Part 1 - Algebra Queries:

Write relational algebra expressions that will produce a relation containing:

• Q1: Loan number with value over $1000.

Πloan\_number (σ loan\_amount>1000(loan))

• Q2: Customers’ name and email with the amount of their loan (the amount of loan should

be NULL if a customer does not have any loan)

Πcustomer\_name, customer\_email, Loan\_amount(Loan⟖customer\_id=Loan.Customer\_id)

• Q3: Retrieve the number of transactions per each account.

• Q4: Retrieve all the customers having their account in “active” state.

Πname,Customer\_id, gender,Birth\_date, City, Address,Postal\_code, Home\_Phone,Mobil\_phone, Email (σstatus IN (“Active”) (Customer ⨝Customer.customer\_id = account.Customer\_id Account))

Part 2 - SQL Queries:

Write a SQL command for the following:

• Q1: Retrieve the customers who are living in “Trondheim” (Returns 5 records)

SELECT \* FROM `Customer` WHERE city = («Trondheim»);

• Q2: Retrieve the customers who have their email address under the commercial internet

domain (.com) (Returns 5 records)

Select \* From `Customer` Where Email LIKE («.com»);

• Q3: Retrieve the information of loans given to the customers in each branch between

2019-06-01 and 2020-06-01. (Returns 4 records)

SELECT \* FROM `LOAN`

WHERE Starting\_Date BETWEEN «2019-06-01» AND «2020-06-01»

ORDER BY Branch\_code;

• Q4: Retrieve the youngest customer who has taken a loan. (Returns 1 record)

SELECT customer.\*, MAX(Birth\_date) FROM `Loan`, `Customer`

Where Customer.Customer\_id = Loan.Customer\_id

• Q5: Write a SQL query that retrieves customers without any loans. (Returns 4 records)

SELECT customer.\* FROM `customer` LEFT OUTER JOIN `loan` ON

Customer.Customer\_id = Loan.Customer\_id

WHERE

Loan\_number IS NULL

• Q6: Retrieve the number of transactions for each account during the year 2019 (Returns 8

records)

SELECT COUNT (Transaction\_id), Account\_number from `depositor`

WHERE depositor.Date Like («2019%»)

GROUP BY Account\_number;

• Q7: Add a new customer with information below then open an inactive account in the

given branch:

INSERT INTO `customer` (name, Customer\_id, Gender, Birth\_date, City, Address, Postal\_code, Home\_Phone, Mobile\_phone, Email) Values («Ryan Ishus», 10016, «M», «1991-01-10», «Trondheim», «Bakkegata 15», 7049, 75432103, 45464783, «ryan00@realmail.no»)

INSERT INTO `Account` (Account\_number, Customer\_id, Branch\_code, Balance, Opening\_date, Account.Status) VALUES («ac1001», 10016, b2, 1000, «2021-01-18», «Inactive»)

Q8: Update the “Status” of account of customer Ryan Ishus to “Active”.

UPDATE `account` SET Account.status = «Active» WHERE Customer\_id IN («10016»)

• Q9: Delete the loans which their loan period is NULL.

DELETE FROM `Loan` WHERE Loan\_period IS NULL