Database Project Assignment

Group Member: Julia Calma 22167597 & Taehee Oh 22160265 | Group ID: 114

PART A

TASK 1: IDENTIFY ENTITIES & DEVELOP BUSINESS RULES

ENTITIES:

Task 1 | ENTITIES

1	**Customer**	Stores individual customer's information who book photography session and order photography, such as personal ID, name, mobile number, address, and email address.
2	**Booking**	A booking for a photography session must be made two months in advance. The booking entity includes the customer preferred date, session type (eg. graduation, wedding), progress of payment, and such.
3	**Photography Session**	Each photography session will have different date and time according to customer's or staff's request. These data can be store in attributes such as session id, date and time, and charge amount.
4	**Location**	The location entity is where a photography session take place that is entirely determined by the costumer, this can either be in the beach, restaurants, city etc. Its attribute are as follows: id, name, street, suburb, city, and postcode.
5	**Digitize Photograph**	The photograph entity is the digitised photo captured in the photograph session. Each photo has its own photo number, date, time, and more.
6	**Photo Combination**	This entity contains the combinations the customers want for each image. This comes with the attributes: id, photocolour, photosize,finishtype, and price.
7	**Orders**	Individual's customer's order for photography can be stored. Storing attributes such as ordered photos, collection method, and total cost.
8	**Destination Location**	The destination location entity is where the courier delivers the photo ordered by the customer. This entity contains the order number, postal address, and charge.
9	**Staff**	Staffs information is stored, details like person ID, name, mobile number, their role.

BUSINESS RULE:

```
Format: Each/A/An ENTITY_1 May/Must Relationship_Verb_Phrase number ENTITY_2
```

