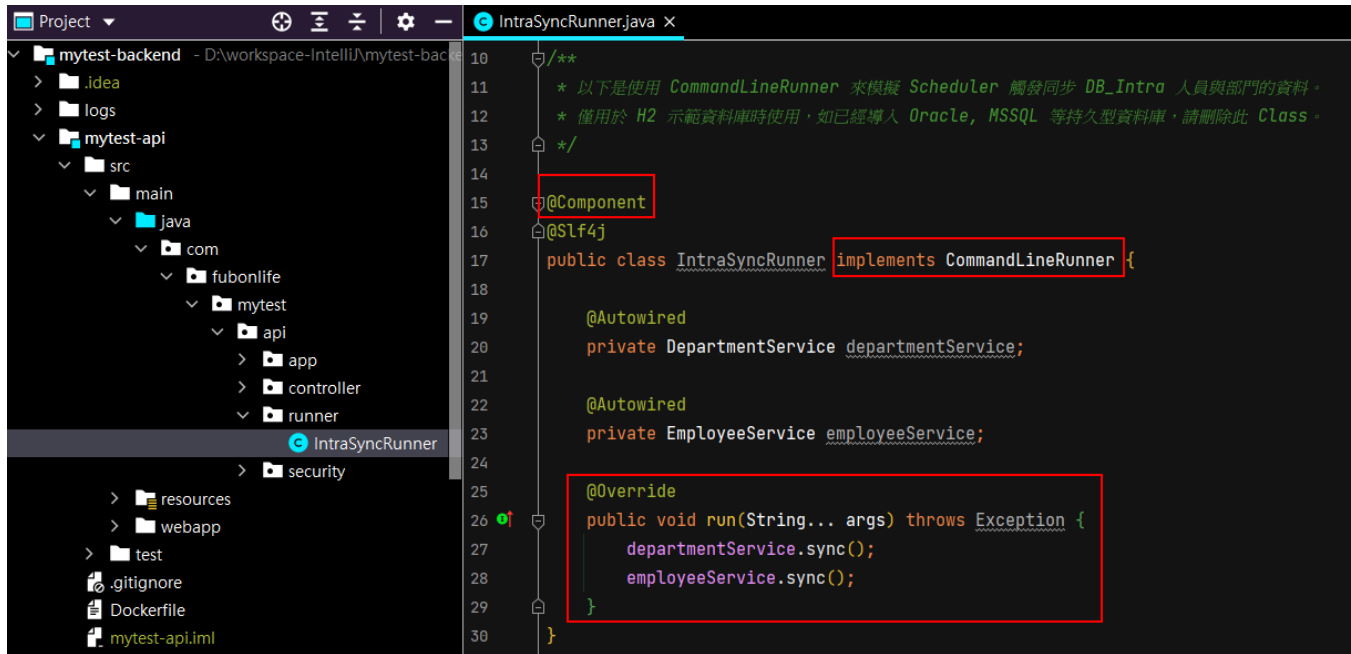


# 介紹 CommandLineRunner

在專案的實際開發中，有時需要專案服務啟動時載入一些資料，或預先完成某些動作

為了解決這樣的問題，Spring boot 提供了一個方法：通過實現介面 `CommandLineRunner` 來實現這樣的需求（無需其他配置）



```
10  /**
11   * 以下是使用 CommandLineRunner 來模擬 Scheduler 觸發同步 DB_Intra 人員與部門的資料。
12   * 僅用於 H2 示範資料庫時使用，如已經導入 Oracle, MSSQL 等持久型資料庫，請刪除此 Class。
13   */
14
15  @Component
16  @Slf4j
17  public class IntraSyncRunner implements CommandLineRunner {
18
19      @Autowired
20      private DepartmentService departmentService;
21
22      @Autowired
23      private EmployeeService employeeService;
24
25      @Override
26      public void run(String... args) throws Exception {
27          departmentService.sync();
28          employeeService.sync();
29      }
30  }
```

⚠ 如果 `CommandLineRunner` 使用不當，就會導致程式出現一些奇怪的異常，更有可能導致應用系統直接停止服務....

情境一：`CommandLineRunner` 執行時間太久了，資料還沒有初始化完成就訪問 API

在 Spring 容器起來之後開始執行，此時 Tomcat 已經可以正常接收請求，但 `CommandLineRunner` 的執行時間過長，資料還沒有初始化完成就訪問 API，此時就開始出錯了

情境二：`CommandLineRunner` 去初始化了一個很耗時的資源，而在初始化資源的時候，又不小心報錯了

Spring 容器會自動關閉，應用系統會停止服務

情境三：`CommandLineRunner` 執行報錯了

Spring 容器會自動關閉，應用系統會停止服務



IntraSyncRunner 是來模擬 Scheduler 觸發同步 DB\_Intra 人員與部門的資料

僅用於 H2 示範資料庫時使用

如已經導入 Oracle, MSSQL 等持久型資料庫，請記得要手動封印 !!