

Aron McDonald

Student Number: ST10291606

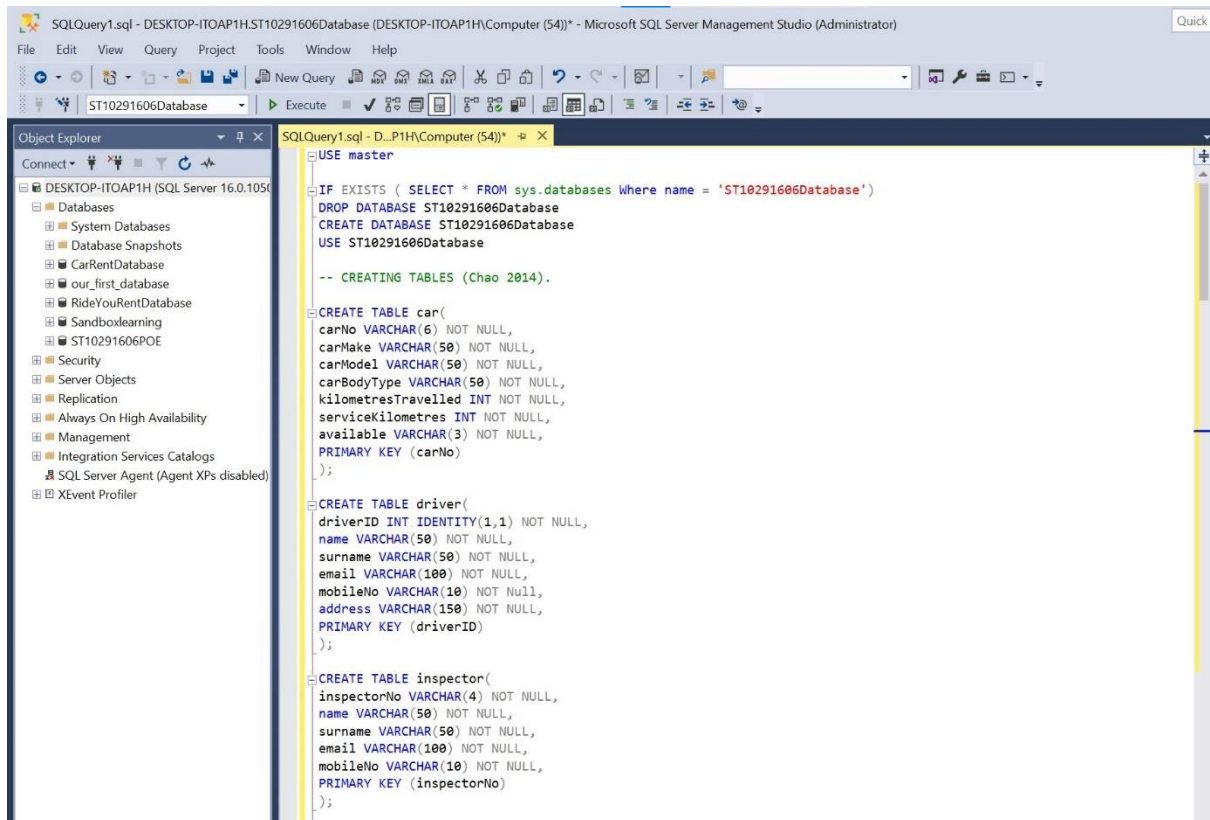
PATHWAY: CLDV6211

Lecturer: Wania Farshida

04 May 2023

POE Task 2

QUESTION 1 + 2



The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left displays the server structure, including the 'ST10291606Database' under the 'DESKTOP-ITOAP1H' server. The main query window, titled 'SQLQuery1.sql - D:\P1H(Computer (54))', contains the following SQL script:

```
USE master

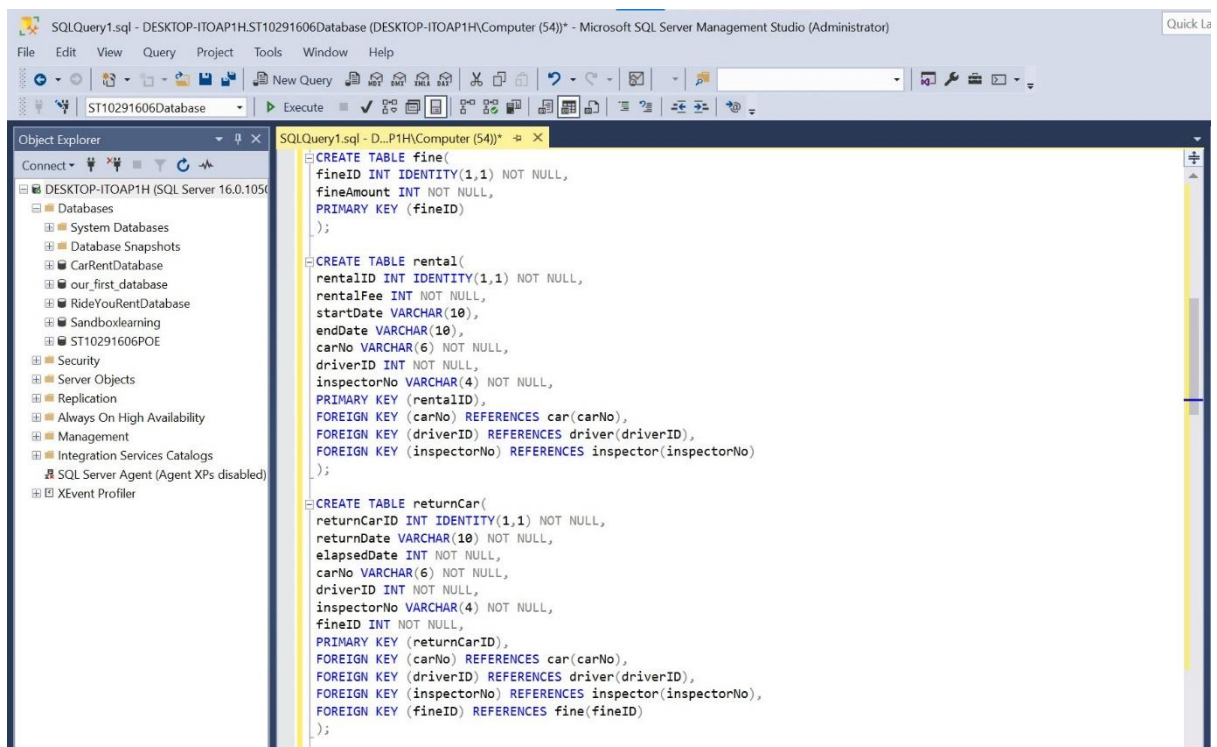
IF EXISTS ( SELECT * FROM sys.databases Where name = 'ST10291606Database')
DROP DATABASE ST10291606Database
CREATE DATABASE ST10291606Database
USE ST10291606Database

-- CREATING TABLES (Chao 2014).

CREATE TABLE car(
carNo VARCHAR(6) NOT NULL,
carMake VARCHAR(50) NOT NULL,
carModel VARCHAR(50) NOT NULL,
carBodyType VARCHAR(50) NOT NULL,
kilometresTravelled INT NOT NULL,
serviceKilometres INT NOT NULL,
available VARCHAR(3) NOT NULL,
PRIMARY KEY (carNo)
);

CREATE TABLE driver(
driverID INT IDENTITY(1,1) NOT NULL,
name VARCHAR(50) NOT NULL,
surname VARCHAR(50) NOT NULL,
email VARCHAR(100) NOT NULL,
mobileNo VARCHAR(10) NOT NULL,
address VARCHAR(150) NOT NULL,
PRIMARY KEY (driverID)
);

CREATE TABLE inspector(
inspectorNo VARCHAR(4) NOT NULL,
name VARCHAR(50) NOT NULL,
surname VARCHAR(50) NOT NULL,
email VARCHAR(100) NOT NULL,
mobileNo VARCHAR(10) NOT NULL,
PRIMARY KEY (inspectorNo)
);
```