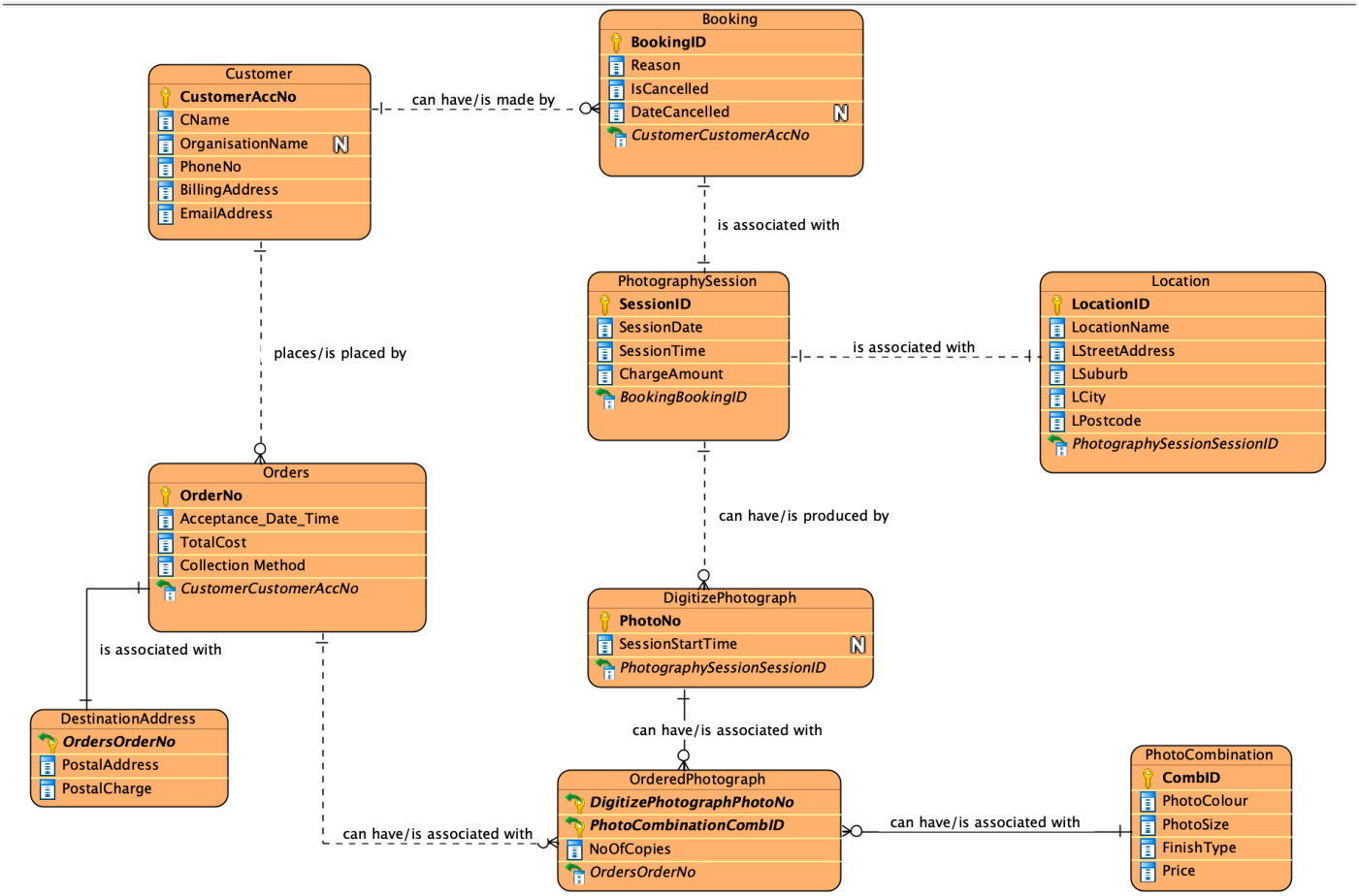
Task 1 | BUSINESS RULES

1	Customer may book multiple bookings.
2	Bookings must be booked by a customer.
3	A photography session must be associated with a booking.
4	A booking must be associated with a photography session.
5	A photography session must be associated with a location.
6	A location must be associated with a photography session.
7	A photography session may have multiple digitised photographs.
8	Multiple digitised photographs must be produced by a photography session.
9	Customer may place multiple Orders.
10	Orders must be placed by a customer.
11	Orders may contain multiple Photographs.
12	Photograph may be part of multiple Orders.
13	Orders may have a Destination Location for couriered orders.
14	Destination Location must be associated with an order.
15	Orders may be picked up.
16	A staff may be associated with multiple Photography Sessions.
17	Photography Session may be handled by a staff.
18	Staff may be associated with multiple Orders.
19	Order may be handled by a staff.
20	Each ordered photographs may have multiple photo combinations
21	Each photo combination must be associated with an ordered photograph.

TASK 2: CONSTRUCT A LOGICAL ENTITY RELATIONSHIP DIAGRAM (ERD)

Attributes names				
Table Name	**Atrribute Name**	**Contents**	**Type**	**PK / FK**
Customer	CustomerAccNo	Customer Account Number	Number	PK
	CName	Customer Name	Character	
	OrganisationName	Organisation Name	Character	
	PhoneNo	Customer Phone Number	Character	
	BillingAddress	Customer Billing Address	Character	
	EmailAddress	Customer Email Address	Character	
Photography Session	SessionID	Session ID	Number	PK
	SessionDate	Session Date	Date	
	SessionTime	Session Time	Character	
	ChargeAmount	Session Charge Amount	Number	
	BookingBookingID	Booking ID	Number	FK
Orders	OrderNo	Order Number	Number	PK
	Acceptance_Date_Time	Order Date Acceptance	Timestamp	
	TotalCost	Total Cost	Number	FK
	CollectionMethod	Collection Method	Character	
	CustomerAccNo	Customer Account Number	Number	
Destination Location:	OderNo	Oder Number	Number	PK
	PostalAddress	Postal Address	Character	
	PostalCharge	Postal Charge	Double	
Location:	LocationID	Location ID	Number	PK
	LocationName	Location Name	Character	
	LStreet	Location Street Name	Character	
	LSuburb	Location Suburb Name	Character	
	LCity	Location City Name	Character	
	LPostcode	Location Post Code	Character	
Booking:	BookingID	Booking ID	Number	PK
	Reason	Reason for Booking	Character	
	IsCancelled	Booking Cancellation Status	Character	
	DateCancelled	Booking Cancelled Date	Date	
	CustomerAccNo	Customer Account Number	Number	FK
Digitize Photograph	PhotoNo	Photography Number	Number	PK
	SessionStartTime	Session Start Time	Character	
	PhotographySessionID	Photography Session ID	Number	FK
Ordered Photography	DigitizePhotographyNo	Digitize Photography Number	Number	PK
	PhotoCombinationID	Photography Combination ID	Number	PK
	NoOfCopies	Number of Copies	Number	
	OrderNo	Oder Number	Number	FK
Photo Combination	CombID	Combination ID	Number	PK
	PhotoColour	Photo Colour	Character	
	PhotoSize	Photo Size	Character	
	FinishType	Finish Type	Character	
	Price	Photo Price	Number	

ERD DIAGRAM:



Note:

Added the ***IsCancelled*** & ***DateCancelled*** attributes to the BOOKING entity because it's needed for whenever a customer wants to cancel the booking so that the company have the database (which is the dates) to calculate if they can get a refund. E.g. customer can get a refund if they cancel the on the web page at least three days in advance before the booking.

SessionType: Removed this attribute from the PHOTOGRAPH SESSION entity because the ***Reason*** attribute why they booked a photography session is already provided in the BOOKING entity so it is not needed because the database will only become repetitive

Staff: Removed this entity because the case study doesn't require that it saves/show the staff who handles the photography session in the database.)(*the staff was only mentioned in the case study to show that the company were understaffed so they needed a website)*

SessionDate:(I removed this because this attribute is already on the PHOTOGRPAHY SESSION entity so I just linked the session ID which is from the PHOTOGRAPH SESSION ENTITY to the PHOTOGRAPH entity.

