

SIT103 Assignment 2

Team member	ID	Contribution (%)
Deon Perera	5005821	50
Taylor Greig	5007802	50

Q1 | Creating the implementation view

STAFF (staffID: char(9), firstName: varchar2(25) , lastName: varchar2(25), email: varchar2(50), phone: int, jobTitle: varchar2(50), hireDate: date, monthlySalary: number)

SUPPLIER (supplierID: char(9), name: varchar2(25), ABN: int, phone: int, email: varchar2(50), address: varchar2(50))

SHIP (shipID: char(9), ETA: date, deliveryStatus: char(1))

INVOICE (invoiceID: char(9), shipID: char(9), staffID: char(9) cost: number, address: varchar2(50), product: varchar2(30), productQuantity: varchar2(2))

PRODUCT (productID: char(9), invoiceID: char(9), make: varchar2(30), model: varchar2(30), classification: varchar2(20), productSize: number, gender: char(1), cost: number, availableQuantity: varchar2(5))

CUSTOMER (customerID: char(9), userName: varchar2(25), password: varchar2(30), dateRegistered: date, dateClosed: date, firstName: varchar2(50), lastName: varchar2(50), address: varchar2(30), City: varchar2(30), postCode: varchar2(12), email: varchar2(50), phone: int, country: varchar2(25))

PRODUCT_ORDER (orderID: char(9), productID: char(9), customerID: char(9) , cost: number, orderDate: date, status: char(1))

PAYMENT (paymentID: char(9), customerID: char(9), accNumber: varchar2(20), BSB: char(9))

Q2 | Implementation

CREATE TABLE Scripts (VSCODE)

```
CREATE TABLE STAFF (  
    staffID char(9) NOT NULL,  
    firstName varchar2(25) NOT NULL,  
    lastName varchar2(25) NOT NULL,  
    email varchar2(50),  
    phone int NOT NULL,  
    jobTitle varchar2(50) NOT NULL,  
    hireDate date NOT NULL,  
    monthlySalary number NOT NULL,  
    CONSTRAINT PK_STAFF PRIMARY KEY(staffID)  
);  
  
CREATE TABLE SUPPLIER (  
    supplierID char(9) NOT NULL,  
    name varchar2(25) NOT NULL,  
    ABN int NOT NULL,  
    phone int NOT NULL,  
    email varchar2(50) NOT NULL,  
    address varchar2(50) NOT NULL,  
    CONSTRAINT PK_SUPP PRIMARY KEY(supplierID)  
);  
  
CREATE TABLE SHIP (  
    shipID char(9) NOT NULL,  
    ETA date,  
    deliveryStatus char(1) NOT NULL,  
    CONSTRAINT PK_SHIP PRIMARY KEY(shipID)  
);  
  
CREATE TABLE INVOICE (  
    invoiceID char(9) NOT NULL,  
    shipID char(9) NOT NULL,  
    staffID char(9) NOT NULL,  
    cost number,  
    address varchar2(50),  
    product varchar2(30),  
    productQuantity varchar2(2),  
    CONSTRAINT PK_INVO PRIMARY KEY(invoiceID),  
    CONSTRAINT FK_INVO FOREIGN KEY(shipID)  
        REFERENCES SHIP(shipID),  
    CONSTRAINT FK2_INVO FOREIGN KEY(staffID)  
        REFERENCES STAFF(staffID)  
);  
  
CREATE TABLE PRODUCT (  
    productID char(9) NOT NULL,  
    invoiceID char(9) NOT NULL,  
    make varchar2(30) NOT NULL,  
    model varchar2(30) NOT NULL,  
    classification varchar2(20),  
    productSize number NOT NULL,  
    gender char(1) NOT NULL,  
    cost number NOT NULL,  
    availableQuantity int NOT NULL,  
    CONSTRAINT PK_PROD PRIMARY KEY(productID),  
    CONSTRAINT FK_PROD FOREIGN KEY(invoiceID)  
        REFERENCES INVOICE(invoiceID)  
);
```

```
CREATE TABLE CUSTOMER (  
    customerID char(9) NOT NULL,  
    userName varchar2(25) NOT NULL,  
    password varchar2(30) NOT NULL,  
    dateRegistered date NOT NULL,  
    dateClosed date,  
    firstName varchar2(50) NOT NULL,  
    lastName varchar2(50) NOT NULL,  
    address varchar2(30) NOT NULL,  
    city varchar2(30) NOT NULL,  
    postCode varchar2(12) NOT NULL,  
    email varchar2(50) NOT NULL,  
    phone int,  
    country varchar2(25) NOT NULL,  
    CONSTRAINT PK_CUST PRIMARY KEY(customerID)  
);  
  
CREATE TABLE PRODUCT_ORDER (  
    orderID char(9) NOT NULL,  
    productID char(9) NOT NULL,  
    customerID char(9) NOT NULL,  
    cost number NOT NULL,  
    orderDate date NOT NULL,  
    status char(1) NOT NULL,  
    CONSTRAINT PK_ORDR PRIMARY KEY(orderID),  
    CONSTRAINT FK1_ORDR FOREIGN KEY(customerID)  
    | REFERENCES CUSTOMER(customerID),  
    CONSTRAINT FK2_ORDR FOREIGN KEY(productID)  
    | REFERENCES PRODUCT(productID)  
);  
  
CREATE TABLE PAYMENT (  
    paymentID char(9) NOT NULL,  
    customerID char(9) NOT NULL,  
    accNumber varchar2(20),  
    BSB char(9),  
    CONSTRAINT PK_PAYM PRIMARY KEY (paymentID),  
    CONSTRAINT FK_PAYM FOREIGN KEY (customerID)  
    | REFERENCES CUSTOMER(customerID)  
);
```


