# CSC 352/452 Database Programming Section 310/510 (Online class)

# DePaul University College of CDM Summer II 2019/2020

## Midterm Exam

**Exam Date**: 8/3/2020

**Exam Time:** 6:00-9:00pm online

You have three hours to complete this exam, which has six questions (some of which have multiple parts) and is worth a total of 100 points. You need to download this exam file to your local machine and then when you completed the exam, upload this MS Word file with all your answers. Please only used this document and only this document for submission.

Please write all of your answers on this exam file/sheets and be sure that your final answer to each question is clearly indicated.

Good luck!

**Name: \_\_\_\_\_\_\_\_\_Ximan Liu\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

(The exam has a total of 9 pages, including blank pages immediately after Problems 6.)

Question #1: Multiple choice questions (A-O):

**Answer bank for** Question #1**.**

If you want your Question #1 to be graded, you **must fill in** the following table with your answers from Question #1.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Q#** | **Answer** | **Q#** | **Answer** | **Q#** | **Answer** |
| **A.** | **D** | **F.** | **B** | **K.** | **C** |
| **B.** | **A** | **G.** | **C** | **L.** | **C** |
| **C.** | **B** | **H.** | **A** | **M.** | **C** |
| **D.** | **A** | **I.** | **C** | **N.** | **B** |
| **E.** | **C** | **J.** | **B** | **O.** | **B** |

1. Which of the following *isn’t* a common error when entering and executing SQL statements?

(a) Misspelling a keyword

(b) Misspelling the name of a table

(c) Connecting as the wrong user

(d) Forgetting to connect to a database

1. Which of the following isn’t true about a stored procedure?
   * + 1. A stored procedure is compiled each time that it is executed.
       2. A stored procedure is stored in the database.
       3. A stored procedure can include both SQL and PL/SQL statements.
2. A database driver is software that lets the

(a) data access model communicates with the application program

(b) data access model communicates with the database

(c) application program communicates with the data access model

(d) application program communicates with the database

1. Which of the following expressions will *not* compute 10% of the balance due if balance due is the invoice total minus the credit total minus the payment total?

(a) **invoice\_total - credit\_total - payment\_total / 10**

(b) **(invoice\_total - payment\_total - credit\_total) / 10**

(c) **(invoice\_total - (payment\_total + credit\_total)) \* 0.10**

(d) **((invoice\_total - payment\_total) - credit\_total) / 10**

1. When coded in a WHERE clause, which of the following search conditions will *not* return a result set that includes all invoices with an invoice\_total value of $1000 or less?

(a) **invoice\_total <= 1000**

(b) **NOT (invoice\_total > 1000)**

(c) **invoice\_total IN (0,1000)**

(d) **invoice\_total BETWEEN 0 AND 1000**

1. A student receives an error due to the following statement in the DECLARATION of a PL/SQL block:

PI CONSTANT NUMBER;

What is wrong with this statement?

1. There is not enough memory in the program for the constant.
2. There is no value associated with the constant
3. There is no datatype associated with the constant.
4. PI is a reserved word.
5. Which statement most accurately describes the result of not creating an exception handler for a raised exception?
6. The program will continue without raising the exception.
7. There will be a memory leak.
8. Control will pass to the exception handler of the caller of the PL/SQL block
9. The program will return a %NOTFOUND error.
10. A procedure declares a user-defined exception but does not raise it explicitly. Which of the following statements is true about this procedure?
11. The user-defined exception will never be raised.
12. The user-defined exception will be handled by a WHEN OTHERS exception

handler.

1. The procedure will fail on compile.
2. The user-defined exception is defined incorrectly.
3. Which line of the following code fragment has an error?
4. A := B + 35;
5. if A < 10 then
6. B := 15;
7. elseif A < 50 then
8. B := 8;
9. end if;
10. line 2
11. line 3
12. line 4
13. line 5
14. Consider the following SELECT statement. What happens if there are no rows satisfying the WHERE condition?