PART B:

TASK 3: CREATE TABLES

```
CREATE TABLE customer
(CUSTOMERACCNO NUMBER(6) NOT NULL,
CNAME VARCHAR2(30)NOT NULL,
ORGANISATIONNAME VARCHAR2(30),
PHONENO VARCHAR2(20),
EMAILADDRESS VARCHAR2(30),
BILLINGADDRESS VARCHAR2(100) NOT NULL,
CONSTRAINT CUSTOMER_CUSTOMERACCNO PRIMARY KEY(CUSTOMERACCNO));
```

```
CREATE TABLE booking
(BOOKINGID VARCHAR2(6) NOT NULL,
REASON VARCHAR2(50)NOT NULL,
ISCANCELLED VARCHAR2(6),
DATECANCELLED DATE,
CUSTOMERACCNO NUMBER(6) NOT NULL,
CONSTRAINT BOOKING_BOOKINGID PRIMARY KEY(BOOKINGID),
CONSTRAINT BOOKING_CUSTOMERACCNO_FK FOREIGN KEY (CUSTOMERACCNO) REFERENCES
CUSTOMER(CUSTOMERACCNO));
```

```
CREATE TABLE ORDERS
(ORDERNO VARCHAR2(6) NOT NULL,
ACCEPTANCE_DATE_TIME DATE NOT NULL ,
TOTALCOST NUMBER(6,2) NOT NULL,
COLLECTIONMETHOD VARCHAR2(20)NOT NULL,
CUSTOMERACCNO NUMBER(6) NOT NULL,
CONSTRAINT ORDERS_ORDERNO PRIMARY KEY(ORDERNO),
CONSTRAINT ORDERS_CUSTOMERACCNO_FK FOREIGN KEY (CUSTOMERACCNO) REFERENCES
CUSTOMER(CUSTOMERACCNO));
```

```
CREATE TABLE destinationAddress
(ORDERNO VARCHAR2(6) NOT NULL,
POSTALADRESS VARCHAR2(100) NOT NULL,
POSTALCHARGE NUMBER(6,2) NOT NULL,
CONSTRAINT destinationAddress_ORDERNO PRIMARY KEY(ORDERNO),
CONSTRAINT destinationAddress_ORDERNO_FK FOREIGN KEY (ORDERNO) REFERENCES
ORDERS(ORDERNO));
```

```
CREATE TABLE photographySession
(SESSIONID VARCHAR2(6) NOT NULL,
SESSIONDATE DATE NOT NULL,
SESSIONTIME VARCHAR2(50)NOT NULL,
CHARGEAMOUNT NUMBER(6,2)NOT NULL,
BOOKINGID VARCHAR2(6) NOT NULL,
CONSTRAINT PHOTOGRAPHYSESSION_SESSIONID PRIMARY KEY(SESSIONID),
```

```
CONSTRAINT PHOTOGRAPHYSESSION_BOOKINGID_FK FOREIGN KEY (BOOKINGID) REFERENCES
BOOKING(BOOKINGID));

CREATE TABLE location
(LOCATIONID VARCHAR2(6) NOT NULL,
SESSIONID VARCHAR2(6) NOT NULL,
LOCATIONNAME VARCHAR2(20)NOT NULL,
LSTREETADDRESS VARCHAR2(50),
LSUBURB VARCHAR2 (20)NOT NULL,
LCITY VARCHAR2(20)NOT NULL,
LPOSTCODE VARCHAR2(5)NOT NULL,
CONSTRAINT LOCATION_LOCATIONID PRIMARY KEY(LOCATIONID),
CONSTRAINT LOCATION_SESSIONID_FK FOREIGN KEY(SESSIONID) REFERENCES
PHOTOGRAPHYSESSION(SESSIONID));

CREATE TABLE digitizePhotograph
(PHOTONO VARCHAR2(6) NOT NULL,
SESSIONSTARTTIME VARCHAR2(8)NOT NULL,
SESSIONID VARCHAR2(6) NOT NULL,
CONSTRAINT digitizePhotograph_PHOTONO PRIMARY KEY(PHOTONO),
CONSTRAINT digitizePhotograph_SESSIONID_FK FOREIGN KEY(SESSIONID) REFERENCES
PHOTOGRAPHYSESSION(SESSIONID));

CREATE TABLE photoCombination
(COMBID VARCHAR2(6) NOT NULL,
PHOTOCOLOUR VARCHAR2(20) NOT NULL,
PHOTOSIZE VARCHAR2(10) NOT NULL,
FINISHTYPE VARCHAR2(20)NOT NULL,
PRICE NUMBER(6,2)NOT NULL,
CONSTRAINT photoCombination_COMBID PRIMARY KEY(COMBID));


CREATE TABLE orderedPhotograph
(PHOTONO VARCHAR2(6) NOT NULL,
COMBID VARCHAR2(6)NOT NULL,
NOOFCOPIES NUMBER(10) NOT NULL,
ORDERNO VARCHAR2(6) NOT NULL,
CONSTRAINT orderedPhotography_PHOTONO_COMBID PRIMARY KEY (PHOTONO,COMBID),
CONSTRAINT  orderedPhotography_PHOTONO_FK FOREIGN KEY (PHOTONO) REFERENCES
digitizePhotograph(PHOTONO),
CONSTRAINT  orderedPhotography_COMBID_FK FOREIGN KEY (COMBID) REFERENCES
photoCombination(COMBID),
CONSTRAINT orderedPhotography_ORDERNO_FK FOREIGN KEY (ORDERNO) REFERENCES
ORDERS(ORDERNO),
CONSTRAINT orderedPhotography_NOOFCOPIES_CHECK CHECK (NOOFCOPIES >= 0 AND
MOD(NOOFCOPIES, 1) = 0));
```

TASK 4: POPULATE DATA

Customer Table:

