Output Data

SQL injection attacks can occur to a poorly protected database. An SQL injection is where a database's queries are interfered with by an attacker. The attacker will take advantage of any web security vulnerabilities in the database and attempt to sabotage or manipulate the existing data. A lack of proper coding can make a person vulnerable to a SQL injection. Parameters, field values and cookies can all be vulnerable to attack. One measure to combat SQL attacks is to have stored procedures. Stored procedures are SQL statements that produce a logical unit to perform a specific task. Like tables, they are database objects. Advantages of stored procedures include effective validation, encapsulation of business logic in a single entity and faster exception handling. A stored procedure is used in the database for its protection.

Queries

Query 1 | 2.2

SELECT Orders.Order_Time, Customer.*

FROM Orders

RIGHT JOIN Customer ON Orders. Customer Email = Customer. Customer Email;



Query 2 | 2.3

SELECT Staff.*, Orders.*

FROM Staff

LEFT JOIN Orders ON Staff.Staff Email = Orders.Staff Email;



Ouerv 3 | 2.4

SELECT Orders.*, Customer.Customer_FirstName, Customer.Customer_LastName

FROM Orders

INNER JOIN Customer ON Orders.Customer_Email = Customer.Customer_Email;

Order_Time	Customer_Email	Service_Name	Staff_Email	Total_Cost	Discount	Quantity	Customer_FirstName	Customer_LastName
2019-08-16 11:08:1	mat.perry-6914@yahoo.com	Hardware Repairs	theolivia@outlook.com	42.00	No	7	Mathew	Perry
2019-08-22 09:09:2	kelenoqagy-5121@outlook.com	PC Building	clarkeff@gmail.com	75.00	No	1	Hellen	Borrke
2019-08-24 11:42:2	ruzooz88@outlook.com	Training	thehutch.s@gmail.com	70.00	No	1	Zoe	Cruz
2019-08-27 13:55:2	4 nazossuzibe-4163@yahoomail.com	Customer Support & Advice Subscription	stanjason222@gmail.com	10.00	No	1	Natalie	Zenia
2019-08-30 11:06:1) jones.gina11@yahoo.com	Antivirus & Firewall Setup	waraddis55@outlook.com	50.00	No	2	Gina	Jones
2019-08-30 14:23:4	4 ellusuddob-6914@gmail.com	Cloud Backup	terry12@gmail.com	25.00	No	1	Ellen	Suddon
2019-09-05 10:14:3	1 myaic@gmail.com	Upgrading	walker.nathan@yahoo.com	90.00	No	2	Mia	Smith
2019-09-07 16:30:0	1 owatehox-6397@gmail.com	Cable Management	theolivia@outlook.com	20.00	No	2	Omar	Nineve
2019-09-08 15:35:1	stee.corrig@yahoo.com	Training	abigailzz@gmail.com	70.00	No	5	Steve	Corrigan
2019-09-14 12:09:2	1 mike.fimme-3910@outlook.com	Software	stanjason222@gmail.com	60.00	No	3	Micheal	Winnifred
2019-09-14 15:17:1	3 richardlang@gmail.com	Cable Management	Chambers9@outlook.com	40.00	No	4	Richard	Langford
2019-09-29 15:12:4	pete.chappers432@outlook.com	Computer Optimisation	abigailzz@gmail.com	30.00	No	1	Peter	Chapman
2019-09-30 09:20:5	1 aquttolin-8530@yopmail.com	Antivirus & Firewall Setup	williams.ian555@yahoo.com	25.00	No	1	Andrea	Hester
2019-10-02 11:45:2	3 day.simon22@outlook.com	Reboot PC	clifden12@outlook.com	30.00	No	1	Simon	Day
2019-10-04 09:58:3	3 wright.alice99@gmail.com	Cloud Backup	clifden12@outlook.com	25.00	No	1	Alice	Wright

Stored Procedure

DELIMITER \$\$

CREATE PROCEDURE 'selectOrders'() NOT DETERMINISTIC CONTAINS SQL SQL SECURITY DEFINER SELECT * FROM Orders;

```
CALL `selectOrders`();

SET @p0='2019-09-05 10:14:34';

CALL `selectOrders`(@p0, @p1);

SELECT @p1 AS `Quantity2`;

END$$

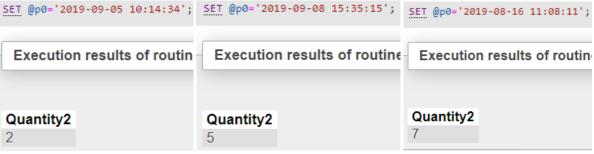
DELIMITER;
```

The main advantage of this stored procedure is that it requires the input of a specific date and time to access the quantity of the order. This makes it less likely to be accessed by an intruder as they are

unlikely to have the exact date and time of the order they want to alter. The images below show its

results:

Call selectOrders;



```
Your SQL query has been executed successfully.

1 row affected by the last statement inside the procedure.

SET @p0='2019-09-05 10:14:34'; CALL `selectOrders`(@p0, @p1); SELECT @p1 AS `Quantity2`;

Execution results of routine `selectOrders`

Quantity2
2
```