Database Management System – cs422 DE

Lab 3 – Week 7

This Lab is based on Transact-SQL.

- Submit your *own work* on time. No credit will be given if the lab is submitted after the due date.
- Note that the completed lab should be submitted in .doc, .docx, .rtf, .pdf or .zip format only.

---

1) [3] Write and execute a T-SQL stored procedure *Factorial*(*n*), which computes and outputs the factorial of the input parameter *n*. If *n* is negative, then the procedure prints an error message.
Attach the screenshots of the output and the command which you used to execute the SP.
ANS: (Note: I used MySQL Workbench)

```
CREATE PROCEDURE `Factorial`(n INT)
BEGIN
  DECLARE result INT;
  SET result = 1;

  IF n <0 THEN
    select 'Error! The value of n should be zero and above';
  ELSEIF n=0 THEN
    SELECT n;
  ELSE
    WHILE n >0 DO
      SET result = result * n;
      SET n=n-1;
    END WHILE;
    SELECT result;
  END IF;
END
```

```
1 •  USE lab3_wk7;
2
3 •  call factorial(5);
```

100% 14:1

Result Grid | Filter Rows: Search | Export:

result
120

2) [7] Create a Table *Employee* with the fields: social security no. (primary key), name, position, no. of dependents, annual salary.

Write and execute a T-SQL procedure *Compute_Tax* to do the following:

- Create a new table *Tax* with fields: social security no., income tax.
- Fill the table *Tax* with data by computing the income tax for each person in the Employee Table.

The income tax is computed from the annual salary S and the number of dependents D.

Net Salary: S - (7000 + D*950)

Tax Computed as follows:

- 10% of the first 15,000 of net salary;
- plus 15% of the next 15,000 of net salary;
- plus 28% of any net salary over 30,000.

For getting full credit for this problem, you need to show me the complete code for the *Compute_Tax* SP. Also attach the screenshots of the *Employee* and the new *Tax* table.

ANS:

Employee table (sample data from https://dlptest.com/sample-data/)

ssn	name	position	noOfDependen...	annualSalary
044-34-6954	Tim Lowe	Accountant	4	80796.00
151-32-2558	Rebecca Zwick	Sales Executive	5	356597.00
172-32-1176	Johnson White	Accountant	3	146416.00
205-52-0027	Agnes Nelson	Accountant	4	301422.00
213-46-8915	Marjorie Green	Marketing Representative	1	263337.00
300-62-3266	Victor Faulkner	Marketing Representative	1	157632.00
322-84-2281	Albert Iorio	Marketing Representative	1	247977.00
404-12-2154	Mireille Townsend	IT Manager	1	180042.00
421-37-1396	Susan Davis	IT Manager	0	245589.00
421-90-3440	Adriane Morrison	Marketing Representative	2	107398.00
449-48-3135	Mark Hall	IT Manager	4	155112.00
451-80-3526	Thomas Santos	Sales Executive	4	120099.00
458-02-6124	Christopher Diaz	Accountant	3	120776.00
461-97-5660	Gail Watson	Sales Executive	5	138875.00
465-73-5022	Teresa Kaminski	Sales Executive	3	152854.00
477-36-0282	Monte Mceachern	Sales Executive	3	358778.00
489-36-8350	Robert Aragon	Marketing Representative	4	245441.00
505-88-5714	Lillian Venson	Sales Executive	3	367008.00
514-14-8905	Ashley Borden	Accountant	1	51704.00
514-30-2668	Jacki Russell	IT Manager	1	185981.00
524-02-7657	Jerome Munsch	Marketing Representative	4	220222.00
559-81-1301	James Heard	Sales Executive	1	398662.00
587-03-2682	Lynette Oyola	Sales Executive	4	51018.00
612-20-6832	Rick Edwards	IT Manager	3	250665.00
624-84-9181	Danny Reyes	IT Manager	1	20618.00
646-44-9061	Charles Jackson	Marketing Representative	2	301049.00
660-03-8360	Lisa Garrison	Accountant	3	149678.00
687-05-8365	Stacey Peacock	Marketing Representative	3	43981.00
690-05-5315	Thomas Conley	Secretary	1	193907.00
751-01-2327	Julie Renfro	Sales Executive	3	109759.00

Tax table

```
1
2 •  SELECT * FROM lab3_wk7.Tax;
```

100% 28:2

Result Grid Filter Rows: Search

ssn	income_tax
044-34-6954	14948.88
151-32-2558	91907.16
172-32-1176	33588.48
205-52-0027	76724.16
213-46-8915	66858.36
300-62-3266	37260.96
322-84-2281	62557.56
404-12-2154	43535.76
421-37-1396	62154.92
421-90-3440	22929.44
449-48-3135	35757.36
451-80-3526	25953.72
458-02-6124	26409.28
461-97-5660	30945.00
465-73-5022	35391.12
477-36-0282	93049.84
489-36-8350	61049.48
505-88-5714	95354.24
514-14-8905	7601.12
514-30-2668	45198.68
524-02-7657	53988.16
559-81-1301	104749.36
587-03-2682	6611.04
612-20-6832	62778.20
624-84-9181	1266.80
646-44-9061	77151.72
660-03-8360	34501.84
687-05-8365	4906.68
690-05-5315	47417.96
751-01-2327	23324.52

Compute_Tax() stored procedure

```
CREATE PROCEDURE `Compute_Tax`()
begin
   declare done tinyint default 0;
   DECLARE c1 cursor for
      select ssn, noOfDependents, annualSalary from Employee2;
   open c1;
      empLoop: loop
         begin
            declare ssn varchar(11);
            declare noOfDependents int;
            declare annualSalary DECIMAL(18,2);
            declare tax DECIMAL(18,2);
            declare netSalary DECIMAL(18,2);

            declare continue handler for sqlstate '02000' set done = 1;
            fetch c1 into ssn,noOfDependents, annualSalary;
               if done then
                  leave empLoop;
               end if;
               SET netSalary = annualSalary - (7000 + noOfDependents * 950);
               IF (netSalary < 0 ) THEN
                  LEAVE empLoop;
                END IF;
               IF (netSalary <= 15000) THEN
                  SET tax = 0.10 * netSalary;
               ELSE
                     IF (netSalary >= 30000) THEN
                        SET tax = 0.10 * 15000 + 0.15 * 15000 + 0.28 * (netSalary - 30000);
                     ELSE
                        SET tax = 0.10 * 15000 + 0.15 * (netSalary -15000);
                    END IF;
              END IF;
               insert into tax (ssn, income_tax) values (ssn, tax);
         end;
      end loop;
   close c1;
end
```