Customer AccNo	CName	Organisation Name	PhoneNo	BillingAddress	EmailAddress
120100	Julia Calma	null	0221238095	5 Stanford Street, Albany, Auckland 0632	julia@apuphoto.co.nz
120101	Tay Oh	null	0218905666	34 Queen Street, Auckland CBD, Auckland 1010	tay@apuphoto.co.nz
120102	Sharmaine Valle	COTTON ON	0204583421	90 Akoranga Drive, Northcote, Auckland 0627	sharm@apuphoto.co.nz
120103	Cindy Nguyen	null	0218975609	640 Great South Road, Manukau, Auckland 2025	cindy@apuphoto.co.nz
120104	Althea Macasaol	PIZZA HUT	0217024538	8 Symonds Street, Grafton, Auckland 1010	althea@apuphoto.co.nz
120105	Miles Ocampo	null	0227864432	80 Saint Lukes Road, Mount Albert, Auckland 1025	miles@apuphoto.co.nz
120106	Elizabeth White	null	0209996453	7 Queen Street, Auckland CBD, Auckland 1010	elizabeth@apuphoto.co.nz
120107	Amanda Lee	null	0226754487	1 Beach Road, Takanini, Auckland 2112	amanda@apuphoto.co.nz

```
INSERT INTO Customer
   (CustomerAccNo, CName, OrganisationName, PhoneNo, BillingAddress, EmailAddress)
VALUES
   ('120100', 'Julia Calma', NULL, '0221238095', '5 Stanford Street, Albany, Auckland 0632', 'julia@apuphoto.co.nz');

INSERT INTO Customer
   (CustomerAccNo, CName, OrganisationName, PhoneNo, BillingAddress, EmailAddress)
VALUES
   ('120101', 'Tay Oh ', NULL, '0218905666', '34 Queen Street, Auckland CBD, Auckland 1010', 'tay@apuphoto.co.nz');

INSERT INTO Customer
   (CustomerAccNo, CName, OrganisationName, PhoneNo, BillingAddress, EmailAddress)
VALUES
   ('120102', 'Sharmaine Valle', 'COTTON ON', '0204583421', '90 Akoranga Drive, Northcote, Auckland 0627','sharm@apuphoto.co.nz');

INSERT INTO Customer
   (CustomerAccNo, CName, OrganisationName, PhoneNo, BillingAddress, EmailAddress)
VALUES
   ('120103', 'Cindy Nguyen', NULL, '0218975609', '640 Great South Road, Manukau, Auckland 2025', 'cindy@apuphoto.co.nz');

INSERT INTO Customer
   (CustomerAccNo, CName, OrganisationName, PhoneNo, BillingAddress, EmailAddress)
VALUES
   ('120104', 'Althea Macasaol', 'PIZZA HUT', '0217024538', '8 Symonds Street, Grafton, Auckland 1010', 'althea@apuphoto.co.nz');

INSERT INTO Customer
   (CustomerAccNo, CName, OrganisationName, PhoneNo, BillingAddress, EmailAddress)
VALUES
   ('120105', 'Miles Ocampo', NULL, '0227864432', '80 Saint Lukes Road, Mount Albert, Auckland 1025', 'miles@apuphoto.co.nz');

INSERT INTO Customer
   (CustomerAccNo, CName, OrganisationName, PhoneNo, BillingAddress, EmailAddress)
VALUES
   ('120106', 'Elizabeth White', NULL, '0209996453', '7 Queen Street, Auckland CBD, Auckland 1010', 'elizabeth@apuphoto.co.nz');

INSERT INTO Customer
   (CustomerAccNo, CName, OrganisationName, PhoneNo, BillingAddress, EmailAddress)
VALUES
   ('120107', 'Amanda Lee', NULL, '0226754487', '1 Beach Road, Takanini, Auckland 2112', 'amanda@apuphoto.co.nz');
COMMIT;
```

```
select * from customer;
```

Script Output | Query Result | Query Result 1 | Query Result 2 | Query Result 3 | Query Result 4 | Query Result 5 | Query...

SQL | All Rows Fetched: 8 in 0.112 seconds

	CUSTOMERACCNO	CNAME	ORGANISATIONNAME	PHONENO	EMAILADDRESS	BILLINGADDRESS
1	120100	Julia Calma	(null)	0221238095	julia@apuphoto.co.nz	5 Stanford Street, Albany, Auckland 0632
2	120101	Tay Oh	(null)	0218905666	tay@apuphoto.co.nz	34 Queen Street, Auckland CBD, Auckland 1010
3	120102	Sharmaine Valle	COTTON ON	0204583421	sharm@apuphoto.co.nz	90 Akoranga Drive, Northcote, Auckland 0627
4	120103	Cindy Nguyen	(null)	0218975609	cindy@apuphoto.co.nz	640 Great South Road, Manukau, Auckland 2025
5	120104	Althea Macasaol	PIZZA HUT	0217024538	althea@apuphoto.co.nz	8 Symonds Street, Grafton, Auckland 1010
6	120105	Miles Ocampo	(null)	0227864432	miles@apuphoto.co.nz	80 Saint Lukes Road, Mount Albert, Auckland 1025
7	120106	Elizabeth White	(null)	0209996453	elizabeth@apuphoto.co.nz	7 Queen Street, Auckland CBD, Auckland 1010
8	120107	Amanda Lee	(null)	0226754487	amanda@apuphoto.co.nz	1 Beach Road, Takanini, Auckland 2112

Booking Table:

