CSC 355 Database Systems 501T-530
Winter 2023
Assignment 5 – PL/SQL and Triggers

Readings: Sections 9.3-9.4 of Ullman/Widom, and the posted PL/SQL Examples and Trigger Examples. If you want an additional PL/SQL reference, I recommend Oracle's PL/SQL User's Guide and Reference.

Part I – Anonymous PL/SQL block

Consider the table C_WORKER with attributes WID, WName, and WSalary, and the table TAXVALUES with attributes MaxAmount and TaxRate, defined and populated by the following script:

```
DROP TABLE C_WORKER CASCADE CONSTRAINTS;
CREATE TABLE C_WORKER
(
       WID            CHAR(3)        PRIMARY KEY,
       WName          VARCHAR2(12),
       WSalary        NUMBER(7,2)
);

INSERT INTO C_WORKER VALUES ('114', 'Alfred', 45000);
INSERT INTO C_WORKER VALUES ('009', 'Bernard', 85000);
INSERT INTO C_WORKER VALUES ('233', 'Cayenne', 72000);
INSERT INTO C_WORKER VALUES ('903', 'Delta', 64000);
INSERT INTO C_WORKER VALUES ('101', 'Eliza', 81000);
INSERT INTO C_WORKER VALUES ('098', 'Francis', 87500);
INSERT INTO C_WORKER VALUES ('092', 'Garfield', 56300);
SELECT * FROM C_WORKER;

DROP TABLE TAXVALUES CASCADE CONSTRAINTS;
CREATE TABLE TAXVALUES
(
       MaxAmount      NUMBER(7,2),
       TaxRate        NUMBER(3,2)
);
INSERT INTO TAXVALUES VALUES (10000.00, 0.13);
SELECT * FROM TAXVALUES;
COMMIT;
```

Write a script containing just an anonymous PL/SQL block that will do the following:

First, read the maximum tax amount and tax rate from the TAXVALUES table, store them in variables, and display their values. (You may assume that the TAXVALUES table contains exactly one record.) Next, for each worker in the C_WORKER table, compute the amount of tax to be withheld from their salary, as follows:

- If the salary times the tax rate is less than the maximum tax amount, then the amount of tax to be withheld is the salary times the tax rate. (So, for the values in the sample TAXVALUES table, a worker with a salary of 40000 would have a tax amount of 40000 * 0.06 = 2400 withheld.)

- If the salary times the tax rate is greater than or equal to the maximum tax amount, then the amount of tax to be withheld is the maximum tax amount. (So, for the values in the sample TAXVALUES table, a worker with a salary of 65000 would have a tax amount of 3000 withheld, since 65000 * 0.06 = 3900 exceeds 3000.)

Output each worker's information on a single line, showing their ID, salary, tax amount, and net pay (i.e., the salary minus the amount of tax withheld). Add a '+' to the end of each line in which the maximum tax amount is withheld. Also, compute the total tax withheld from all the workers' salaries and report that total at the end. For the sample data given, the output should be:

```
Tax rate: .13
Maximum Tax: 10000

114:  45000   5850  39150
009:  85000  10000  75000 +
233:  72000   9360  62640
903:  64000   8320  55680
101:  81000  10000  71000 +
098:  87500  10000  77500 +
092:  56300   7319  48981

Total tax withheld: 60849
```

Note: Be aware that this is just an example – your anonymous PL/SQL procedure and trigger should work in general, not just for the given sample data.

Part II – Triggers

Consider the table T_AUDIT with attributes WID, OLD_Sal, NEW_Sal, NEW_Tax, auditDate defined and populated by the following script:

```
DROP TABLE T_AUDIT CASCADE CONSTRAINTS;
 CREATE TABLE T_AUDIT
(
       WID            CHAR(3) REFERENCES C_WORKER(WID),
       OLD_Sal        NUMBER(7,2),
       NEW_Sal        NUMBER(7,2),
       OLD_Tax        NUMBER(7,2),
       NEW_Tax        NUMBER(7,2),
       auditDate      DATE DEFAULT SYSDATE,
       PRIMARY KEY (WID, auditDATE)
);
```

Write a SQL trigger that will do the following:

The trigger should be named **NewTaxByLastFirst** (replace Last and First with your last and first names, respectively. In my case, it should be NewTaxByPerazaEliecer). The trigger should be fired after a salary of a worker changes. It only should run for those workers not withholding the maximum tax before the update. It should insert all the information requested in the T_AUDIT table for each time the trigger body runs. It should output each worker's information on a single line, showing their ID, old salary, old tax amount, new salary, new tax amount, and difference of tax withheld. The output should look like this:

```
   114:    45000     5850    49500     6435      585
   233:    72000     9360    79200    10296      936
   903:    64000     8320    70400     9152      832
   092:    56300     7319    61930   8050.9    731.9
```

Note: this output if the result of executing `UPDATE C_WORKER SET WSalary = 1.1 * WSalary;`

	WID	OLD_SAL	NEW_SAL	OLD_TAX	NEW_TAX	AUDITDATE
1	114	45000	49500	5850	6435	19-FEB-23
2	233	72000	79200	9360	10296	19-FEB-23
3	903	64000	70400	8320	9152	19-FEB-23
4	092	56300	61930	7319	8050.9	19-FEB-23

Note: this is the result of running `SELECT * FROM t_audit;` after the above UPDATE has run.

The tax rate and max amount may be hard coded for the trigger condition but must be retrieved from the tax table when used in the body of the trigger.

Part III – Script

- Include a comment at the top of your script file giving your name, the course number and section number, the assignment number, and the date of submission (i.e., fill in the appropriate values for name, and submission date in a comment such as the one below):

```
/*
YourName
CSC 355 Section SectionNumber
Homework 5
SubmissionDate
*/
```

- Add a single line comment for each of the parts, right before your code for the part

```
-- Part I - Anonymous PL/SQL block
       Here goes your code for Part I

-- Part II - Triggers
       Here goes your code for Part II
```

- Save your script as HW04_SCRIPT_YOURNAME.sql

Submission

- Submit one file to the Homework 5 submission folder, containing your HW05_SCRIPT_YOURNAME.sql script file
- You do not have to submit any output.

Remarks:

1. It is your responsibility to make sure that the files you have uploaded are readable and in the correct locations. You should verify that you can successfully download them from the submissions folder after submitting them to be sure that they have been uploaded correctly.
2. Remember that all work must be completed individually and without copying, either entirely or in part, from any examples posted or from anyone else's work. Do not post this assignment to any website in search of answers, and do not consult posted answers on any website while completing the assignment.
3. You may copy and paste the supplied script into SQLDeveloper to set up the tables and test your code, but your submitted solutions should include only the code you have written for the solution – not any of the supplied code. Points will be deducted if you left any of the supplied codes in your script.