技術筆記

筆記更新紀錄

|  |  |  |
| --- | --- | --- |
| 日期 yyyy/MM/dd | 更新人員 | 更新備註 |
| 2020/01/19 | Miles | 建立文件 |
| 2020/01/20 | Miles | 更新HTTP相關解說，添加TCP/IP相關解說 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

目錄

[1 Session簡介 3](#_Toc33634017)

[2 Session 原理 3](#_Toc33634018)

[3 Session 與 Cookie 4](#_Toc33634019)

## 1 Session簡介

|  |
| --- |
| Session 是什麼 |
| Session 是什麼 ? Session是一個存放Client相互溝通的機制，又或許是一個設計模式。 |
| 因為Server端無法一直保持著與Client的連線狀態，因此而誕生了Session這個機制。 |

## 2 Session 原理

|  |
| --- |
| Session機制原理 |
| 打個比方，我們今天去店家排隊買東西，並且取得了號碼牌，當你暫時離開店家之後，店家就已經忘記你是誰了，這時候就必須靠你手上的這張號碼牌來認定你就是他們的顧客、你要買商品。 |
| 如何讓Client取得號碼牌，一者是使用Cookie，另一個是直接輸出並嵌入頁面之中的方法。 |

## 3 Session 與 Cookie

|  |
| --- |
| Sesseion 與 Cookie 互動方式 |
| session 存放在伺服端，主要是讓Web應用程式可以看似記得是瀏覽器發出的請求，連接數個請求間的關係。  記得:我覺得應該是說這個存放資訊的箱子，並且放在伺服端內。 |
| 至於所謂的連接數個請求的關係這個部分由Web容器幫你執行。 |
| Session 裡面會存放Session Id |
| 以下參考    瀏覽器發送請求會一併帶Session Id去接收端  如果沒有Session Id的時候Web 容器會產生一個Session物件與Session Id，並且將這個Session Id回傳給瀏覽器，而瀏覽器會將其Session Id存放在Cookie中 |

## 4 Session 實作

|  |
| --- |
| 使用Spring Boot 當作一個範例 |
|  |
|  |