SELECT COUNT(\*) INTO V\_COUNT

FROM EMP

WHERE SALARY < 10;

1. A NO\_DATA\_FOUND exception is raised
2. The SELECT statement executes successfully
3. A TOO\_FEW\_ROWS exception is raised
4. COUNT(\*) is not allowed to use in PL/SQL
5. Procedure C calls procedure D. If an error is raised in procedure D and no exception handler exists in procedure D, what happens?

(a) An error message is displayed to the user.

(b) An error is raised in procedure C.

(c) Program control moves to procedure C’s EXCEPTION section.

(d) The results are unpredictable.

1. How many values can a procedure return to a calling environment?

(a) none

(b) the same as the number of parameters

(c) the same as the number of parameters that have the OUT mode

(d) at least one

1. Which parameter mode must be used to have one value sent to a parameter and a different value returned by the same parameter?

(a) IN

(b) OUT

(c) IN OUT

(d) OUT IN

1. Passing a parameter value by value means the value is

(a) used in an assignment statement

(b) copied to the formal parameter

(c) referenced with a pointer and not copied

(d) used in several parameters.

1. Which of the following is a compiler hint used to override the default parameter-passing method?
2. ByReference
3. NOCOPY
4. ByValue
5. COPY

Question #2: The following set of relations records information about university students, courses and assigned grades. The Student relation contains information about the student, including the name (full name is stored in one column), address and the (projected) year of graduation. The Course relation records information about courses with course name serving as the primary key, department for the course and the number of credits it provides. Finally, the Grade relation records information about the grades given; CName is the foreign key referring to the primary key of the Course relation and StudentID is the foreign key referring to the primary key of the Student relation. The grades are a numeric value given on a 4-point system.

**STUDENT(StudentID, Name, Address, GradYear)**

**GRADE(CName, StudentID, CGrade)**

**COURSE(CName, Department, Credits)**

For each part below, give an SQL query to display the requested information.

**a.** Display the Name, Address and Graduation Year for students whose first name is Jane (you can assume names are listed as ‘FirstName LastName’), ordered from the most recent graduation year to the least recent.

**Your answer**:

**SELECT Name, Address, GradYear**

**FROM STUDENT**

**WHERE Name = ‘Jane%’**

**ORDER BY GradYear;**

**b.** For each course, display the average and the highest (maximum) grade. Please use a single SQL query.

**Your answer**:

**SELECT CName, AVG(CGrade), MAX(CGrade)**

**FROM GRADE**

**GROUP BY CName;**

Question #3: Consider the following program:

DROP TABLE mytable;

CREATE TABLE mytable (name varchar2(20), ssn char(9));

DECLARE

fun NUMBER;

myexception EXCEPTION;

BEGIN

BEGIN

BEGIN

INSERT INTO mytable VALUES('John Doe', '111222333');

RAISE myexception;

END;

INSERT INTO mytable VALUES('David Wood', '333222111');

COMMIT;

fun := 5/0;

BEGIN

INSERT INTO mytable VALUES('Mary Jones', '222222222');

RAISE myexception;

END;

EXCEPTION

WHEN others THEN

INSERT INTO mytable VALUES('Larry Young', '999999999');

END;

INSERT INTO mytable VALUES('Bruce Smith', '000000000');

EXCEPTION

WHEN myexception THEN

INSERT INTO mytable VALUES('Andrew Lee', '888888888');

END;

/

Give the rows in table mytable after the execution of the above anonymous PL/SQL block. Explain briefly your answer.

**Your answer**:

**NAME SSN**

**1 John Doe 111222333**

**2 Larry Young 999999999**

**3 Bruce Smith 000000000**

**The table has two identifiers which is NAME and SSN, so there are here as shown.**

**Firstly, the program will pop up NAME and SSN about John Doe, because he is the first content when BEGIN happened.**

**fun := 5/0 cannot be calculated because 0 cannot be divisor, so program will go to EXCEPTION WHEN others THEN. And content of Larry Young will show up.**

**Other user contents are not a user-defined exception, so will not pop out.**

**After this term end, finally there is a row of content for Bruth Smith, that’s why his content showed here.**

Question #4:

* 1. Suppose SSN and GPA are two columns of table students. Assume that the datatype of column SSN is not known and that GPA is known to have a datatype of NUMBER.