BookingID	Reason	IsCancelled	DateCancelled	CustomerAccNo
ABC200	Birthday	YES	2023-05-25	120100
ABC201	Corporate Events	NO	NULL	120101
ABC202	Wedding Receptions	YES	2023-05-20	120102
ABC203	Seminars	YES	2023-04-27	120103
ABC204	Food Commercial	NO	NULL	120104
ABC205	Graduation	NO	NULL	120105
ABC206	Baby Shower	NO	NULL	120106
ABC207	Family Reunion	NO	NULL	120107

```
INSERT INTO Booking
  (BookingID, Reason, IsCancelled, DateCancelled, CustomerAccNo)
VALUES
  ('ABC200', 'Birthday', 'YES', TO_DATE('2023-05-25', 'YYYY-MM-DD'), '120100');

INSERT INTO Booking
  (BookingID, Reason, IsCancelled, DateCancelled, CustomerAccNo)
VALUES
  ('ABC201', 'Corporate Events', 'NO', NULL, '120101');

INSERT INTO Booking
  (BookingID, Reason, IsCancelled, DateCancelled, CustomerAccNo)
VALUES
  ('ABC202', 'Wedding Receptions', 'YES', TO_DATE('2023-05-20', 'YYYY-MM-DD'), '120102');

INSERT INTO Booking
  (BookingID, Reason, IsCancelled, DateCancelled, CustomerAccNo)
VALUES
  ('ABC203', 'Seminars', 'YES', TO_DATE('2023-04-27', 'YYYY-MM-DD'), '120103');

INSERT INTO Booking
  (BookingID, Reason, IsCancelled, DateCancelled, CustomerAccNo)
VALUES
  ('ABC204', 'Food Commercial', 'NO', NULL, '120104');

INSERT INTO Booking
  (BookingID, Reason, IsCancelled, DateCancelled, CustomerAccNo)
VALUES
  ('ABC205', 'Graduation', 'NO', NULL, '120105');

INSERT INTO Booking
  (BookingID, Reason, IsCancelled, DateCancelled, CustomerAccNo)
VALUES
  ('ABC206', 'Baby Shower', 'NO', NULL, '120106');

INSERT INTO Booking
  (BookingID, Reason, IsCancelled, DateCancelled, CustomerAccNo)
VALUES
  ('ABC207', 'Family Reunion', 'NO', NULL, '120107');
COMMIT;
```

```
SELECT * FROM BOOKING;
```

Script Output | Query Result | Query Result 1 | Query Result 2 | Query Result 3

SQL | All Rows Fetched: 8 in 0.101 seconds

	BOOKINGID	REASON	ISCANCELLED	DATECANCELLED	CUSTOMERACCNO
1	ABC200	Birthday	YES	25/05/23	120100
2	ABC201	Corporate Events	NO	(null)	120101
3	ABC202	Wedding Receptions	YES	20/05/23	120102
4	ABC203	Seminars	YES	27/04/23	120103
5	ABC204	Food Commercial	NO	(null)	120104
6	ABC205	Graduation	NO	(null)	120105
7	ABC206	Baby Shower	NO	(null)	120106
8	ABC207	Family Reunion	NO	(null)	120107

Orders Table:

OrderNo	ACCEPTANCE_DATE_TIME	TOTALCOST	COLLECTIONMETHOD	CustomerAccNo
090300	2023-06-25	125.00	Courier	120101
090301	2023-07-05	135.00	Courier	120104
090302	2023-07-13	**100.00**	Pickup	120105
090303	2023-07-19	**110.00**	Pickup	120106
090304	2023-09-01	160.00	Courier	120107

```
INSERT INTO Orders
  (OrderNo, Acceptance_Date_Time, TotalCost, CollectionMethod, CustomerAccNo)
VALUES
  ('090300', TO_DATE('2023-06-25', 'YYYY-MM-DD'), '125.00', 'Courier', '120101');

INSERT INTO Orders
  (OrderNo, Acceptance_Date_Time, TotalCost, CollectionMethod, CustomerAccNo)
VALUES
  ('090301', TO_DATE('2023-07-05', 'YYYY-MM-DD'), '135.00', 'Courier', '120104');

INSERT INTO Orders
  (OrderNo, Acceptance_Date_Time, TotalCost, CollectionMethod, CustomerAccNo)
VALUES
  ('090302', TO_DATE('2023-07-13', 'YYYY-MM-DD'), '100.00', 'Pickup', '120105');

INSERT INTO Orders
  (OrderNo, Acceptance_Date_Time, TotalCost, CollectionMethod, CustomerAccNo)
VALUES
  ('090303', TO_DATE('2023-07-19', 'YYYY-MM-DD'), '110.00', 'Pickup', '120106');

INSERT INTO Orders
  (OrderNo, Acceptance_Date_Time, TotalCost, CollectionMethod, CustomerAccNo)
VALUES
  ('090304', TO_DATE('2023-09-01', 'YYYY-MM-DD'), '160.00', 'Courier', '120107');
COMMIT;
```

```
SELECT * FROM ORDERS;
```

Script Output | Query Result | Query Result 1 | Query Result 2 | Query Result 3 | Quer

