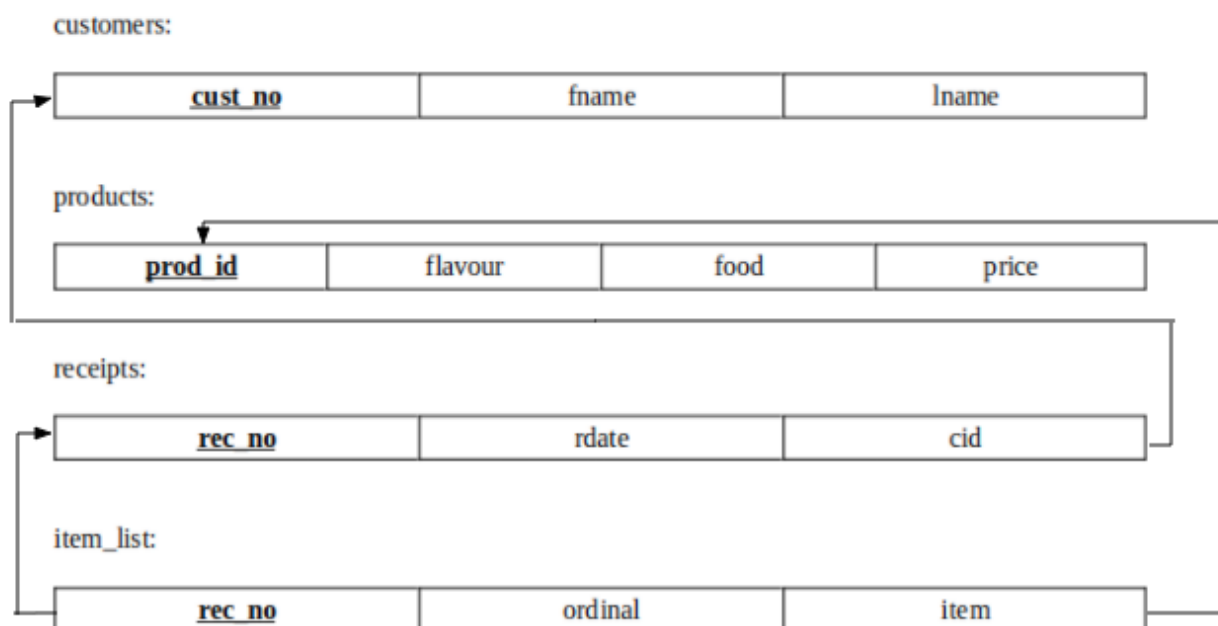


## Assignment 8 – Exception Handling

### Validation:

NAME: Krithika Swaminathan SEM: IV REG: A ROLL NO.: 057 SUBJ: DATABASE LAB				
S. No.	Date	Title	Page No.	Teacher's Sign / Remarks
1.	10/03/2022	A1: DDL Commands	2/10	
2.	17/03/2022	A2: DML Commands	8/10	Page 51/52
3.	07/04/2022	A3: Joins and Subqueries	9/10	Page 71/72
4.	21/04/2022	A4: Views	10/10	Page 21/22
-	23/04/2022	LAB TEST : A1, 2, 3	18/15	23/1A
5.	28/04/2022	A5: PL/SQL	10/10	28/1A
6.	12/05/2022	A6: PL/SQL	10/10	12/15/22
7.	26/05/2022	A7: Triggers	9/10	26/1A
8.	26/05/2022	A8: Exception handling	10/10	26/1A

### Schema diagram:



**Data file:**

```
SQL> @C:/Krithika/DBL/a8data.sql;
SQL> REM Population of Bakery Database
```

```
SQL> drop table item_list;
Table dropped.
```

```
SQL> drop table receipts;
Table dropped.
```

```
SQL> drop table products;
Table dropped.
```

```
SQL> drop table customers;
Table dropped.
```

```
SQL>
SQL> create table customers(
2      cust_no number(2) constraint c_pk primary key,
3      lname varchar2(20),
4      fname varchar2(20)
5      );
```

Table created.

```
SQL>
SQL> insert into customers values(1, 'LOGAN', 'JULIET');
1 row created.
```

```
.
.
.
```

```
SQL> insert into customers values(21, 'JOHN', 'DAVID');
1 row created.
```

```
SQL> create table products(
2      prod_id varchar2(20) constraint prod_pk primary key,
3      flavour varchar2(20),
4      food varchar2(20),
5      price number
6      );
```