INSERT Scripts (VSCODE)

```
1-insert-staff.sql ×
1-insert-staff.sql
1  INSERT INTO
2  STAFF (staffID,firstName,lastName,email,phone,jobTitle,hireDate,monthlySalary)
3  VALUES (530693059,'Elise','Foster','e.foster@randatmail.com',1197273774,'BE','26-SEP-10',4166)
4  ;
5
6  INSERT INTO
7  STAFF (staffID,firstName,lastName,email,phone,jobTitle,hireDate,monthlySalary)
8  VALUES (894140352,'Chloe','Bennett','c.bennett@randatmail.com',8162026847,'BE','21-JAN-12',4166)
9  ;
10
11 INSERT INTO
12 STAFF (staffID,firstName,lastName,email,phone,jobTitle,hireDate,monthlySalary)
13 VALUES (996767097,'Melissa','Nelson','m.nelson@randatmail.com',9873245135,'MN','17-JUL-12',6250)
14 ;
15
16 INSERT INTO
17 STAFF (staffID,firstName,lastName,email,phone,jobTitle,hireDate,monthlySalary)
18 VALUES (744946969,'Aida','Andrews','a.andrews@randatmail.com',0751612728,'SC','08-FEB-17',3335)
19 ;
20

2-insert-supp.sql ×
2-insert-supp.sql
1  INSERT INTO
2  SUPPLIER (supplierID,name,ABN,phone,email,address)
3  VALUES (495905806,'Shoes Dash',14599527869,0421281555,'ShoesDash@gmail.com','8 Donnelly Ave, Norlane, VIC 3214'
4  );
5
6  INSERT INTO
7  SUPPLIER (supplierID,name,ABN,phone,email,address)
8  VALUES (351271208,'Shoes Plush',84831085360,0353996645,'ShoesPlush@gmail.com','89 Moodemere St, Noble Park, VIC 3174'
9  );
10
11 INSERT INTO
12 SUPPLIER (supplierID,name,ABN,phone,email,address)
13 VALUES (258183609,'Shoquipo',85385281504,0745224502,'Shoquipo@gmail.com','21 Donoghoe Crs, Queanbeyan East, NSW 2620'
14 );
15
```

```
3-insert-ship.sql ×
3-insert-ship.sql
1  INSERT INTO
2  SHIP (shipID,ETA,deliveryStatus)
3  VALUES (773169875,'23-Dec-21','D'
4  );
5
6
7  INSERT INTO
8  SHIP (shipID,ETA,deliveryStatus)
9  VALUES (588511160,'15-Mar-22','P'
10 );
11
12
13  INSERT INTO
14  SHIP (shipID,ETA,deliveryStatus)
15  VALUES (960508009,'05-May-22','P'
16 );
17

2-insert-supp.sql  4-insert-invoice.sql ×
4-insert-invoice.sql
1
2  INSERT INTO
3  INVOICE (invoiceID,shipID,staffID,cost,address,product,productQuantity)
4  VALUES (170853235,773169875,530693059,704,'113 Chatsworth Drive, CARLISLE, 6101, Australia','Red Max 270',4
5  );
6
7  INSERT INTO
8  INVOICE (invoiceID,shipID,staffID,cost,address,product,productQuantity)
9  VALUES (442405550,588511160,894140352,335,'46 Lane Street, BALWYN, 3103, Australia','Green Max 220',5
10 );
11
12  INSERT INTO
13  INVOICE (invoiceID,shipID,staffID,cost,address,product,productQuantity)
