Student: Amir Yakubov

Group: IA-2002

Assignment 6

Practice task 1.

a. Create the "shipping\_service" and "package" tables, including all of columns, data types, and constraints. Make sure to create the foreign key constraint.

Create table shipping\_service(

ship\_id decimal(12)primary key,

service\_name varchar(32) not null

);

Create table package(

package\_id decimal(12)primary key,

description varchar(25) not null,

date\_delivered date not null ,

ship\_id decimal(12),

Foreign key (ship\_id) references shipping\_service(ship\_id));

select \* from shipping\_service, package;

b. Insert the following companies into "shipping\_service" adding ship\_id of your choice.

insert into shipping\_service values

(100,'Kazpost'),

(101,'DHL' ),

(102, 'FedEx'),

(103, 'PonyExpress');

c. Insert the following package details into the "package" table adding package\_id of your choice.

insert into shipping\_service values

(104, ' no delivery service');

insert into package values

(1000,'IT devices','2019.05.25', 100),

(1001,'fragile box', '2019.06.15',101 ),

(1002, 'sport equipment', '2018.04.20',102),

(1004, 'null', '2019.07.22',104);

d. Write a single query to change package description to "Books," date\_delivered to "17-June-2019" for DHL delivering service.

update package

set description = 'Books', date\_delivered = '2019.06.17'

where package\_id = 1001;

e. Write a single query to add a new record to "package" table delivered by FedEx, your choice of "description" and "delivered date".

insert into package values

(1005,'books','2019.03.03', 102);

f. Write a single query to delete all the packages delivered after June.

delete from package

where date\_delivered > '2019.06.30';

g. Write a single query to delete the table "shipping service". What kind of problems you are encountered with? Please, explain.

Error. I cannot drop the shipping\_service table object because other objects depend on it.

Practice task 2.

a. Create tables including all columns, data types, and constraints. Make sure to create PK and FK constraints.

Create table job(

job\_id decimal(12)primary key,

name varchar(32),

fixed\_salary decimal(32)

);

Create table employee(

employee\_id decimal primary key,

fname varchar(32),

lname varchar(32),

hire\_date date,

quit\_date date,

job\_id decimal(12),

Foreign key (job\_id) references job(job\_id));

select \* from job, employee;

b. Write a single query to change the constraint for quit\_date to NULL.

UPDATE employee

SET quit\_date = NULL;

c. Insert the following data with your choice of job\_id and employee\_id.

- John Robbins is hired for “IT specialist” position on 20, April 2018. Salary: 5000 $.

- Adam Levine is hired for “Accountant” position on 1, January 2017. Salary 2000$.

- John Harvard has worked as a Lawyer from 1.02.2015-03.05.2019. Salary 7000$.

- Bob Lewis was hired for “IT specialist” position on 18, April 2015. Salary: 5000 $.

- Henry Gold worked as an “PR manager” from 01 January, 2015 to 24 May, 2019. Salary 4000$.

insert into job values

(100,'IT specialist', 5000 ),

(101,'Accountant', 2000 ),

(102, 'Lawyer', 7000),

(104, 'PR manager', 4000);

insert into employee values

(1000,'John', 'Robbins', '2018.04.20',NULL, 100),

(1001,'Adam', 'Levine', '2017.01.01', NULL, 101 ),

(1002,'John', 'Harvard', '2015.02.01','2019.05.03', 102),

(1003,'Bob', 'Levis', '2015.04.18',NULL,100),

(1004, 'Henry','Gold','2015.01.01','2019.05.24',104);

d. Write a single query to display the job\_id and hire date for all employees with fname "John".

select job\_id, hire\_date

from employee

where fname like 'John';

e. Write a single query to display the fname and lname of employees whose salary is higher than 5000$.

select fname, lname

from employee,job

where employee.job\_id = job.job\_id and

fixed\_salary > 5000 ;

f. Write a single query to display the lname and hire date for all employees who were hired between 2015 and 2019.