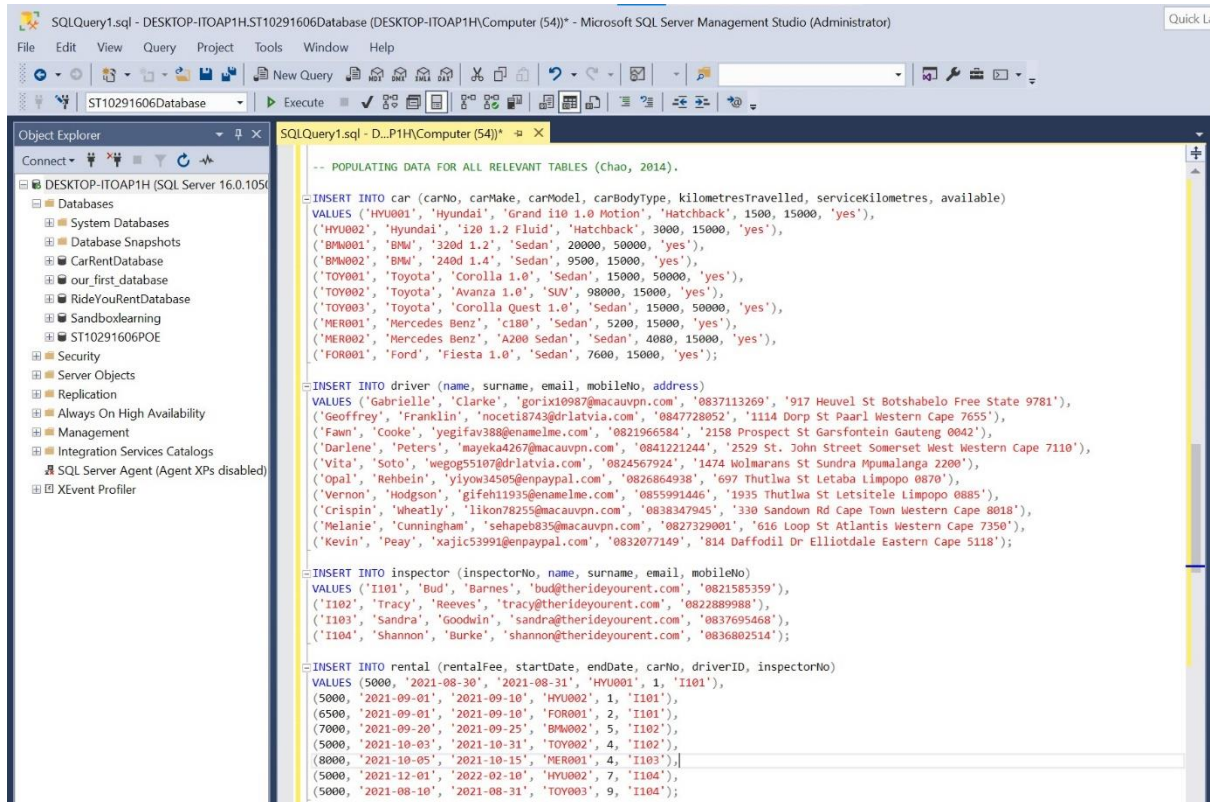
The screenshot shows the continuation of the SQL script in the same Microsoft SQL Server Management Studio interface. The query window contains the following SQL script:

```
CREATE TABLE fine(
fineID INT IDENTITY(1,1) NOT NULL,
fineAmount INT NOT NULL,
PRIMARY KEY (fineID)
);

CREATE TABLE rental(
rentalID INT IDENTITY(1,1) NOT NULL,
rentalFee INT NOT NULL,
startDate VARCHAR(10),
endDate VARCHAR(10),
carNo VARCHAR(6) NOT NULL,
driverID INT NOT NULL,
inspectorNo VARCHAR(4) NOT NULL,
PRIMARY KEY (rentalID),
FOREIGN KEY (carNo) REFERENCES car(carNo),
FOREIGN KEY (driverID) REFERENCES driver(driverID),
FOREIGN KEY (inspectorNo) REFERENCES inspector(inspectorNo)
);

CREATE TABLE returnCar(
returnCarID INT IDENTITY(1,1) NOT NULL,
returnDate VARCHAR(10) NOT NULL,
elapsedDate INT NOT NULL,
carNo VARCHAR(6) NOT NULL,
driverID INT NOT NULL,
inspectorNo VARCHAR(4) NOT NULL,
fineID INT NOT NULL,
PRIMARY KEY (returnCarID),
FOREIGN KEY (carNo) REFERENCES car(carNo),
FOREIGN KEY (driverID) REFERENCES driver(driverID),
FOREIGN KEY (inspectorNo) REFERENCES inspector(inspectorNo),
FOREIGN KEY (fineID) REFERENCES fine(fineID)
);
```