14  VALUES (519205148,960508009,996767097,348,'106 Ageston Road, MYSTERTON, 3351, Australia','Green 1 Mid',2
15 );
16
17
18  INSERT INTO
19  INVOICE (invoiceID,shipID,staffID,cost,address,product,productQuantity)
20  VALUES (97589600,753078719,744946969,130,'115 Barker Street, WALBUNDRIE, 2642, Australia','Force 1',2
```

```
5-insert-prod.sql ×
5-insert-prod.sql
1  INSERT INTO
2  PRODUCT (productID,invoiceID,make, model,classification,productSize,gender,cost,availableQuantity)
3  VALUES (685020547,170853235
4  ,'Nike','Red Max 270','Runner',13,'F',200,24
5  );
6
7  INSERT INTO
8  PRODUCT (productID,invoiceID,make, model,classification,productSize,gender,cost,availableQuantity)
9  VALUES (920402458,442405550
10 ,'Jordan','Green Max 220','Trainer',16,'F',91,12
11 );
12
13
14  INSERT INTO
15  PRODUCT (productID,invoiceID,make, model,classification,productSize,gender,cost,availableQuantity)
16  VALUES (704583577,519205148
17  ,'Asics','Green 1 Mid','Runner',11,'F',134,12

2-insert-supp.sql  6-insert-cust.sql ×
6-insert-cust.sql
1
2  INSERT INTO
3  CUSTOMER (customerID,orderID,userName,password,dateRegistered,dateClosed,firstName,lastName,address,city,postCode,email,phone,country)
4  VALUES(997011801,865380134,'OldgreecCrook','mVwcdsnz','08-Aug-21','04-Sep-21','Stefan','Mcconnell','113 Chatsworth Drive','CARLISLE','6101','wbsba5@temporary-mail.net',493339747,
5  );
6
7  INSERT INTO
8  CUSTOMER (customerID,orderID,userName,password,dateRegistered,dateClosed,firstName,lastName,address,city,postCode,email,phone,country)
9  VALUES(328285458,884396350,'StatworWidget','TbmCqpRj','27-Jan-2020','01-Jan-2021','Frank','Cline','46 Lane Street','BALWYN',3103,'n71a@temporary-mail.net',483840778,'Australia'
10 );
11
12  INSERT INTO
13  CUSTOMER (customerID,orderID,userName,password,dateRegistered,dateClosed,firstName,lastName,address,city,postCode,email,phone,country)
14  VALUES(530145686,462830131,'Kahikatea','UxLjY5Ya','04-Nov-19','','Adriana','Chandler','106 Ageston Road','MYSTERTON',3351,'ChandlerAdriana@gmail.com',461118219,'Australia'
15 );
16
17  INSERT INTO
18  CUSTOMER (customerID,orderID,userName,password,dateRegistered,dateClosed,firstName,lastName,address,city,postCode,email,phone,country)
19  VALUES(171432124,297081756,'Headmaster','zftbESBw','08-Nov-18','','Jemma','Simon','115 Barker Street','WALBUNDRIE',2642,'SimonJemma@gmail.com',453921210,'Australia'
20 );
```