SQL | All Rows Fetched: 5 in 0.151 seconds

	ORDERNO	ACCEPTANCE_DATE_TIME	TOTALCOST	COLLECTIONMETHOD	CUSTOMERACCNO
1	090300	25/06/23	125	Courier	120101
2	090301	05/07/23	135	Courier	120104
3	090302	13/07/23	100	Pickup	120105
4	090303	19/07/23	110	Pickup	120106
5	090304	01/09/23	160	Courier	120107

DestinationAddress Table:

OrderNo	POSTALADDRESS	POSTALCHARGE
090300	21 Schnapper Rock, Manukau, Auckland 0621	11.00
090301	15 Beach Road, Mission Bay, Auckland 1071	9.00
090302	57 Dominion Road, Mount Eden, Auckland 1024	10.00
090303	31 Queen Street, Parnell, Auckland 1052	8.00
090304	8 Symonds Street, Grafton, Auckland 1010	8.00

```
INSERT INTO DestinationAddress
  (OrderNo, PostalAddress, PostalCharge)
VALUES
  ('090301', '15 Beach Road, Mission Bay, Auckland 1071', '9.00');

 INSERT INTO DestinationAddress
  (OrderNo, PostalAddress, PostalCharge)
VALUES
  ('090302', '157 Dominion Road, Mount Eden, Auckland 1024', '10.00');

 INSERT INTO DestinationAddress
  (OrderNo, PostalAddress, PostalCharge)
VALUES
  ('090303', '131 Queen Street, Parnell, Auckland 1052', '8.00');

 INSERT INTO DestinationAddress
  (OrderNo, PostalAddress, PostalCharge)
VALUES
  ('090304', '8 Symonds Street, Grafton, Auckland 1010', '8.00');

COMMIT;
```

```
SELECT*FROM DestinationAddress;
```

Query Result

SQL | All Rows Fetched: 5 in 0.092 seconds

	ORDERNO	POSTALADDRESS	POSTALCHARGE
1	090300	21 Schnapper Rock, Manukau, Auckland 0621	11
2	090301	15 Beach Road, Mission Bay, Auckland 1071	9
3	090302	157 Dominion Road, Mount Eden, Auckland 1024	10
4	090303	131 Queen Street, Parnell, Auckland 1052	8
5	090304	8 Symonds Street, Grafton, Auckland 1010	8

PhotographySession Table:

SessionID	SESSIONDATE	SESSIONTIME	CHARGEAMOUNT	BOOKINGID
MIL400	2023-05-28	10:00 – 11:15	150.00	ABC201
MIL401	2023-06-15	08:30 – 09:45	150.00	ABC204
MIL402	2023-06-30	10:00 - 11:15	150.00	ABC205
MIL403	2023-07-05	13:30 - 14:45	230.00	ABC206
MIL404	2023-08-18	17:30 – 18:45	230.00	ABC207

```
INSERT INTO PhotographySession
  (SessionID, SessionDate, SessionTime, ChargeAmount, BookingID)
VALUES
  ('MIL400', TO_DATE('2023-05-28', 'YYYY-MM-DD'), '10:00 – 11:15' , '150.00', 'ABC201');

INSERT INTO PhotographySession
  (SessionID, SessionDate, SessionTime, ChargeAmount, BookingID)
VALUES
  ('MIL401', TO_DATE('2023-06-15', 'YYYY-MM-DD'), '08:30 – 09:45' , '150.00', 'ABC204');

INSERT INTO PhotographySession
  (SessionID, SessionDate, SessionTime, ChargeAmount, BookingID)
VALUES
  ('MIL402', TO_DATE('2023-06-30', 'YYYY-MM-DD'), '10:00 – 11:15' , '150.00', 'ABC205');

INSERT INTO PhotographySession
  (SessionID, SessionDate, SessionTime, ChargeAmount, BookingID)
VALUES
  ('MIL403', TO_DATE('2023-07-05', 'YYYY-MM-DD'), '13:30 – 14:45' , '230.00', 'ABC206');

INSERT INTO PhotographySession
  (SessionID, SessionDate, SessionTime, ChargeAmount, BookingID)
VALUES
  ('MIL404', TO_DATE('2023-08-18', 'YYYY-MM-DD'), '17:30 – 18:45' , '230.00', 'ABC207');

COMMIT;
```

```
SELECT * FROM PhotographySession;
```

Script Output | Query Result | Query Result 1 | Query Result 2 | Query

SQL | All Rows Fetched: 5 in 0.063 seconds

	SESSIONID	SESSIONDATE	SESSIONTIME	CHARGEAMOUNT	BOOKINGID
1	MIL400	28/05/23	10:00 – 11:15	150	ABC201
2	MIL401	15/06/23	08:30 – 09:45	150	ABC204
3	MIL402	30/06/23	10:00 – 11:15	150	ABC205
4	MIL403	05/07/23	13:30 – 14:45	230	ABC206
5	MIL404	18/08/23	17:30 – 18:45	230	ABC207

Location Table:

LOCATIONID	SESSIONID	LOCATIONNAME	LSTREETADDRESS	LSUBURB	LCITY	LPOSTCODE
PAR500	MIL400	Park	38 Hillcrest Road	Westmere	Auckland	1061
PAR501	MIL401	APU Studio	9 Shore Road	Remuera	Auckland	1011
PAR502	MIL402	Beach	56 Marine Parade	Milford	Auckland	1071
PAR503	MIL403	APU Studio	9 Shore Road	Remuera	Auckland	1011
PAR504	MIL404	Cafe	73 Manukau Road	Epsom	Auckland	1052

```
INSERT INTO Location
  (LocationID, SessionID, LocationName, LStreetAddress, LSuburb, LCity, LPostcode)
VALUES
  ('PAR500', 'MIL400', 'Park', '38 Hillcrest Road', 'Westmere' , 'Auckland','1061');

INSERT INTO Location
  (LocationID, SessionID, LocationName, LStreetAddress, LSuburb, LCity, LPostcode)
VALUES
  ('PAR501', 'MIL401', 'APU Studio', '9 Shore Road', 'Remuera' , 'Auckland','1011');

INSERT INTO Location
  (LocationID, SessionID, LocationName, LStreetAddress, LSuburb, LCity, LPostcode)
VALUES
  ('PAR502', 'MIL402', 'Beach', '56 Marine Parade', 'Milford' , 'Auckland','1071');

INSERT INTO Location
  (LocationID, SessionID, LocationName, LStreetAddress, LSuburb, LCity, LPostcode)
VALUES
  ('PAR503', 'MIL403', 'APU Studio', '9 Shore Road', 'Remuera' , 'Auckland','1011');

INSERT INTO Location
  (LocationID, SessionID, LocationName, LStreetAddress, LSuburb, LCity, LPostcode)
VALUES
  ('PAR504', 'MIL404', 'Cafe', '73 Manukau Road', 'Epsom' , 'Auckland','1052');

COMMIT;
```

```
SELECT * FROM LOCATION;
```

Script Output | Query Result | Query Result 1 | Query Result 2 | Query Result 3 | Query

SQL | All Rows Fetched: 5 in 0.078 seconds

	LOCATIONID	SESSIONID	LOCATIONNAME	LSTREETADDRESS	LSUBURB	LCITY	LPOSTCODE
1	PAR500	MIL400	Park	38 Hillcrest Road	Westmere	Auckland	1061
2	PAR501	MIL401	APU Studio	9 Shore Road	Remuera	Auckland	1011
3	PAR502	MIL402	Beach	56 Marine Parade	Milford	Auckland	1071
4	PAR503	MIL403	APU Studio	9 Shore Road	Remuera	Auckland	1011
5	PAR504	MIL404	Cafe	73 Manukau Road	Epsom	Auckland	1052

DigitizePhotograph Table:

PHOTONO	SESSIONSTARTTIME	SESSIONID
IMG001	10:00:00	MIL400
IMG002	08:30:00	MIL401
IMG003	10:00:00	MIL402
IMG004	13:30:00	MIL403

IMG005	17:30:00	MIL404

```
INSERT INTO DigitizePhotograph
  (PhotoNo, SessionStartTime, SessionID)
VALUES
  ('IMG001', '10:00:00','MIL400');

INSERT INTO DigitizePhotograph
  (PhotoNo, SessionStartTime, SessionID)
VALUES
  ('IMG002', '08:30:00','MIL401');

INSERT INTO DigitizePhotograph
  (PhotoNo, SessionStartTime, SessionID)
VALUES
  ('IMG003', '10:00:00','MIL402');

INSERT INTO DigitizePhotograph
  (PhotoNo, SessionStartTime, SessionID)
VALUES
  ('IMG004', '13:30:00','MIL403');

 INSERT INTO DigitizePhotograph
  (PhotoNo, SessionStartTime, SessionID)
VALUES
  ('IMG005', '17:30:00','MIL404');
 COMMIT;
```

```
SELECT * FROM DigitizePhotograph;
```

Script Output | Query Result | Query Result 1

SQL | All Rows Fetched: 5 in 0.078 second

	PHOTONO	SESSIONSTARTTIME	SESSIONID
1	IMG001	10:00:00	MIL400
2	IMG002	08:30:00	MIL401
3	IMG003	10:00:00	MIL402
4	IMG004	13:30:00	MIL403
5	IMG005	17:30:00	MIL404

PhotoCombination Table:

COMBID	PHOTOCOLOUR	PHOTOSIZE	FINISHTYPE	PRICE
COM600	B and W	4X6	GLOSSY	2.00
COM601	B and W	6X6	MATTE	6.00
COM602	Colour	4X6	MATTE	17.00

COM603	B and W	7X10	GLOSSY	25.00
COM604	Sepia	4X6	GLOSSY	13.00

```
INSERT INTO PhotoCombination
  (CombID, PhotoColour, PhotoSize, FinishType, Price)
VALUES
  ('COM600', 'B and W', '4X6', 'GLOSSY', '2.00');

INSERT INTO PhotoCombination
  (CombID, PhotoColour, PhotoSize, FinishType, Price)
VALUES
  ('COM601', 'B and W', '6X6', 'MATTE', '6.00');

 INSERT INTO PhotoCombination
  (CombID, PhotoColour, PhotoSize, FinishType, Price)
VALUES
  ('COM602', 'Colour', '4X6', 'GMATTE', '17.00');

 INSERT INTO PhotoCombination
  (CombID, PhotoColour, PhotoSize, FinishType, Price)
VALUES
  ('COM603', 'B and W', '7X10', 'GLOSSY', '25.00');

 INSERT INTO PhotoCombination
  (CombID, PhotoColour, PhotoSize, FinishType, Price)
VALUES
  ('COM604', 'Sepia', '4X6', 'GLOSSY', '13.00');

COMMIT;
```

```
SELECT * FROM PhotoCombination;
```