select fname, hire\_date

from employee,job

where employee.job\_id = job.job\_id and hire\_date

between '2015.01.01' and '2019.01.01' ;

g. Write a single query to display the employees with the same job position.

select fname, lname

from employee,job

where employee.job\_id = job.job\_id and employee.job\_id

in (select job\_id from employee group by job\_id having count(\*)>1);

h. Write a single query to modify the salary of present employee(s) up to 10000$.

update job

set fixed\_salary=10000 from employee

where employee.job\_id = job.job\_id and quit\_date is null;

i. Write a single query to remove the employees who are not working anymore.

delete from employee

where quit\_date is not null;

Practice task 3.

a. Write a SQL query to find the number of jobs available in the "employee" table.

create table employee (

emp\_id decimal(12) primary key,

fname varchar(32),

lname varchar(32),

job\_name varchar(32),

salary int,

manager\_id decimal(12),

dep\_id decimal(12)

);

select\*from employee;

insert into employee values

(100,'Bob','Marley', 'Programmer', 240000, 01, 1001),

(101,'Craig','David', 'Accountant', 220000, 02, 1002),

(102,'Steven','Hock', 'Salesman', 3000, 03, 1003),

(103,'Bradley','Cooper', 'PR', 4000, 04, 1004),

(104,'Adam','Levian', 'Audit', 5000, 05, 1005),

(105,'Mark','May', 'Tax', 6000, 08, 1004),

(106,'Arthur', 'King', 'Finance', 7000, 02, 1002),

(107, 'Daniel', 'Faviet', 'Programmer', 3200, 01, 1000),

(108, 'Karen', 'Dias', 'IT Manager' ,100000, 03, 1004),

(109, 'James', 'Oliver', 'HR',7000, 01 ,1005),

(110, 'Jay', 'Shuttle', 'Audit', 5000, 05, 1000),

(111, 'Harry', 'Potter', 'Programmer', 45000, 02, 1005),

(112, 'Ki', 'Gee', 'Salesman', 300000, 04, 1002),

(113, 'Peter', 'Vargas', 'QA', 100000, 03, 1001),

(114, 'Winston', 'Taylor', 'PR' ,12000, 03, 1006),

(115, 'Kelly', 'Chung', 'Finance', 39000, 06, 1007),

(116, 'Kenny', 'Freenay', 'Economist', 5000, 07, 1007),

(117, 'Shirley', 'Higgins', 'Salesman', 400, 03, 1009),

(118, 'Donald' ,'Cornell', 'Programmer' ,300000, 01, 1009);

b. Write a SQL query to get the total salaries payable to employees.

select sum (salary)

from employee;

c. Write a SQL query to get the minimum salary paid to employees.

select min (salary)

from employee;

d. Write a SQL query to get the maximum salary of an employee working as a "Programmer".

select max (salary)

from employee

where job\_name = 'Programmer';

e. Write a SQL query to get the "average salary " and "number of employees" working in the department (ID "1002").

select count(\*) as "number of employees", avg(salary)::numeric(10,2) as "average"

from employee

where dep\_id=1002;

f. Write a SQL query to get the highest, lowest, total, and average salary of employees.

select max(salary) as "highest",

min(salary) as "lowest",

sum(salary) as "total",

avg(salary)::numeric(10,2) as "average"

from employee;

g. Write a SQL query to get the "number of employee(s)" working for each job position.

select count(distinct job\_name) as "number of employees"

from employee;

h. Write a SQL query to find the "manager ID" and the "salary of the lowest-paid employee(s)" under that manager.

select manager\_id, salary

from employee

where salary = (select min(salary) from employee);

i. Write a SQL query to get the "department ID" and the "total salary payable in each department".

select manager\_id, sum (salary)

from employee

group by manager\_id

j. Write a SQL query to get "the average salary" for each job position excluding "salesman".

select job\_name, avg (salary)::numeric(10,2)

from employee

where job\_name not in('Salesman')

group by job\_name;