7-insert-ordr.sql ×

insert-ordr.sql

```
1  INSERT INTO
2  PRODUCT_ORDER(orderID,productID,customerID,cost,orderDate,status)
3  VALUES (865380134,685020547,997011801,1627,'07-Sep-21','P'
4  );
5
6  INSERT INTO
7  PRODUCT_ORDER(orderID,productID,customerID,cost,orderDate,status)
8  VALUES (884396350,920402458,328285458,1919,'30-Sep-21','P'
9  );
10
11 INSERT INTO
12 PRODUCT_ORDER(orderID,productID,customerID,cost,orderDate,status)
13 VALUES (462830131,704583577,530145686,1999,'25-Oct-21','P'
14 );
15
16 INSERT INTO
17 PRODUCT_ORDER(orderID,productID,customerID,cost,orderDate,status)
```

2-insert-supp.sql

8-insert-pay.sql ×

8-insert-pay.sql

```
1  INSERT INTO
2  PAYMENT (paymentID,customerID,accNumber,BSB)
3  VALUES(826349,997011801,76455446,177490499
4  );
5
6  INSERT INTO
7  PAYMENT (paymentID,customerID,accNumber,BSB)
8  VALUES(881988,328285458,11742329,147680589
9  );
10
11 INSERT INTO
12 PAYMENT (paymentID,customerID,accNumber,BSB)
13 VALUES(646257,530145686,23464819,575058914
14 );
15
16 INSERT INTO
17 PAYMENT (paymentID,customerID,accNumber,BSB)
18 VALUES(920879,171432124,65962407,130777572
19 );
20
```

```
SELECT * FROM (SSID)
```

```
SQL> SELECT * FROM Ship;
SHIPID    ETA      Insert D
-----
773169875 23-DEC-21 D
588511160 15-MAR-22 P
960508009 05-MAY-22 P
753078719 16-AUG-21 D
229422601 16-JAN-22 P
673003396 28-SEP-21 P
995106524 09-MAR-22 P
825599595 25-DEC-21 D

8 rows selected.

SQL> SELECT * FROM Invoice;
INVOICEID SHIPID    STAFFID      COST ADDRESS                                PRODUCT                                PR
-----
170853235 773169875 530693059    704 113 Chatsworth Drive, CARLISLE, 6101, Australia Red Max 270                          4
442405550 588511160 894140352    335 46 Lane Street, BALWYN, 3103, Australia Green Max 220                        5
519205148 960508009 969767097    348 106 Ageston Road, MYSTERTON, 3351, Australia Green 1 Mid                          2
97589600 753078719 744946969    130 115 Barker Street, WALBUNDRIE, 2642, Australia Force 1                              2
598274435 229422601 456826902    600 102 Barker Street, WEMBLEY, 6014, Australia Max 270                              3
17889520 673003396 976462721    950 29 Magnolia Drive, YILKARI, 2460, Australia Max 270 Mens                        5
691915009 995106524 491479349    100 24 Magnolia Drive, TIMMERING, 4818, Australia Ultra Boost RDY                     1
527237204 825599595 986942953    450 44 Eshelby Drive, QUIDONG, 2632, Australia Air Max 90                          5
```

```
SQL> SET LIN 80
SQL> SELECT * FROM Staff;
```

STAFFID	FIRSTNAME	LASTNAME	EMAIL	PHONE	JOBTITLE	HIREDATE	MONTHLYSALARY
336693059	Elise	Foster	e.foster@randatmail.com	1197273774	BE	26-SEP-10	4166
894148352	Chloe	Bennett	c.bennett@randatmail.com	8162826847	BE	21-JAN-12	4166
896767097	Melisse	Nelson	m.nelson@randatmail.com	9873245135	MN	17-JUL-12	6250
744946969	Aida	Andrews	a.andrews@randatmail.com	751612728	SC	08-FEB-17	3335
456826992	Stuart	Murray	s.murray@randatmail.com	2789860904	SC	21-JUL-17	3335
476463721	Dale	Taylor	d.taylor@randatmail.com	4307036462	SC	04-SEP-16	3335
491479349	Savana	Cameron	s.cameron@randatmail.com	1208812504	SC	05-MAR-19	3335
886942953	Brooke	Henderson	b.henderson@randatmail.com	3104499255	SC	02-JUN-19	3335
260216361	Justin	Johnson	j.johnson@randatmail.com	1369917351	SC	24-SEP-19	3335
391065384	Valeria	Thompson	v.thompson@randatmail.com	1942679757	SC	25-NOV-19	3335

```
10 rows selected.
```

```
SQL> SELECT * FROM Supplier;
```