QUESTION 3 + 4



SQLQuery1.sql - DESKTOP-ITOAP1H.ST10291606Database (DESKTOP-ITOAP1H\Computer (54)) - Microsoft SQL Server Management Studio (Administrator)

Object Explorer: DESKTOP-ITOAP1H (SQL Server 16.0.1050)

- Databases
 - System Databases
 - Database Snapshots
 - CarRentDatabase
 - our_first_database
 - RideYouRentDatabase
 - Sandboxlearning
 - ST10291606POE
- Security
- Server Objects
- Replication
- Always On High Availability
- Management
- Integration Services Catalogs
- SQL Server Agent (Agent XPs disabled)
- XEvent Profiler

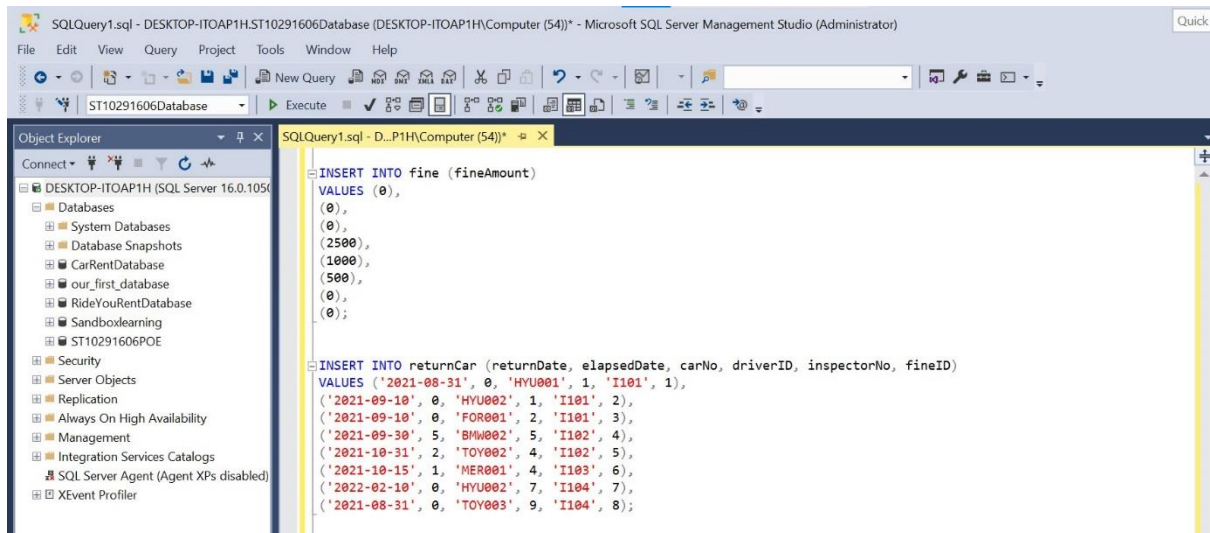
```
-- POPULATING DATA FOR ALL RELEVANT TABLES (Chao, 2014).

INSERT INTO car (carNo, carMake, carModel, carBodyType, kilometresTravelled, serviceKilometres, available)
VALUES ('HYU001', 'Hyundai', 'Grand i10 1.0 Motion', 'Hatchback', 1500, 15000, 'yes'),
('HYU002', 'Hyundai', 'i20 1.2 Fluid', 'Hatchback', 3000, 15000, 'yes'),
('BMW001', 'BMW', '320d 1.2', 'Sedan', 20000, 50000, 'yes'),
('BMW002', 'BMW', '240d 1.4', 'Sedan', 9500, 15000, 'yes'),
('TOY001', 'Toyota', 'Corolla 1.0', 'Sedan', 15000, 50000, 'yes'),
('TOY002', 'Toyota', 'Avanza 1.0', 'SUV', 98000, 15000, 'yes'),
('TOY003', 'Toyota', 'Corolla Quest 1.0', 'Sedan', 15000, 50000, 'yes'),
('MER001', 'Mercedes Benz', 'c180', 'Sedan', 5200, 15000, 'yes'),
('MER002', 'Mercedes Benz', 'A200 Sedan', 'Sedan', 4000, 15000, 'yes'),
('FOR001', 'Ford', 'Fiesta 1.0', 'Sedan', 7600, 15000, 'yes');

INSERT INTO driver (name, surname, email, mobileNo, address)
VALUES ('Gabrielle', 'Clarke', 'gorix19907@macauvnp.com', '0837113269', '917 Heuvel St Botshabelo Free State 9781'),
('Geoffrey', 'Franklin', 'noceti18743@drlatvia.com', '0847728052', '1114 Dorp St Paarl Western Cape 7655'),
('Fawn', 'Cooke', 'yegifav388@enamelme.com', '0821966584', '2158 Prospect St Garsfontein Gauteng 0042'),
('Darlene', 'Peters', 'mayeka4267@macauvnp.com', '0841221244', '2529 St. John Street Somerset West Western Cape 7110'),
('Vita', 'Soto', 'wegogs5107@drlatvia.com', '0824567924', '1474 Wolmarans St Sunda Mpumalanga 2200'),
('Opal', 'Rehbein', 'yiyow34505@enpaypal.com', '0826864938', '697 Thutlwa St Letaba Limpopo 0870'),
('Vernon', 'Hodgson', 'gifehi1935@enamelme.com', '0855991446', '1935 Thutlwa St Letsitele Limpopo 0885'),
('Crispin', 'Wheatly', 'likon78255@macauvnp.com', '0838347945', '330 Sandown Rd Cape Town Western Cape 8018'),
('Melanie', 'Cunningham', 'sehaped835@macauvnp.com', '0827329001', '616 Loop St Atlantis Western Cape 7350'),
('Kevin', 'Peay', 'xajic53991@enpaypal.com', '0832077149', '814 Daffodil Dr Elliotdale Eastern Cape 5118');

INSERT INTO inspector (inspectorNo, name, surname, email, mobileNo)
VALUES ('I101', 'Bud', 'Barnes', 'budd@therideyourent.com', '0821585359'),
('I102', 'Tracy', 'Reeves', 'tracy@therideyourent.com', '0822889988'),
('I103', 'Sandra', 'Goodwin', 'sandra@therideyourent.com', '0837695468'),
('I104', 'Shannon', 'Burke', 'shannon@therideyourent.com', '0836802514');

INSERT INTO rental (rentalFee, startDate, endDate, carNo, driverID, inspectorNo)
VALUES (5000, '2021-08-30', '2021-08-31', 'HYU001', 1, 'I101'),
(5000, '2021-09-01', '2021-09-10', 'HYU002', 1, 'I101'),
(6500, '2021-09-01', '2021-09-10', 'FOR001', 2, 'I101'),
(7000, '2021-09-20', '2021-09-25', 'BMW002', 5, 'I102'),
(5000, '2021-10-03', '2021-10-31', 'TOY002', 4, 'I102'),
(8000, '2021-10-05', '2021-10-15', 'MER001', 4, 'I103'),
(5000, '2021-12-01', '2022-02-10', 'HYU002', 7, 'I104'),
(5000, '2021-08-10', '2021-08-31', 'TOY003', 9, 'I104');
```



