LAB 7 SOLUTIONS

Exercise 1

Q1 Write a table-valued function that takes an id number (integer) as input and returns a table which contains all details of projects being worked on an employee of that id number.

```
CREATE function project_details(@id integer)
  RETURNS @projects1 table
  (projectid CHAR(8) NOT NULL,
  deptcode CHAR(5),
  description VARCHAR(200),
  startdate DATE,
  enddate DATE,
  revenue NUMERIC(12, 2)
)
as
begin
     insert into @projects1
     select * from projects
     where projectid in
     (select projectid from workson where employeeid =
@id);
     return
end;


select * from dbo.project_details(2);
```

Q2 Write a recursive scalar-valued function that takes input n to return the nth Fibonacci number.

```
create function fib
(@x integer)
returns integer
as
     begin
          declare @z integer
          if(@x = 0)
               set @z = 0;
          else if(@x=1)
               set @z = 1
          else
               set @z = dbo.fib(@x - 2)+ dbo.fib(@x -
1);
          return @z;
     end
```

```
select dbo.fib(11) as 'fibonacci';
```

Q3. Write an iterative table-valued function that takes input n to return n Fibonacci numbers sequentially in a table.

```
create function iterFib
(@x integer)
returns @fib table
(fibonacci integer)
as
     begin
          declare @temp1 integer, @temp2 integer;
          set @temp1 = 1
          set @temp2 = 1
          insert into @fib select 1;
          if(@x = 1)     return;
          else if (@x = 2)
               begin
                    insert into @fib select 1;
                    return;
               end
          else begin
          insert into @fib select 1;
          while(@x > 2)
               begin
                    insert into @fib select
@temp1+@temp2;
                    set @temp1 = @temp1 + @temp2;
                    set @temp2 = @temp1 - @temp2;
                    set @x = @x - 1;
               end
          end
     return;
     end

select * from dbo.iterFib(20);
```

Exercise 2

Q1	Write a T-SQL query to print all of the information about employees with last names that have exactly 8 characters and end in 'ware'.

```
DECLARE EMPLOYEE_CUR CURSOR LOCAL FOR SELECT EMPLOYEEID,
FIRSTNAME, LASTNAME, DEPTCODE, SALARY
FROM EMPLOYEES;
DECLARE @EMPLOYEEID VARCHAR(15),
@FIRSTNAME VARCHAR(30),
@LASTNAME VARCHAR(30),
@DEPTCODE VARCHAR(15),
@SALARY VARCHAR(15);

OPEN EMPLOYEE_CUR;
WHILE 1=1 BEGIN
FETCH NEXT FROM EMPLOYEE_CUR INTO  @EMPLOYEEID,
@FIRSTNAME, @LASTNAME, @DEPTCODE, @SALARY; IF
@@FETCH_STATUS <> 0 BREAK ; IF @LASTNAME LIKE '____WARE'
BEGIN
PRINT ISNULL(@EMPLOYEEID, '') + ' ' + ISNULL(@FIRSTNAME,
'') + ' ' + ISNULL(@LASTNAME, '') + ' ' +
ISNULL(@DEPTCODE, '') + ' ' + ISNULL(@SALARY, '');
END
END;
```

Q2	Write a T-SQL query to print all the ID and last name of all employees who work for department ACCNT and make less than $30,000.

```
DECLARE EMPLOYEE_CUR CURSOR LOCAL FOR SELECT EMPLOYEEID,
FIRSTNAME, LASTNAME, DEPTCODE, SALARY
FROM EMPLOYEES;
DECLARE @EMPLOYEEID VARCHAR(15),
@FIRSTNAME VARCHAR(30),
@LASTNAME VARCHAR(30),
@DEPTCODE VARCHAR(15),
@SALARY NUMERIC(9,2);

OPEN EMPLOYEE_CUR;
WHILE 1=1 BEGIN
```

```
FETCH NEXT FROM EMPLOYEE_CUR INTO  @EMPLOYEEID,
@FIRSTNAME, @LASTNAME, @DEPTCODE, @SALARY; IF
@@FETCH_STATUS <> 0 BREAK ; IF @DEPTCODE LIKE 'ACCNT' AND
@SALARY > 30000
BEGIN
PRINT @EMPLOYEEID + ' '+ @LASTNAME;
END   END;
```

Q3 Write a T-SQL query to print the names of all people who work in the Consulting department and who spend more than 20% of their time on the project with ID ADT4MFIA.

```
DECLARE @C_FIRST VARCHAR(30)
DECLARE @C_LAST VARCHAR(30)
DECLARE C_NAME CURSOR LOCAL FOR
SELECT FIRSTNAME,LASTNAME FROM EMPLOYEES
WHERE DEPTCODE IN( SELECT CODE FROM DEPARTMENTS WHERE
NAME='CONSULTING')AND EMPLOYEEID
IN( SELECT W1.EMPLOYEEID FROM WORKSON W1
WHERE  W1.PROJECTID='ADT4MFIA' AND W1.ASSIGNEDTIME/(SELECT
SUM(W.ASSIGNEDTIME) FROM WORKSON W WHERE W.EMPLOYEEID =
W1.EMPLOYEEID GROUP BY W.EMPLOYEEID)>.2);
BEGIN
OPEN C_NAME;
WHILE 1=1 BEGIN
FETCH C_NAME INTO @C_FIRST,@C_LAST; IF @@FETCH_STATUS <> 0
BREAK ;
PRINT ISNULL(@C_FIRST, '') + ' ' + ISNULL(@C_LAST, '');
END;
CLOSE C_NAME;
DEALLOCATE C_NAME;
END;
```

Q4 Write a T-SQL query to find the first and last name of all employees who are paid more than someone in the Accounting department.

```
DECLARE @C_FIRST VARCHAR(30);
DECLARE @C_LAST VARCHAR(30);
DECLARE C_NAME CURSOR LOCAL FOR
SELECT FIRSTNAME, LASTNAME FROM EMPLOYEES WHERE SALARY >
ANY(SELECT SALARY  FROM EMPLOYEES E, DEPARTMENTS D WHERE
E.DEPTCODE=D.CODE AND D.NAME='ACCOUNTING');
BEGIN OPEN C_NAME;
WHILE 1=1 BEGIN
```

```
FETCH C_NAME INTO @C_FIRST,@C_LAST;
PRINT ISNULL(@C_FIRST, '') + ' ' + ISNULL(@C_LAST, ''); IF
@@FETCH_STATUS <> 0 BREAK ; END;  CLOSE C_NAME;
DEALLOCATE C_NAME;
END;
```

Note: Solutions provided are for your own reference and may have other possible variations or interpretations. In case of any query, kindly contact your lab instructors.