SUPPLIER	NAME	ABN	PHONE	EMAIL	ADDRESS
495905806	Shoes Dash	1.4680E+10	4121281555	ShoesDash@gmail.com	8 Donnelly Ave, Norlane, VIC 3214
351271208	Shoes Plush	8.48321E+10	353996645	ShoesPlush@gmail.com	89 Woodmere St, Noble Park, VIC 3174
258183669	Shoquipo	8.5385E+10	745224502	Shoquipo@gmail.com	21 Donoghoe Crs, Queenbeyan East, NSW 2620
397214394	Shox	3.6927E+10	449916938	Shox@gmail.com	5 Carrington St, Ballina, NSW 2478
222115028	SGolds	3.9839E+10	367656502	SGolds@gmail.com	37 Tuncurry St, Tuncurry, NSW 2428
555959908	Agiles	2.1568E+10	361425715	Agiles@gmail.com	33 Prendwick Way, Willeton, WA 6155
557671929	Swayll	1.5094E+10	349565040	Swayll@gmail.com	6 Just Pl, Nicholls, ACT 2913
764686557	Wrap	6.2220E+10	353909077	Wrap@gmail.com	71 Shutehaven Cct, Bushland Beach, QLD 4818

```
8 rows selected.
```


Testing | Writing queries to test your relational database

Q3. (SSID)

```
SQL> desc staff
Name
-----
STAFFID          NOT NULL CHAR(9)
FIRSTNAME        NOT NULL VARCHAR2(25)
LASTNAME         NOT NULL VARCHAR2(25)
EMAIL            VARCHAR2(50)
PHONE            NOT NULL NUMBER(30)
JOBTITLE         NOT NULL VARCHAR2(50)
HIREDATE         NOT NULL DATE
MONTHLYSALARY    NOT NULL NUMBER

SQL> SELECT * FROM Staff;

STAFFID  FIRSTNAME  LASTNAME  EMAIL                                PHONE  JOBTITLE  HIREDATE  MONTHLYSALARY
-----
530693059 Elise      Foster    e.foster@randatmail.com            1197273774 BE  26-SEP-10  4166
884140352 Chloe     Bennett  c.bennett@randatmail.com           8162026847 BE  21-JAN-12  4166
996767097 Melissa  Nelson   m.nelson@randatmail.com            9873245135 MN  17-JUL-12  6250
744946969 Aida      Andrews  a.andrews@randatmail.com           751612728 SC   08-FEB-17  3335
456826902 Stuart   Murray  s.murray@randatmail.com            2789699094 SC   21-JUL-17  3335
976462721 Dale      Taylor   d.taylor@randatmail.com            4307036462 SC   04-SEP-18  3335
491479349 Savana   Cameron  s.cameron@randatmail.com           1208012504 SC   05-MAR-19  3335
986942953 Brooke  Henderson b.henderson@randatmail.com         3104990255 SC   02-JUN-19  3335
260216361 Justin  Johnson  j.johnson@randatmail.com           1369917351 SC   24-SEP-19  3335
391065384 Valeria  Thompson v.thompson@randatmail.com          1942679757 SC   25-NOV-19  3335

10 rows selected.

SQL> ALTER TABLE STAFF ADD emergencyContact varchar2(50);
Table altered.

SQL> ALTER TABLE STAFF ADD emergencyContactNumber int;
Table altered.

SQL> desc staff
Name
-----
STAFFID          NOT NULL CHAR(9)
FIRSTNAME        NOT NULL VARCHAR2(25)
LASTNAME         NOT NULL VARCHAR2(25)
EMAIL            VARCHAR2(50)
PHONE            NOT NULL NUMBER(30)
JOBTITLE         NOT NULL VARCHAR2(50)
HIREDATE         NOT NULL DATE
MONTHLYSALARY    NOT NULL NUMBER
EMERGENCYCONTACT VARCHAR2(50)
EMERGENCYCONTACTNUMBER NUMBER(36)

SQL> SELECT * FROM Staff;

STAFFID  FIRSTNAME  LASTNAME  EMAIL                                PHONE  JOBTITLE  HIREDATE  MONTHLYSALARY
-----
EMERGENCYCONTACT  EMERGENCYCONTACTNUMBER
-----
530693059 Elise      Foster    e.foster@randatmail.com            1197273774 BE  26-SEP-10  4166
884140352 Chloe     Bennett  c.bennett@randatmail.com           8162026847 BE  21-JAN-12  4166
996767097 Melissa  Nelson   m.nelson@randatmail.com            9873245135 MN  17-JUL-12  6250

STAFFID  FIRSTNAME  LASTNAME  EMAIL                                PHONE  JOBTITLE  HIREDATE  MONTHLYSALARY
-----
EMERGENCYCONTACT  EMERGENCYCONTACTNUMBER
-----
744946969 Aida      Andrews  a.andrews@randatmail.com           751612728 SC   08-FEB-17  3335
456826902 Stuart   Murray  s.murray@randatmail.com            2789699094 SC   21-JUL-17  3335
976462721 Dale      Taylor   d.taylor@randatmail.com            4307036462 SC   04-SEP-18  3335

STAFFID  FIRSTNAME  LASTNAME  EMAIL                                PHONE  JOBTITLE  HIREDATE  MONTHLYSALARY
-----
EMERGENCYCONTACT  EMERGENCYCONTACTNUMBER
-----
491479349 Savana   Cameron  s.cameron@randatmail.com           1208012504 SC   05-MAR-19  3335
986942953 Brooke  Henderson b.henderson@randatmail.com         3104990255 SC   02-JUN-19  3335
260216361 Justin  Johnson  j.johnson@randatmail.com           1369917351 SC   24-SEP-19  3335

STAFFID  FIRSTNAME  LASTNAME  EMAIL                                PHONE  JOBTITLE  HIREDATE  MONTHLYSALARY
-----
EMERGENCYCONTACT  EMERGENCYCONTACTNUMBER
-----
391065384 Valeria  Thompson v.thompson@randatmail.com          1942679757 SC   25-NOV-19  3335

10 rows selected.

SQL>
```

```
SQL> SELECT Staff.emergencycontact AS "Emergency Contact", Staff.emergencycontactnumber AS "Phone Number" FROM Staff;
```