SQLQuery1.sql - DESKTOP-ITOAP1H.ST10291606Database (DESKTOP-ITOAP1H\Computer (54)) - Microsoft SQL Server Management Studio (Administrator)

Object Explorer: DESKTOP-ITOAP1H (SQL Server 16.0.1050)

- Databases
 - System Databases
 - Database Snapshots
 - CarRentDatabase
 - our_first_database
 - RideYouRentDatabase
 - Sandboxlearning
 - ST10291606POE
- Security
- Server Objects
- Replication
- Always On High Availability
- Management
- Integration Services Catalogs
- SQL Server Agent (Agent XPs disabled)
- XEvent Profiler

```
INSERT INTO fine (fineAmount)
VALUES (0),
(0),
(2500),
(1000),
(500),
(0),
(0);

INSERT INTO returnCar (returnDate, elapsedDate, carNo, driverID, inspectorNo, fineID)
VALUES ('2021-08-31', 0, 'HYU001', 1, 'I101', 1),
('2021-09-10', 0, 'HYU002', 1, 'I101', 2),
('2021-09-10', 0, 'FOR001', 2, 'I101', 3),
('2021-09-30', 5, 'BMW002', 5, 'I102', 4),
('2021-10-31', 2, 'TOY002', 4, 'I102', 5),
('2021-10-15', 1, 'MER001', 4, 'I103', 6),
('2022-02-10', 0, 'HYU002', 7, 'I104', 7),
('2021-08-31', 0, 'TOY003', 9, 'I104', 8);
```

QUESTION 3 + 4 RESULTS

SQLQuery1.sql - DESKTOP-ITOAP1H.ST10291606Database (DESKTOP-ITOAP1H\Computer (54)) - Microsoft SQL Server Management Studio (Administrator)

File Edit View Query Project Tools Window Help

ST10291606Database Execute

Object Explorer

Connect

DESKTOP-ITOAP1H (SQL Server 16.0.1050)

Databases

- System Databases
- Database Snapshots
- CarRentDatabase
- our_first_database
- RideYouRentDatabase
- SandboxLearning
- ST10291606POE
- Security
- Server Objects
- Replication
- Always On High Availability
- Management
- Integration Services Catalogs
- SQL Server Agent (Agent XPs disabled)
- XEvent Profiler

SQLQuery1.sql - D:\P1H\Computer (54) *

70 %

Results Messages

carNo	carMake	carModel	carBodyType	kilometresTravelled	serviceKilometres	available
3	FOR001	Ford	Fiesta 1.0	Sedan	7600	15000
4	HYU001	Hyundai	Grand i10 1.0 Motion	Hatchback	1500	15000
5	HYU002	Hyundai	i20 1.2 Fluid	Hatchback	3000	15000
6	MER001	Mercedes Benz	c180	Sedan	5200	15000
7	MER002	Mercedes Benz	A200 Sedan	Sedan	4080	15000
8	TOY001	Toyota	Corolla 1.0	Sedan	15000	50000
9	TOY002	Toyota	Avanza 1.0	SUV	98000	15000
10	TOY003	Toyota	Corolla Quest 1.0	Sedan	15000	50000

driverID	name	surname	email	mobileNo	address
1	Gabrielle	Clarke	gonx10967@macaupn.com	0837113269	917 Heuvel St Botshabelo Free State 9781
2	Geoffrey	Franklin	nocel8743@driahia.com	0847728052	1114 Dorp St Paarl Western Cape 7655
3	Fawn	Cooke	yegfrav388@enamelme.com	0821966584	2158 Prospect St Garfontein Gauteng 0042
4	Darlene	Peters	mayeka4267@macaupn.com	0841221244	2529 St. John Street Somerset West Western Cape ...
5	Vita	Soto	wegog55107@driahia.com	0824567924	1474 Wolmarans St Sunda Mpumalanga 2200
6	Opal	Rehbein	yicow34505@enpaypal.com	0826864938	697 Thutwa St Letaba Limpopo 0870
7	Vernon	Hodgson	grfeh11935@enamelme.com	0855991446	1935 Thutwa St Letsele Limpopo 0885
8	Crispin	Wheatly	likon78255@macaupn.com	0838347945	330 Sandown Rd Cape Town Western Cape 8018

inspectorNo	name	surname	email	mobileNo
1	Bud	Barnes	bud@therideyourent.com	0821585359
2	Tracy	Reeves	tracy@therideyourent.com	0822889988
3	Sandra	Goodwin	sandra@therideyourent.com	0837695468
4	Shannon	Burke	shannon@therideyourent.com	0836802514

fineID	fineAmount
1	0
2	0
3	0
4	2500
5	1000
6	500
7	0
8	0

rentalID	rentalFee	startDate	endDate	carNo	driverID	inspectorNo
1	5000	2021-08-30	2021-08-31	HYU001	1	I101
2	5000	2021-09-01	2021-09-10	HYU002	1	I101
3	6500	2021-09-01	2021-09-10	FOR001	2	I101
4	7000	2021-09-20	2021-09-25	BMW001	5	I102
5	5000	2021-10-03	2021-10-31	TOY002	4	I102
6	8000	2021-10-05	2021-10-15	MER001	4	I103
7	5000	2021-12-01	2022-02-10	HYU002	7	I104
8	5000	2021-08-10	2021-08-31	TOY003	9	I104

returnCarID	returnDate	elapsedDate	carNo	driverID	inspectorNo	fineID
1	2021-08-31	0	HYU001	1	I101	1
2	2021-09-10	0	HYU002	1	I101	2
3	2021-09-10	0	FOR001	2	I101	3
4	2021-09-30	5	BMW001	5	I102	4
5	2021-10-31	2	TOY002	4	I102	5
6	2021-10-15	1	MER001	4	I103	6
7	2022-02-10	0	HYU002	7	I104	7
8	2021-08-31	0	TOY003	9	I104	8