Script Output | Query Result | Query Result 1 | Query Resu

SQL | All Rows Fetched: 5 in 0.213 seconds

	COMBID	PHOTOCOLOUR	PHOTOSIZE	FINISHTYPE	PRICE
1	COM600	B and W	4X6	GLOSSY	2
2	COM601	B and W	6X6	MATTE	6
3	COM602	Colour	4X6	GMATTE	17
4	COM603	B and W	7X10	GLOSSY	25
5	COM604	Sepia	4X6	GLOSSY	13

OrderedPhotograph Table:

PHOTONO	COMBID	NOOFCOPIES	ORDERNO
IMG001	COM600	2	090300
IMG002	COM601	6	090301

IMG003	COM600	4	090302
IMG004	COM603	7	090303
IMG005	COM604	9	090304

```
INSERT INTO OrderedPhotograph
  (PhotoNo, CombID, NoOfCopies, OrderNo)
VALUES
  ('IMG001', 'COM600', '2', '090300');

INSERT INTO OrderedPhotograph
  (PhotoNo, CombID, NoOfCopies, OrderNo)
VALUES
  ('IMG002', 'COM601', '6', '090301');

INSERT INTO OrderedPhotograph
  (PhotoNo, CombID, NoOfCopies, OrderNo)
VALUES
  ('IMG003', 'COM600', '4', '090302');

INSERT INTO OrderedPhotograph
  (PhotoNo, CombID, NoOfCopies, OrderNo)
VALUES
  ('IMG004', 'COM603', '7', '090303');

INSERT INTO OrderedPhotograph
  (PhotoNo, CombID, NoOfCopies, OrderNo)
VALUES
  ('IMG005', 'COM604', '9', '090304');

  COMMIT;
```

	PHOTONO	COMBID	NOOFCOPIES	ORDERNO
1	IMG001	COM600	2	090300
2	IMG002	COM601	6	090301
3	IMG003	COM600	4	090302
4	IMG004	COM603	7	090303
5	IMG005	COM604	9	090304

TASK C

TASK 5: CONSTRUCT SQL QUERIES

PURPOSE OF THE QUERY 1:

To search for all customers who booked daytime photography sessions.

SQL SELECT query:

```
SELECT c.CName AS "Customer", s.SessionDate AS "Session Date", s.SessionTime AS "Session Time"
FROM Customer c, Booking b, PhotographySession s
WHERE c.CustomerAccNo = b.CustomerAccNo
AND b.BookingID = s.BookingID
AND IsCancelled = 'NO'
AND s.SessionTime BETWEEN '08:30' AND '14:45'
```

Output/Result of the Query:

	Customer	Session Date	Session Time
1	Althea Macasaol	15/06/23	08:30 – 09:45
2	Miles Ocampo	30/06/23	10:00 – 11:15
3	Tay Oh	28/05/23	10:00 – 11:15
4	Elizabeth White	05/07/23	13:30 – 14:45

PURPOSE OF THE QUERY 2:

To count all orders where collection method is by courier (sort by PostalCharge from highest to lowest)

SQL SELECT query:

```
SELECT COUNT(o.OrderNo)AS "COUNT", o.CollectionMethod, o.OrderNo, d.PostalCharge AS "POSTAL CHARGE ($)"
FROM Orders o, DestinationAddress d
WHERE o.OrderNo = d.OrderNo
AND o.CollectionMethod = 'Courier'
GROUP BY o.CollectionMethod, o.OrderNo, d.PostalCharge
ORDER BY d.PostalCharge DESC;
```

Output/Result of the Query:

	COUNT	COLLECTIONMETHOD	ORDERNO	POSTAL CHARGE ($)
1	1	Courier	090300	11
2	1	Courier	090301	9
3	1	Courier	090304	8

PURPOSE OF THE QUERY 3:

To search for organisations customers.

SQL SELECT query:

```
SELECT CName AS "Customer",organisationName "Organisation"
FROM CUSTOMER
WHERE organisationName IS NOT NULL;
```

Output/Result of the Query:

	Customer	Organisation
1	Sharmaine Valle	COTTON ON
2	Althea Macasaol	PIZZA HUT

PURPOSE OF THE QUERY 4:

To find out the average sale of each month (display by highest average to lowest average).

SQL SELECT query:

```
SELECT TO_CHAR(SessionDate, 'Month') AS Month, AVG(ChargeAmount) AS "Average Charge"
FROM PhotographySession
GROUP BY TO_CHAR(SessionDate, 'Month')
ORDER BY "Average Charge" DESC;
```

Output/Result of the Query:

	MONTH	Average Charge
1	August	230
2	July	230
3	May	150
4	June	150

PURPOSE OF THE QUERY 5:

To search for the most used location in photography sessions.

SQL SELECT query:

```
SELECT l.LocationName AS "Location", COUNT(*) AS "Session Count"
FROM PhotographySession s
JOIN Location l ON s.SessionID = l.SessionID
GROUP BY l.LocationName
ORDER BY "Session Count" DESC
FETCH FIRST 1 ROWS ONLY;
```

Output/Result of the Query:

	Location	Session Count
1	APU Studio	2