Part A

2 practice sql commands

Show databases:

Text, letter

Description automatically generated

Use users

Text

Description automatically generated

Show tables

Graphical user interface, application

Description automatically generated

Select with where

Table

Description automatically generated

Select with WHERE 2

Text, letter

Description automatically generated

Update

Text, letter

Description automatically generated

Select with comments

Diagram

Description automatically generated with medium confidence

Select with multiple conditions

Diagram

Description automatically generated

Part B

1.

Table

Description automatically generated

The success of the injection was due to in the user name field I input string “admin’ #”, the # character is a comments sign in SQL which will cause the query to ignore all subsequent conditions.

Use command line to inject

Graphical user interface, text, application

Description automatically generated

Returned result:

Text, letter

Description automatically generated

Part C

Update Alice’s nickname and password

Table

Description automatically generated

How was it done?

1. Login with Alice’s username with the injection method in part 2
2. Update through unsafe\_edit\_front.php the nickname and password

Part D

Answer: in the safe home page, it uses parameter binding rather than sql query string conjunction, which is able to prevent sql injection.

Access through safe home

A picture containing text, receipt

Description automatically generated

Result

Text, letter

Description automatically generated