Query executed successfully. DESKTOP-ITOAP1H (16.0 RTM) DESKTOP-ITOAP1H\Comput... ST10291606Database 00:00:00 48 rows

QUESTION 5

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The title bar indicates the connection is to 'ST10291606Database (DESKTOP-ITOAP1H\Computer (54)) - Microsoft SQL Server Management Studio (Administrator)'. The 'Object Explorer' on the left shows the database structure, including 'Databases' and 'Server Objects'. The 'SQLQuery1.sql' editor in the center contains the following SQL code:

```
INSERT INTO rental (rentalID, rentalFee, startDate, endDate, carNo, driverID, inspectorNo)
VALUES
('2021-09-30', 5, 'BMW002', 5, 'I102', 4),
('2021-10-31', 2, 'TOY002', 4, 'I102', 5),
('2021-10-15', 1, 'MER001', 4, 'I103', 6),
('2022-02-10', 0, 'HYU002', 7, 'I104', 7),
('2021-08-31', 0, 'TOY003', 9, 'I104', 8);

-- ANSWER TO Q.5
SELECT * FROM rental
WHERE startDate BETWEEN '2021-08-01' AND '2021-10-30';

-- ANSWER TO Q.6
SELECT * FROM rental
WHERE inspectorNo = 'I101';

-- ANSWER TO Q.7
SELECT * FROM returnCar
WHERE carNo = 'TOY001' or carNo = 'TOY002' or carNo = 'TOY003';

-- ANSWER TO Q.8
SELECT COUNT(*) FROM rental
WHERE carNo = 'HYU001' or carNo = 'HYU002';

-- ANSWER TO Q.9
UPDATE car
SET carModel = 'Focus'
WHERE carNo = 'FOR001';

-- ANSWER TO Q.10
```

The 'Results' pane at the bottom displays the output of the first query, showing 7 rows of rental data:

	rentalID	rentalFee	startDate	endDate	carNo	driverID	inspectorNo
1	1	5000	2021-08-30	2021-08-31	HYU001	1	I101
2	2	5000	2021-09-01	2021-09-10	HYU002	1	I101
3	3	6500	2021-09-01	2021-09-10	FOR001	2	I101
4	4	7000	2021-09-20	2021-09-25	BMW002	5	I102
5	5	5000	2021-10-03	2021-10-31	TOY002	4	I102
6	6	8000	2021-10-05	2021-10-15	MER001	4	I103
7	8	5000	2021-08-10	2021-08-31	TOY003	9	I104

The status bar at the bottom indicates 'Query executed successfully.' and 'DESKTOP-ITOAP1H (16.0 RTM) DESKTOP-ITOAP1H\Comput... ST10291606Database 00:00:00 7 rows'.

QUESTION 6

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The title bar indicates the connection is to 'DESKTOP-ITOAP1H (SQL Server 16.0.10506.1)'. The Object Explorer on the left shows the database structure, including 'ST10291606Database'. The main query window displays a SQL script with several queries and their results.

SQL Query 1.sql - D:\P1H\Computer (54)*

```
-- ANSWER TO Q.5
SELECT * FROM rental
WHERE startDate BETWEEN '2021-08-01' AND '2021-10-30';

-- ANSWER TO Q.6
SELECT * FROM rental
WHERE inspectorNo = 'I101';

-- ANSWER TO Q.7
SELECT * FROM returnCar
WHERE carNo = 'TOY001' or carNo = 'TOY002' or carNo = 'TOY003';

-- ANSWER TO Q.8
SELECT COUNT(*) FROM rental
WHERE carNo = 'HYU001' or carNo = 'HYU002';

-- ANSWER TO Q.9
UPDATE car
SET carModel = 'Focus'
WHERE carNo = 'FOR001';
```

Results

rentalID	rentalFee	startDate	endDate	carNo	driverID	inspectorNo
1	5000	2021-08-30	2021-09-31	HYU001	1	I101
2	5000	2021-09-01	2021-09-10	HYU002	1	I101
3	6500	2021-09-01	2021-09-10	FOR001	2	I101

Query executed successfully. DESKTOP-ITOAP1H (16.0 RTM) DESKTOP-ITOAP1H\Comput... ST10291606Database 00:00:00 3 rows

QUESTION 7

The screenshot displays the Microsoft SQL Server Management Studio (SSMS) interface. The title bar indicates the connection to 'DESKTOP-ITOAP1H\Computer (54)' and the database 'ST10291606Database'. The Object Explorer on the left shows the database structure, including 'Databases' and 'Security'. The main query window contains the following SQL script:

```
-- ANSWER TO Q.5
SELECT * FROM rental
WHERE startDate BETWEEN '2021-08-01' AND '2021-10-30';

-- ANSWER TO Q.6
SELECT * FROM rental
WHERE inspectorNo = 'I101';

-- ANSWER TO Q.7
SELECT * FROM returnCar
WHERE carNo = 'TOY001' OR carNo = 'TOY002' OR carNo = 'TOY003';

-- ANSWER TO Q.8
SELECT COUNT(*) FROM rental
WHERE carNo = 'HYU001' OR carNo = 'HYU002';

-- ANSWER TO Q.9
UPDATE car
SET carModel = 'Focus'
WHERE carNo = 'FOR001';

--ANSWER TO Q.10
SELECT rental.carNo, car.available, driver.name, driver.surname, rental.startDate, rental.endDate, rental.rentalFee
FROM rental
JOIN driver ON rental.driverID = driver.driverID
JOIN car ON car.carNo = rental.carNo
```

The Results pane at the bottom shows the output of the last query (Q.10), displaying 2 rows of data:

	returnCarID	returnDate	elapsedDate	carNo	driverID	inspectorNo	fineID
1	5	2021-10-31	2	TOY002	4	I102	5
2	8	2021-08-31	0	TOY003	9	I104	8

The status bar at the bottom indicates 'Query executed successfully.' and 'DESKTOP-ITOAP1H (16.0 RTM) DESKTOP-ITOAP1H\Comput... ST10291606Database 00:00:00 2 rows'.

