Oblig-1 Databaser Bjørn Kristian Strand

Part 1 - Algebra Queries:

Q1: Loan number with value over $1000.
A:

```
Πloan_number(σloan_ammount > 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)
A:

```
Πname, email, Loan_ammount(customer ⟕ customer_id = loan.customer_id
Loan)
```

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

```
Account_number COUNT Transaction_id(Account⋈ Account_number =
depositor.Account_number Depositor )
```

Q4: Retrieve all the customers having their account in "active" state
A:

```
σstatus = 'Active'(Customer⋈ Customer_id = Account.Customer_id Account )
```

Part 2 - SQL Queries:

Q1: Retrieve the customers who are living in "Trondheim" (Returns 5 records)
A:

```
SELECT * FROM `customer`
WHERE City = "Trondheim";
```

Q2: Retrieve the customers who have their email address under the commercial internet
domain (.com) (Returns 5 records)
A:

```
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)
A:

```
SELECT * FROM `loan`
WHERE Starting_Date BETWEEN '2019-06-01' AND '2020-06-01';
```

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

```
SELECT * FROM customer WHERE Birth_date = (SELECT MAX(Birth_date)
FROM customer
JOIN loan on customer.Customer_id=loan.Customer_id);
```

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

```
SELECT * FROM `customer`
LEFT OUTER JOIN loan ON customer.Customer_id = loan.Customer_id
WHERE loan.Loan_number IS NULL;
```

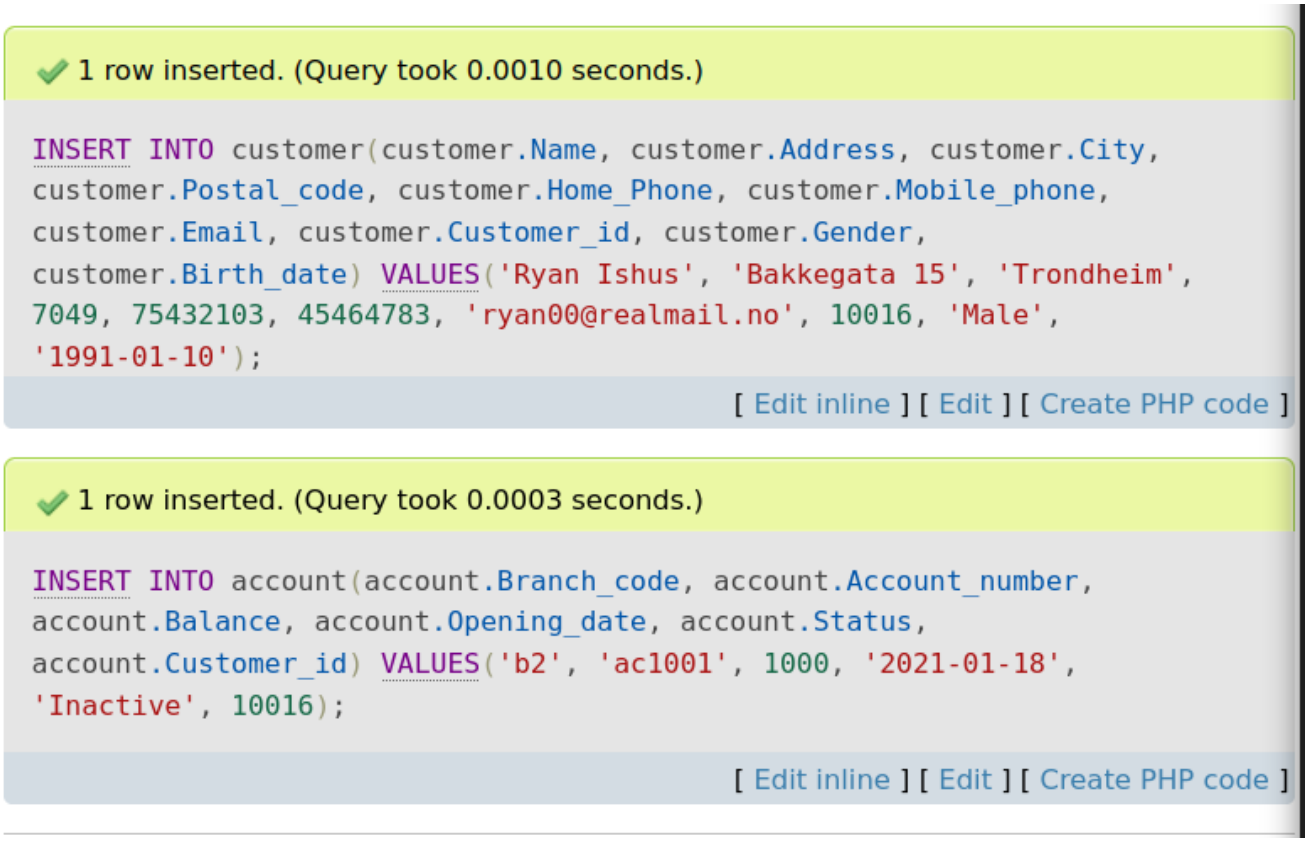
Q6: Retrieve the number of transactions for each account during the year 2019 (Returns 8
records)
A:

```
SELECT Account_number, COUNT(Transaction_id) FROM `depositor`
WHERE Date LIKE '2019-%'
GROUP BY Account_number;
```

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

```
INSERT INTO customer(customer.Name, customer.Address, customer.City,
customer.Postal_code, customer.Home_Phone, customer.Mobile_phone,
customer.Email, customer.Customer_id, customer.Gender,
customer.Birth_date)
VALUES('Ryan Ishus', 'Bakkegata 15', 'Trondheim', 7049, 75432103,
45464783, 'ryan00@realmail.no', 10016, 'Male', '1991-01-10');
INSERT INTO account(account.Branch_code, account.Account_number,
account.Balance, account.Opening_date, account.Status,
```

```
account.Customer_id)
VALUES('b2', 'ac1001', 1000, '2021-01-18', 'Inactive', 10016);
```



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

```
UPDATE account
SET Status = 'Active'
WHERE Customer_id = (
SELECT customer.Customer_id FROM `customer`
    WHERE customer.Name = 'Ryan Ishus'
);
```

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

```
DELETE FROM loan
WHERE Loan_period IS NULL;
```