* Config Server: <https://ithelp.ithome.com.tw/articles/10192773>

1. Mave dependency:

|  |
| --- |
| spring-cloud-config-server |

1. application.properties

spring.cloud.config.server.git.uri= file://D:/Bill/workspace/Spring/Spring Cloud/config-repo

server.port=8080

* + - spring.cloud.config.server.git.uri

設定應用起動時從哪裡讀取設定檔，可以從Github，也可以從本地Git檔案

Local: file://{$path}

Remote: <https://github.com/spring-cloud-samples/config-repo>

Local需要先使用SourceTree等工具在本機建立Repository

* + - 建立application.yml

位置: D:/Bill/workspace/Spring/Spring Cloud/config-repo

* + - 建立reservation-service.properties

位置: D:/Bill/workspace/Spring/Spring Cloud/config-repo

* + - 加上@EnableConfigServer就可以啟動ConfigServer的功能了，執行 <http://localhost:8888/reservation-service/master>即可看到對應的設定檔response
* Config Client: <https://ithelp.ithome.com.tw/articles/10192937>

1. Mave dependency:

|  |  |
| --- | --- |
| spring-boot-starter-actuator | spring-boot-starter-data-jpa |
| spring-boot-starter-data-rest | spring-boot-starter-web |
| spring-cloud-starter-config | h2 |

1. 把原本的application.properties重新命名為bootstrap.properties並改成以下內容

spring.application.name=reservation-service

spring.cloud.config.uri=http://localhost:8888

* spring.application.name 應用自己的名稱，到時候可以從介面上看到，也必須對應到設定檔的名稱
* pring.cloud.config.uri Config-Server的位置

1. 程式利用

@Value("${message}")

private String message;

取得設定檔變數

1. 將ConfigServer，ConfigClient啟動，執行<http://localhost:8000/message> 即可輸出設定檔內容