QUESTION 8

The screenshot shows the Microsoft SQL Server Enterprise Manager interface. The left pane displays the Object Explorer with the following structure:

- DESKTOP-ITOAP1H (SQL Server 16.0.1050)
- Databases
 - System Databases
 - Database Snapshots
 - CarRentDatabase
 - our_first_database
 - RideYouRentDatabase
 - Sandboxlearning
 - ST10291606POE
- Security
- Server Objects
- Replication
- Always On High Availability
- Management
- Integration Services Catalogs
- SQL Server Agent (Agent XPs disabled)
- XE Event Profiler

The right pane shows the SQL Query window with the following code:

```
-- ANSWER TO Q.5
SELECT * FROM rental
WHERE startDate BETWEEN '2021-08-01' AND '2021-10-30';

-- ANSWER TO Q.6
SELECT * FROM rental
WHERE inspectorNo = 'I101';

-- ANSWER TO Q.7
SELECT * FROM returnCar
WHERE carNo = 'TOY001' or carNo = 'TOY002' or carNo = 'TOY003';

-- ANSWER TO Q.8
SELECT COUNT(*) FROM rental
WHERE carNo = 'HYU001' or carNo = 'HYU002';

-- ANSWER TO Q.9
UPDATE car
SET carModel = 'Focus'
WHERE carNo = 'FOR001';

-- ANSWER TO Q.10
SELECT rental.carNo, car.available, driver.name, driver.surname, rental.startDate, rental.endDate, rental.rentalFee
FROM rental
JOIN driver on rental.driverID = driver.driverID
JOIN car on car.carNo = rental.carNo
```

The Results pane shows the following data:

	(No column name)
1	3

The status bar at the bottom indicates: Query executed successfully. DESKTOP-ITOAP1H (16.0 RTM) | DESKTOP-ITOAP1H\Comput... ST10291606Database 00:00:00 1 rows

QUESTION 9

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The title bar indicates the connection is to 'DESKTOP-ITOAP1H\Computer (54)' in the 'ST10291606Database'. The Object Explorer on the left shows the database structure, including 'Databases', 'System Databases', 'Database Snapshots', 'CarRentDatabase', 'our_first_database', 'RideYouRentDatabase', 'SandboxLearning', 'ST10291606POE', 'Security', 'Server Objects', 'Replication', 'Always On High Availability', 'Management', 'Integration Services Catalogs', 'SQL Server Agent (Agent XPs disabled)', and 'XEvent Profiler'.

The SQL Query window displays the following queries:

```
-- ANSWER TO Q.5
SELECT * FROM rental
WHERE startDate BETWEEN '2021-08-01' AND '2021-10-30';

-- ANSWER TO Q.6
SELECT * FROM rental
WHERE inspectorNo = 'I101';

-- ANSWER TO Q.7
SELECT * FROM returnCar
WHERE carNo = 'TOY001' or carNo = 'TOY002' or carNo = 'TOY003';

-- ANSWER TO Q.8
SELECT COUNT(*) FROM rental
WHERE carNo = 'HYU001' or carNo = 'HYU002';

-- ANSWER TO Q.9
UPDATE car
SET carModel = 'Focus'
WHERE carNo = 'FOR001';
SELECT * FROM car;

--ANSWER TO Q.10
SELECT rental.carNo, car.available, driver.name, driver.surname, rental.startDate, rental.endDate, rental.rentalFee
FROM rental
JOIN driver on rental.driverID = driver.driverID
JOIN car on car.carNo = rental.carNo
```

The Results window shows the output of the last query (Q.10), displaying 10 rows of data. The columns are: carNo, carMake, carModel, carBodyType, kilometresTravolled, serviceKilometres, and available.

carNo	carMake	carModel	carBodyType	kilometresTravolled	serviceKilometres	available	
1	BMW001	BMW	320d 1.2	Sedan	20000	50000	yes
2	BMW002	BMW	240d 1.4	Sedan	9500	15000	yes
3	FOR001	Ford	Focus	Sedan	7600	15000	yes
4	HYU001	Hyundai	Grand i10 1.0 Motion	Hatchback	1500	15000	yes
5	HYU002	Hyundai	i20 1.2 Fluid	Hatchback	3000	15000	yes
6	MER001	Mercedes Benz	c180	Sedan	5200	15000	yes
7	MER002	Mercedes Benz	A200 Sedan	Sedan	4080	15000	yes
8	TOY001	Toyota	Corolla 1.0	Sedan	15000	50000	yes
9	TOY002	Toyota	Avanza 1.0	SUV	98000	15000	yes
10	TOY003	Toyota	Corolla Quest 1.0	Sedan	15000	50000	yes

The status bar at the bottom indicates: 'Query executed successfully. DESKTOP-ITOAP1H (16.0 RTM) DESKTOP-ITOAP1H\Comput... ST10291606Database 00:00:00 10 rows'.

QUESTION 10

SQLQuery1.sql - DESKTOP-ITOAP1H.ST10291606Database (DESKTOP-ITOAP1H\Computer (54)) - Microsoft SQL Server Management Studio (Administrator)

Object Explorer: DESKTOP-ITOAP1H (SQL Server 16.0.10506.1) > Databases > ST10291606Database

SQLQuery1.sql - D:\P1H\Computer (54)*

```
-- ANSWER TO Q.7
SELECT * FROM returnCar
WHERE carNo = 'TOY001' or carNo = 'TOY002' or carNo = 'TOY003';

-- ANSWER TO Q.8
SELECT COUNT(*) FROM rental
WHERE carNo = 'HYU001' or carNo = 'HYU002';

-- ANSWER TO Q.9
UPDATE car
SET carModel = 'Focus'
WHERE carNo = 'FOR001';
SELECT * FROM car;

--ANSWER TO Q.10
SELECT rental.carNo, car.available, driver.name, driver.surname, rental.startDate, rental.endDate, rental.rentalFee
FROM rental
JOIN driver on rental.driverID = driver.driverID
JOIN car on car.carNo = rental.carNo;

-- ANSWER TO Q.11
SELECT DISTINCT carMake
FROM car
WHERE available = 'yes';

-- ANSWER TO Q.12
SELECT *
```