Emergency Contact	Phone Number
Anna Walker	449685711
Michael Rodriguez	440822228
Sophie David	438186514
David Fox	449161907
Jerry Givens	483299637
Douglas Taylor	461203627
Ronald Bailey	490691282
Brian Salinas	453053757
Peter Champion	467344229
James Latimer	449900866

10 rows selected.

```
SQL>
```

```
SQL> UPDATE STAFF SET emergencyContact = 'Anna Walker' WHERE Staff.staffID = 530693059;
```

1 row updated.

```
SQL>
```

```
SQL> SELECT Staff.emergencycontact AS "Emergency Contact", Staff.emergencycontactnumber AS "Phone Number" FROM Staff;
```

Emergency Contact	Phone Number
Anna Walker	449685711
Michael Rodriguez	440822228
Sophie David	438186514
David Fox	449161907
Jerry Givens	483299637
Douglas Taylor	461203627
Ronald Bailey	490691282
Brian Salinas	453053757
Peter Champion	467344229
James Latimer	449900866

10 rows selected.

```
SQL>
```

Q4. (SSID)

```
SQL> SELECT Staff.staffid AS "Staff ID", Staff.firstname AS "First Name", Staff.lastname AS "Last Name", Staff.jobtitle AS "Employment Position", Staff.monthlysalary AS "Monthly Salary ($)" FROM Staff ;
```

Staff ID	First Name	Last Name	Employment Position	Monthly Salary (\$)
530693059	Elise	Foster	BE	4166
894140352	Chloe	Bennett	BE	4166
996767097	Melissa	Nelson	MN	6250
744946969	Aida	Andrews	SC	3335
456826992	Stuart	Murray	SC	3335
976462721	Dale	Taylor	SC	3335
491479349	Savanna	Cameron	SC	3335
986942953	Brooke	Henderson	SC	3335
260216361	Justin	Johnson	SC	3335
391065384	Valeria	Thompson	SC	3335