Write a **stored procedure** that when supplied with the SSN of a student, looks up the GPA of the student and passes the GPA back to the caller of the stored procedure. If the information of the student is not found, the value of GPA passed back is - 1.

**Your answer**:

**CREATE OR REPLACE PROCEDURE lookfor\_gpa (lookfor\_ssn IN NUMBER)**

**AS SSN students.GPA%TYPE;**

**BEGIN**

**SELECT GPA from students**

**WHERE SSN = lookfor\_ssn;**

**EXCEPTION**

**WHEN NO\_DATA\_FOUND THEN**

**RETURN GPA := -1;**

**END;**

**/**

* 1. Create a stored procedure named insert\_account that lets a user add a new row to the Accounts table in your schema. This procedure should have two parameters, one for each of the two columns in this table. Then, code a CALL statement that tests the procedure. (Note that this table doesn’t allow duplicate account descriptions.)

**Your answer**:

**CREATE OR REPLACE PROCEDURE insert\_account (a IN NUMBER, b IN NUMBER)**

**AS**

**BEGIN**

**INSERT INTO Accounts VALUES (a, b);**

**END;**

**/**

**CALL insert\_account(1, 2);**

Question #5: Give the rows of table TEMP after the following anonymous PL/SQL block has been executed.

-- Structure for TEMP table

CREATE TABLE TEMP

(n1 number, n2 number, n3 varchar2(30));

--PL/SQL Block

DECLARE

x NUMBER := 0;

counter NUMBER := 0;

BEGIN

FOR i IN 1..2 LOOP

x := x + 1000;

counter := counter + 1;

INSERT INTO TEMP VALUES (x, counter, 'in OUTER loop');

/\* start an inner block \*/

DECLARE

x NUMBER := 0; -- this is a local version of x

BEGIN

FOR i IN 1..2 LOOP

x := x + 1; -- this increments the local x

counter := counter + 1;

INSERT INTO TEMP VALUES (x, counter, 'inner loop');

END LOOP;

END;

END LOOP;

COMMIT;

END;

/

**Your answer**:

Rows inserted into TEMP are:

**N1 N2 N3**

**1000 1 in OUTER loop**

**1 2 inner loop**

**2 3 inner loop**

**2000 4 in OUTER loop**

**1 5 inner loop**

**2 6 inner loop**

Question #6: There are three tables ***students1***, ***students2*** and ***students3*** with different schemas. However, they have two common columns called **Name** and **SSN**. The datatypes of **Name**  and **SSN** are the same in these three tables. A given student is represented in **only one** of the tables ***students1***, ***students2*** and ***students3***. However, it is not known which table will contain information of this student.

Write a PL/SQL block (either an anonymous PL/SQL block or a function or procedure) that will display the *student name* as well as the *name of the table* containing information of this student when the SSN of this student is supplied to this PL/SQL block when it is called.

**Your answer**:

**CREATE OR REPLACE PROCEDURE locateStudent(aSSN IN students1.SSN%TYPE)**

**stu\_name students1.Name%TYPE;**

**tabName varchar2(10);**

**BEGIN**

**begin**

**select Name, 'students1' into stu\_name, tabName**

**from students1**

**where SSN = aSSN;**

**GOTO myLabel;**

**exception**

**WHEN NO\_DATA\_FOUND THEN**

**NULL;**

**end;**

**begin**

**select Name, 'students2' into stu\_name, tabName**

**from students2**

**where SSN = aSSN;**

**GOTO myLabel;**

**exception**

**WHEN NO\_DATA\_FOUND THEN**

**NULL;**

**end;**

**select Name, 'students3' into stu\_name, tabName**

**from students3**

**where SSN = aSSN;**

**<<myLabel>>**

**dbms\_output.put\_line('Student name is ' || stu\_name );**

**dbms\_output.put\_line(aSSN || ' is found in table ' || tabName);**

**EXCEPTION**

**WHEN NO\_DATA\_FOUND**

**THEN**

**dbms\_output.put\_line(aSSN || ' is not found in any of the three tables');**

**END locateStudent;**

**/**