Results

	carNo	available	name	surname	startDate	endDate	rentalFee
1	HYU001	yes	Gabrielle	Clarke	2021-08-30	2021-08-31	5000
2	HYU002	yes	Gabrielle	Clarke	2021-09-01	2021-09-10	5000
3	FOR001	yes	Geoffrey	Franklin	2021-09-01	2021-09-10	6500
4	BMW002	yes	Vita	Solo	2021-09-20	2021-09-25	7000
5	TOY002	yes	Darlene	Peters	2021-10-03	2021-10-31	5000
6	MER001	yes	Darlene	Peters	2021-10-05	2021-10-15	8000
7	HYU002	yes	Vernon	Hodgson	2021-12-01	2022-02-10	5000
8	TOY003	yes	Melanie	Cunningham	2021-08-10	2021-08-31	5000

Query executed successfully. DESKTOP-ITOAP1H (16.0 RTM) DESKTOP-ITOAP1H\Comput... ST10291606Database 00:00:00 8 rows

QUESTION 11

The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar indicates the connection is to 'DESKTOP-ITOAP1H\Computer (54)' in the 'ST10291606Database'. The Object Explorer on the left shows the database structure, including 'CarRentDatabase' and 'our_first_database'. The main query window contains the following SQL script:

```
UPDATE car
SET carModel = 'Focus'
WHERE carNo = 'FOR001';
SELECT * FROM car

--ANSWER TO Q.10
SELECT rental.carNo, car.available, driver.name, driver.surname, rental.startDate, rental.endDate, rental.rentalFee
FROM rental
JOIN driver on rental.driverID = driver.driverID
JOIN car on car.carNo = rental.carNo

-- ANSWER TO Q.11
SELECT DISTINCT carMake
FROM car
WHERE available = 'yes';

-- ANSWER TO Q.12
SELECT *
FROM car
WHERE serviceKilometres - kilometresTravelled <= 9000;

--ANSWER TO Q.13
SELECT returnCarID,
returnDate,
elapsedDate,
carNo,
driverID
```

The Results pane at the bottom shows the output of the query, displaying 5 rows of car makes:

carMake
1 BMW
2 Ford
3 Hyundai
4 Mercedes Benz
5 Toyota

The status bar at the bottom indicates 'Query executed successfully.' and '5 rows'.

QUESTION 12

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The title bar indicates the current file is 'SQLQuery1.sql' and the server is 'DESKTOP-ITOAP1H\ST10291606Database (DESKTOP-ITOAP1H\Computer (54))'.

The Object Explorer on the left shows the server structure, including databases like 'ST10291606POE'.

The main query editor contains the following SQL code:

```
-- ANSWER TO Q.11
SELECT DISTINCT carMake
FROM car
WHERE available = 'yes';

-- ANSWER TO Q.12
SELECT *
FROM car
WHERE serviceKilometres - kilometresTravelled <= 9000;

--ANSWER TO Q.13
SELECT returnCarID,
returnDate,
elapsedDate,
carNo,
driverID,
inspectorNo,
fineID,
SUM(elapsedDate * 500) AS late_fee
FROM returnCar
GROUP BY returnCarID, returnDate, elapsedDate, carNo, driverID, inspectorNo, fineID
```

The Results pane at the bottom shows the output of the first query, displaying 3 rows of data:

	carNo	carMake	carModel	carBodyType	kilometresTravelled	serviceKilometres	available
1	BMW002	BMW	240d 1.4	Sedan	9500	15000	yes
2	FOR001	Ford	Focus	Sedan	7600	15000	yes
3	TOY002	Toyota	Avanza 1.0	SUV	98000	15000	yes

The status bar at the bottom indicates 'Query executed successfully.' and shows the server name 'DESKTOP-ITOAP1H (16.0 RTM)', the database 'ST10291606Database', and the execution time '00:00:00' with '3 rows' returned.

QUESTION 13

The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar indicates the connection is to 'DESKTOP-ITOAP1H\Computer (54)' in the 'ST10291606Database'.

Object Explorer: The left pane shows the database structure. Under 'Databases', the following are listed: System Databases, Database Snapshots, CarRentDatabase, our_first_database, RideYouRentDatabase, SandboxLearning, and ST10291606POE.

SQL Query Editor: The main pane contains the following SQL script:

```
-- ANSWER TO Q.11
SELECT DISTINCT carMake
FROM car
WHERE available = 'yes';

-- ANSWER TO Q.12
SELECT
FROM car
WHERE serviceKilometres - kilometresTravelled <= 9000;

--ANSWER TO Q.13
SELECT returnCarID,
returnDate,
elapsedDate,
carNo,
driverID,
inspectorNo,
fineID,
SUM(elapsedDate * 500) AS late_fee
FROM returnCar
GROUP BY returnCarID, returnDate, elapsedDate, carNo, driverID, inspectorNo, fineID
```

Results: The bottom pane displays the results of the third query. It shows 8 rows of data with the following columns: returnCarID, returnDate, elapsedDate, carNo, driverID, inspectorNo, fineID, and late_fee.

	returnCarID	returnDate	elapsedDate	carNo	driverID	inspectorNo	fineID	late_fee
1	1	2021-08-31	0	HYU001	1	I101	1	0
2	2	2021-09-10	0	HYU002	1	I101	2	0
3	3	2021-09-10	0	FOR001	2	I101	3	0
4	4	2021-09-30	5	BMW002	5	I102	4	2500
5	5	2021-10-31	2	TOY002	4	I102	5	1000
6	6	2021-10-15	1	MER001	4	I103	6	500
7	7	2022-02-10	0	HYU002	7	I104	7	0
8	8	2021-08-31	0	TOY003	9	I104	8	0

Status Bar: The bottom status bar shows 'Query executed successfully.' and 'DESKTOP-ITOAP1H (16.0 RTM) | DESKTOP-ITOAP1H\Comput... | ST10291606Database | 00:00:00 | 8 rows'.