10 rows selected.

```
SQL> UPDATE STAFF SET monthlysalary = monthlysalary + (monthlysalary * 0.0136) WHERE jobTitle = 'MN' ;
```

1 row updated.

```
SQL> UPDATE STAFF SET monthlysalary = monthlysalary + (monthlysalary * 0.0182) WHERE jobTitle = 'SC' ;
```

7 rows updated.

```
SQL> UPDATE STAFF SET monthlysalary = monthlysalary + (monthlysalary * 0.0182) WHERE jobTitle = 'BE' ;
```

2 rows updated.

```
SQL> SELECT Staff.staffid AS "Staff ID", Staff.firstname AS "First Name", Staff.lastname AS "Last Name", Staff.jobtitle AS "Employment Position", Staff.monthlysalary AS "Monthly Salary ($)" FROM Staff ;
```

Staff ID	First Name	Last Name	Employment Position	Monthly Salary (\$)
530693059	Elise	Foster	BE	4208.4932
894140352	Chloe	Bennett	BE	4208.4932
996767097	Melissa	Nelson	MN	6335
744946969	Aida	Andrews	SC	3369.017
456826992	Stuart	Murray	SC	3369.017
976462721	Dale	Taylor	SC	3369.017
491479349	Savanna	Cameron	SC	3369.017
986942953	Brooke	Henderson	SC	3369.017
260216361	Justin	Johnson	SC	3369.017
391065384	Valeria	Thompson	SC	3369.017

10 rows selected.

```
SQL>
```

Q5. (SSID)

```
SQL> SELECT PRODUCT.model FROM PRODUCT WHERE PRODUCT.make = '&Shoe_Maker';
Enter value for shoe_maker: Nike
old 1: SELECT PRODUCT.model FROM PRODUCT WHERE PRODUCT.make = '&Shoe_Maker'
new 1: SELECT PRODUCT.model FROM PRODUCT WHERE PRODUCT.make = 'Nike'

MODEL
-----
Red Max 270
Max 270 Mens

SQL> SELECT PRODUCT.model FROM PRODUCT WHERE PRODUCT.make = '&Shoe_Maker';
Enter value for shoe_maker: Jordan
old 1: SELECT PRODUCT.model FROM PRODUCT WHERE PRODUCT.make = '&Shoe_Maker'
new 1: SELECT PRODUCT.model FROM PRODUCT WHERE PRODUCT.make = 'Jordan'

MODEL
-----
Green Max 220
Max 270

SQL> SELECT PRODUCT.model FROM PRODUCT WHERE PRODUCT.make = '&Shoe_Maker';
Enter value for shoe_maker: Asics
old 1: SELECT PRODUCT.model FROM PRODUCT WHERE PRODUCT.make = '&Shoe_Maker'
new 1: SELECT PRODUCT.model FROM PRODUCT WHERE PRODUCT.make = 'Asics'

MODEL
-----
Green 1 Mid
Force 1

SQL> SELECT PRODUCT.model FROM PRODUCT WHERE PRODUCT.make = '&Shoe_Maker';
Enter value for shoe_maker: Lacoste
old 1: SELECT PRODUCT.model FROM PRODUCT WHERE PRODUCT.make = '&Shoe_Maker'
new 1: SELECT PRODUCT.model FROM PRODUCT WHERE PRODUCT.make = 'Lacoste'

MODEL
-----
Ultra Boost RDY
Air Max 90

SQL> 
```

