

Corrigés TP - Formation Mobile Débutant (Enrichie)

Séances 1 à 7 - Codes complets Kotlin & XML

Séance 1 - TP Personnalisation Hello World

XML :

```
android:id="@+id/helloText"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Bienvenue Ria"
android:textSize="24sp"
android:textColor="@android:color/holo_blue_dark"/>
```

Kotlin :

```
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
    }
}
```

Séance 2 - TP Factorielle

Kotlin :

```
fun fact(n: Int): Int {  
    var result = 1  
    for (i in 1..n) result *= i  
    return result  
}
```

```
fun main() {  
    println("Entrer un nombre :")  
    val n = readLine()!!.toInt()  
    println("Factorielle = ${fact(n)}")  
}
```

Séance 3 - TP Carte de Visite

XML :

Kotlin :

```
btnShow.setOnClickListener {  
    val name = inputName.text.toString()  
    val job = inputJob.text.toString()  
    output.text = "Je suis $name, $job"  
}
```

Séance 4 - TP Quiz 2 écrans

MainActivity.kt :

```
val intent = Intent(this, ResultActivity::class.java)
intent.putExtra("reponse", userAnswer)
startActivity(intent)
```

ResultActivity.kt :

```
val answer = intent.getStringExtra("reponse")
textView.text = "Votre réponse : $answer"
```

Séance 5 - TP ToDo List

Kotlin :

```
val tasks = mutableListOf()
```

```
fun addTask(task: String) {  
    tasks.add(task)  
    saveTasks()  
}
```

```
fun saveTasks() {  
    val prefs = getSharedPreferences("MyPrefs", MODE_PRIVATE)  
    prefs.edit().putStringSet("tasks", tasks.toSet()).apply()  
}
```

```
fun loadTasks() {  
    val prefs = getSharedPreferences("MyPrefs", MODE_PRIVATE)  
    tasks.clear()  
    tasks.addAll(prefs.getStringSet("tasks", setOf())!!)  
}
```

Séance 6 - TP App Photo

Manifest :

Kotlin :

```
val intent = Intent(MediaStore.ACTION_IMAGE_CAPTURE)
startActivityForResult(intent, 100)
```

```
override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {
    if (requestCode == 100 && resultCode == RESULT_OK) {
        val image = data?.extras?.get("data") as Bitmap
        imageView.setImageBitmap(image)
    }
}
```

Séance 7 - TP App Météo (API)

Retrofit Service :

```
interface WeatherService {  
    @GET("weather")  
    fun getWeather(@Query("q") city: String, @Query("appid") key: String): Call  
}
```

Kotlin :

```
val retrofit = Retrofit.Builder()  
    .baseUrl("https://api.openweathermap.org/data/2.5/")  
    .addConverterFactory(GsonConverterFactory.create())  
    .build()  
  
val service = retrofit.create(WeatherService::class.java)  
service.getWeather("Paris", "YOUR_API_KEY").enqueue(object: Callback {  
    override fun onResponse(call: Call, response: Response) {  
        val temp = response.body()?.main?.temp  
        textView.text = "Température : $temp °C"  
    }  
    override fun onFailure(call: Call, t: Throwable) {  
        textView.text = "Erreur : ${t.message}"  
    }  
})
```