PROOF OF AZURE MIGRATION

SQLQuery1 (ST10291606 database).sql - st10291606poc.database.windows.net (SQL Server 12.0.2000.8 - adminst10291606) - Microsoft SQL Server Management Studio (Administrator)

Object Explorer: st10291606poc.database.windows.net (SQL Server 12.0.2000.8 - adminst10291606) > Databases > ST10291606POE_DB > Tables

```
SQLQuery1 (ST10291606 (71))
WHERE available = 'yes';

-- ANSWER TO Q.12
SELECT *
FROM car
WHERE serviceKilometres <= kilometresTravelled;

-- ANSWER TO Q.13
SELECT returnCarID,
returnDate,
elapsedDate,
carNo,
driverID,
inspectorNo,
fineID,
SUM(elapsedDate > 500) AS late_fee
FROM returnCar
GROUP BY returnCarID, returnDate, elapsedDate, carNo, driverID, inspectorNo, fineID
```

Results: Messages

returnCarID	returnDate	elapsedDate	carNo	driverID	inspectorNo	fineID	late_fee
1	2021-08-31	0	HYU001	1	1101	1	0
2	2021-09-10	0	HYU002	1	1101	2	0
3	2021-09-10	0	FOR001	2	1101	3	0
4	2021-09-30	5	BMW002	5	1102	4	2500
5	2021-10-31	2	TOY002	4	1102	5	1000
6	2021-10-15	1	MER001	4	1103	6	500
7	2022-02-10	0	HYU002	7	1104	7	0
8	2021-08-31	0	TOY003	8	1104	8	0

Query executed successfully. st10291606poc.database.wind... adminst10291606 (71) ST10291606POE_DB 00:00:00 8 rows

SQLQuery1 (ST10291606 database).sql - st10291606poc.database.windows.net (SQL Server 12.0.2000.8 - adminst10291606) - Microsoft SQL Server Management Studio (Administrator)

Object Explorer: st10291606poc.database.windows.net (SQL Server 12.0.2000.8 - adminst10291606) > Databases > ST10291606POE_DB > Tables

```
SQLQuery1 (ST10291606 (71))
WHERE available = 'yes';

-- ANSWER TO Q.12
SELECT *
```

Connect to Server

SQL Server

Server type: Database Engine

Server name: st10291606poc.database.windows.net

Authentication: SQL Server Authentication

Login: adminst10291606

Password: [REDACTED]

Remember password: ☐

Results: Messages

carMake
1 BMW
2 Ford
3 Hyundai
4 Mercedes
5 Toyota

carNo	carMake	carModel	carBodyType	kilometresTravelled	serviceKilometres	available
1 BMW002	BMW	240i 1.4	Sweden	8500	15000	yes
2 FOR001	Ford	Focus	Sedan	7600	15000	yes
3 TOY002	Toyota	Avenza	SUV	88000	15000	yes

returnCarID	returnDate	elapsedDate	carNo	driverID	inspectorNo	fineID	late_fee
1	2021-08-31	0	HYU001	1	1101	1	0
2	2021-08-10	0	HYU002	1	1101	2	0
3	2021-09-10	0	FOR001	2	1101	3	0
4	2021-09-30	5	BMW	5	1102	4	2500
5	2021-10-31	2	TOY002	4	1102	5	1000
6	2021-10-15	1	MER001	4	1103	6	500
7	2022-02-10	0	HYU002	7	1104	7	0
8	2021-08-31	0	TOY003	8	1104	8	0

Query executed successfully. st10291606poc.database.wind... adminst10291606 (71) ST10291606POE_DB 00:01:11 47 rows

Microsoft Azure | Search resources, services, and docs (G+/I)

Home > ST10291606POE_DB (st10291606poe/ST10291606POE_DB)

ST10291606POE_DB (st10291606poe/ST10291606POE_DB) | Query editor (preview)

Search

Overview, Activity log, Tags, Diagnose and solve problems, Getting started, Query editor (preview)

Settings: Compute + storage, Connection strings, Properties, Locks, Data management, Replicas, Sync to other databases, Integrations, Azure Synapse Link, Stream analytics (preview), Add Azure Search, Power Platform, Power BI

Showing limited object explorer here. For full capabilities, please click here to open Azure Data Studio.

Tables: dbo.car, dbo.drive, dbo.fine, dbo.inspr, dbo.rentz, dbo.retur, Views, Stored Procs

Query 1

```
--ANSWER TO Q.13
SELECT returnCarID,
returnDate,
elapsedDate,
carNo,
driverID,
inspectorNo,
fineID,
SUM(elapsedDate * 500) AS late_fee
FROM returnCar
GROUP BY returnCarID, returnDate, elapsedDate, carNo, driverID, inspectorNo, fineID
```

Run, Cancel query, Save query, Export data as, Show only Editor

Results, Messages

Search to filter items...

returnDate	elapsedDate	carNo	driverID	inspectorNo	fineID	late_fee
2021-08-31	0	HYU001	1	I101	1	0
2021-09-10	0	HYU002	1	I101	2	0
2021-09-10	0	FOR001	2	I101	3	0
2021-09-30	5	BMW002	5	I102	4	2500
2021-10-31	2	TOY002	4	I102	5	1000
2021-10-15	1	MER001	4	I103	6	500

Query succeeded | 0s

References

Chao, L., 2014. *Cloud Database Development & Management*. Boca Raton, FL: CRC Press.

docs.actian.com, 2023. *docs.actian.com*. [Online]

Available at: https://docs.actian.com/vector/5.1/index.html#page/SQLLang/Money_Data_Type.htm [Accessed 04 May 2023].

essentialsq1.com, 2023. *essentialsq1.com*. [Online]

Available at: <https://www.essentialsq1.com/calculate-running-total-sql/> [Accessed 04 May 2023].

geeksengine.com, 2023. *geeksengine.com*. [Online]

Available at: <http://www.geeksengine.com/database/basic-select/arithmetic-operations.php#:~:text=When%20you%20need%20to%20perform%20calculations%20in%20SQL,tutorial%20and%20practices%20if%20you%20haven%27t%20done%20so.> [Accessed 04 May 2023].

Kadlec, J., 2008. *mssqltips.com*. [Online]

Available at: <https://www.mssqltips.com/sqlservertip/1547/calculating-mathematical-values-in-sql-server/> [Accessed 04 May 2023].

w3schools.com, 2023. *w3schools.com*. [Online]

Available at: https://www.w3schools.com/sql/sql_ref_keywords.asp [Accessed 04 May 2023].