Table created.

```
SQL>
SQL> insert into products values('20-BC-C-10','Chocolate','Cake',8.95);
1 row created.
```

```
.
.
.
```

```
SQL> insert into products values('51-BLU','Blueberry','Danish',1.15);
1 row created.
```

```
SQL> create table receipts(
2     rec_no number(5) constraint rec_pk primary key,
3     rdate date,
4     cid number(2) constraint rec_fk references customers(cust_no)
5 );
```

Table created.

```
SQL>
SQL> INSERT INTO Receipts values(18129, '28-Oct-2007', 15);
1 row created.
```

```
.
.
.
```

```
SQL> INSERT INTO Receipts values(34378, '23-Oct-2007', 6);
1 row created.
```

```
SQL> create table item_list(
2     rec_no number(5) constraint it_fk1 references receipts(rec_no),
3     ordinal number(2),
4     item varchar2(20) constraint it_fk2 references products(prod_id),
5     constraint item_pk primary key(rec_no,ordinal)
6 );
```

Table created.

```
SQL>
SQL> insert into item_list values(18129, 1, '70-TU');
1 row created.
```

```
.
.
.
```

```
SQL> insert into item_list values(34378, 2, '45-VA');
1 row created.
```

```
SQL>
SQL> REM *** End of database population ***
```

```
SQL>
SQL> REM ***** END OF DATA FILE *****
```

**Script file:**

```
SQL> @z:/a8exception.sql;
SQL> REM Assignment 8
SQL>
SQL> REM -----
> REM *** ASSIGNMENT QUESTIONS ***
SQL> REM -----
> REM Consider the schema used in Assignment 3.
SQL>
SQL>
SQL> REM **_____Write a PL/SQL block to handle the following exceptions:_____**
SQL>
SQL> REM 1. For the given receipt number, if there are no rows then display as "No order with the
SQL> REM given receipt <number>". If the receipt contains more than one item, display as
SQL> REM "The given receipt <number> contains more than one item". If the receipt contains
SQL> REM single item, display as "The given receipt <number> contains exactly one item". Use
SQL> REM predefined exception handling.
SQL>
SQL> create or replace procedure numitems (rno in receipts.rec_no%type) is
2 numrows number;
3 begin
4     select count(item) into numrows from item_list where rec_no = rno;
5     if (numrows = 0) then
6         dbms_output.put_line('No order with the given receipt number '||rno);
7     elsif (numrows = 1) then
8         dbms_output.put_line('The given receipt '||rno||' contains exactly one item.');
```

Procedure created.

```
SQL>
SQL> declare
2 rid receipts.rec_no%type;
3
4 begin
5
6 rid := &receipt;
7 numitems(rid);
8
9 end;
10 /
Enter value for receipt: 34378
```

```
old 6: rid := &receipt;
new 6: rid := 34378;
The given receipt 34378 contains more than one item
```

PL/SQL procedure successfully completed.

```
SQL> /
Enter value for receipt: 17947
old 6: rid := &receipt;
new 6: rid := 17947;
The given receipt 17947 contains exactly one item.
```

PL/SQL procedure successfully completed.

```
SQL> /
Enter value for receipt: 88888
old 6: rid := &receipt;
new 6: rid := 88888;
No order with the given receipt number 88888
```

PL/SQL procedure successfully completed.

```
SQL>
SQL> REM Validations:
SQL>
SQL> select rec_no, count(item) as numrows from item_list group by rec_no having rec_no =
34378 or rec_no = 17947 or rec_no = 88888;
```

REC_NO	NUMROWS
34378	2
17947	1

```
SQL>
SQL>
SQL> REM 2. While inserting the receipt details, raise an exception when the receipt date is greater
than the current date.
SQL>
SQL> create or replace trigger check_rdate
2 BEFORE INSERT ON receipts FOR EACH ROW
3 declare
4     cursor c1 is select * from receipts where rdate > (select sysdate from dual);
5     record c1%rowtype;
6 begin
7     open c1;
8     fetch c1 into record;
9     if c1%found then
10         raise_application_error(-20000,'Receipt date cannot be after current date!');
11     end if;
12 end;
```

13 /

Trigger created.

SQL>

SQL> REM Validations:

SQL>

SQL> insert into receipts values (12299,'01-JUL-2022',11);

insert into receipts values (12299,'01-JUL-2022',11)

\*

ERROR at line 1:

ORA-20000: Receipt date cannot be after current date!

ORA-06512: at "1057.CHECK\_RDATE", line 8

ORA-04088: error during execution of trigger '1057.CHECK\_RDATE'

SQL>

SQL>

SQL>

SQL> REM \*\*\*\*\* END OF FILE \*\*\*\*\*

---