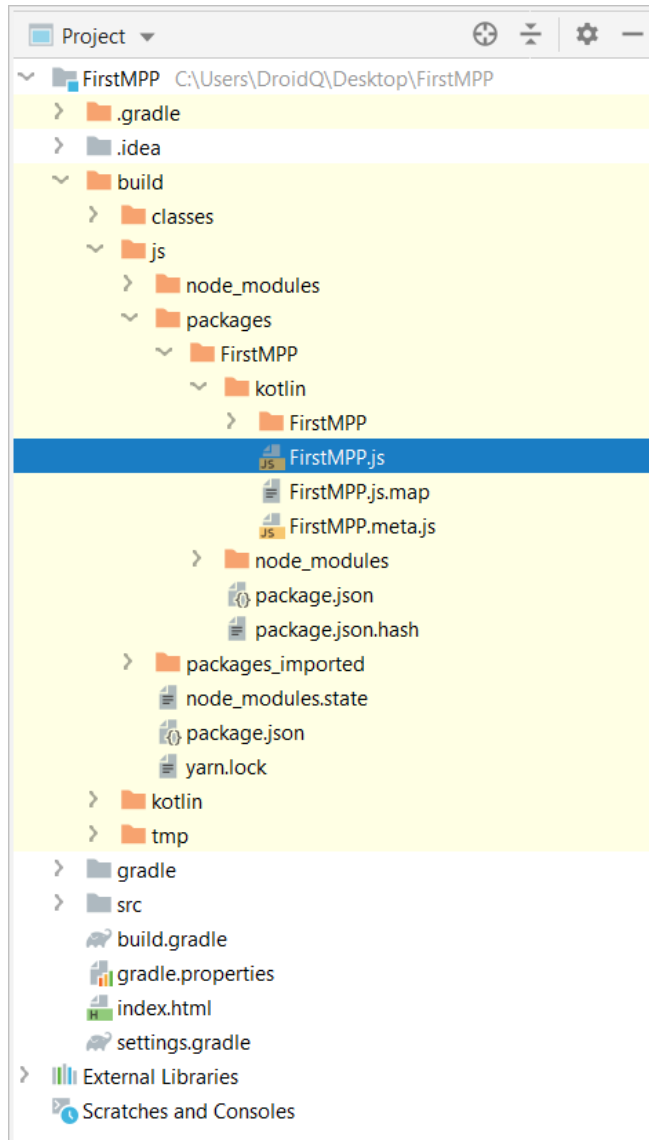
actual class Sample {
    actual fun myAge() = 24
}

actual object Platform {
    actual val name: String = "JS"
}

fun main(args: Array<String>) {
    println(hello())
}
```


JS

FirstMPP.js



```
function hello() {  
    return 'Hello from JS' + (' check is ' + (new Sample()).checkMe());  
}  
function Sample() {  
}  
Sample.prototype.checkMe = function () {  
    return 24;  
};  
Sample.$metadata$ = {  
    kind: Kind_CLASS,  
    simpleName: 'Sample',  
    interfaces: []  
};  
function Platform() {  
    Platform_instance = this;  
    this.name = 'JS';  
}
```

HAPPY HOUR

THANK YOU