Q6. (SSID)

```
q6.sql > ...
1  SELECT VALUE FROM NLS_SESSION_PARAMETERS
2  WHERE PARAMETER = 'NLS_DATE_FORMAT';
3
4  ALTER SESSION
5  SET NLS_DATE_FORMAT = 'YYYY-MM-DD';
6
7  CREATE VIEW Q6 AS
8  SELECT
9  Customer.lastname AS "Customer Surname",
10 Product.make AS "Make",
11 Product.model AS "Model",
12 Product.cost AS "List Price",
13 Product_order.orderdate AS "Order Date"
14 FROM Customer, Product_order, Product
15 WHERE Customer.customerid = Product_order.customerid
16 AND Product_order.productID = Product.productID
17 AND Customer.customerID = 328285458
18 ORDER BY Product_order.orderdate DESC
19 ;
20
21 COLUMN "List Price" FORMAT $999,999.99
22
23 SELECT * FROM Q6;
24
25 SELECT SUM("List Price") AS "Total Amount Spent 2021"
26 FROM Q6;
```

```
SQL> SELECT VALUE FROM NLS_SESSION_PARAMETERS WHERE PARAMETER = 'NLS_DATE_FORMAT';

VALUE
-----
YYYY-MM-DD

SQL> ALTER SESSION SET NLS_DATE_FORMAT = 'YYYY-MM-DD';

Session altered.

SQL> CREATE VIEW Q6 AS SELECT Customer.lastname AS "Customer Surname", Product.make AS "Make", Product.model AS "Model",
Product.cost AS "List Price", Product_order.orderdate AS "Order Date" FROM Customer, Product_order, Product WHERE Customer.cu
stomerid = Product_order.customerid AND Product_order.productID = Product.productID AND Customer.customerID = 328285458 ORDER B
Y Product_order.orderdate DESC ;

View created.

SQL> COLUMN "List Price" FORMAT $999,999.99
SQL> SELECT * FROM Q6;

Customer Surname      Make      Model      List Price Order Date
-----
Cline                 Lacoste   Ultra Boost RDY      $62.00 2021-12-15
Cline                 Jordan    Max 270              $188.00 2021-10-22
Cline                 Nike      Red Max 270          $200.00 2021-10-01
Cline                 Jordan    Green Max 220         $91.00 2021-09-30

SQL> SELECT SUM("List Price") AS "Total Amount Spent 2021" FROM Q6;

Total Amount Spent 2021
-----
541

SQL>
```

Q7. (SSID)

This script is designed to streamline and increase staff productivity by creating a view of a query that can be referenced by staff.

This algorithm will search for purchases that have not been dispatched to the customer.

The algorithm functions by searching for items in the PRODUCT_ORDER.status that do not have the status "C" (abbreviated to completed).

PRODUCT_ORDER.status	Implication
C	Completed
P	Pending

If the product order has not been completed, a staff member will be designated to prepare and dispatch the purchased products. The order is recognised as "completed" when the product has been dispatched,

This algorithm displays the customer's name, the item purchased, the quantity purchased and their address. Furthermore, it displays the staff member designated with forwarding the order for delivery.

```
q7.sql > VIEW dispatch
1  column 'Quantity' format A9
2  column 'Customer Name' format A20
3  column 'Staff Name' format A20
4
5  CREATE VIEW dispatch AS
6      SELECT
7          Customer.firstName || ' ' || Customer.lastName AS "Customer Name",
8          Product.model AS "Purchased",
9          Invoice.productQuantity "Quantity",
10         Invoice.address AS "Address",
11         Staff.firstName || ' ' || Staff.lastName AS "Staff Assigned"
12     FROM Customer, Product_order, Product, Invoice, Staff
13     WHERE Customer.customerID = Product_order.customerID
14           AND Product_order.status <> 'C'
15           AND Product_order.productID = Product.productID
16           AND Product.invoiceID = Invoice.invoiceID
17           AND Invoice.staffID = Staff.staffID
18     ORDER BY Product_order.orderDate DESC
19 ;
```

```
SQL> @q7
View created.
SQL> SELECT * FROM dispatch;
```

Customer Name	Purchased	Quantity	Address	Staff Assigned
Jemma Simon	Force 1	2	115 Barker Street, WALBUNDRIE, 2642, Australia	Aida Andrews
Adriana Chandler	Green 1 Mid	2	106 Ageston Road, MYSTERTON, 3351, Australia	Melissa Nelson
Elliott Stacey	Max 270 Mens	5	29 Magnolia Drive, YILKARI, 2460, Australia	Dale Taylor
Frank Cline	Green Max 220	5	46 Lane Street, BALWYN, 3103, Australia	Chloe Bennett
Stefan Mcconnell	Red Max 270	4	113 Chatsworth Drive, CARLISLE, 6101, Australia	Elise Foster

```
SQL>
```