|  |
| --- |
|  |
| Database Application Development |
| SWD317 AE1 |
|  |
| **Mark Tomlin** |
| **Student Number: Q09947311** |

|  |
| --- |
|  |

# Task 1a

## The Function Code

-- Title:                          Function 1

-- Purpose:                    To calculate and return the total number of different task-types involved in a given

project

-- Language:                  Oracle PL/SQL

-- Author:                      Mark Tomlin

-- Date:                          10th December 2013

-- Version:                     1.0

CREATE OR REPLACE FUNCTION tot\_task\_types (p project.project\_no%TYPE)

RETURN NUMBER IS

task\_count NUMBER;

BEGIN

SELECT COUNT(DISTINCT(t.task\_type\_no))

INTO task\_count

FROM project p,stage s,task t

WHERE s.stage\_id = t.stage\_id

AND p = s.project\_no;

RETURN task\_count;

END;

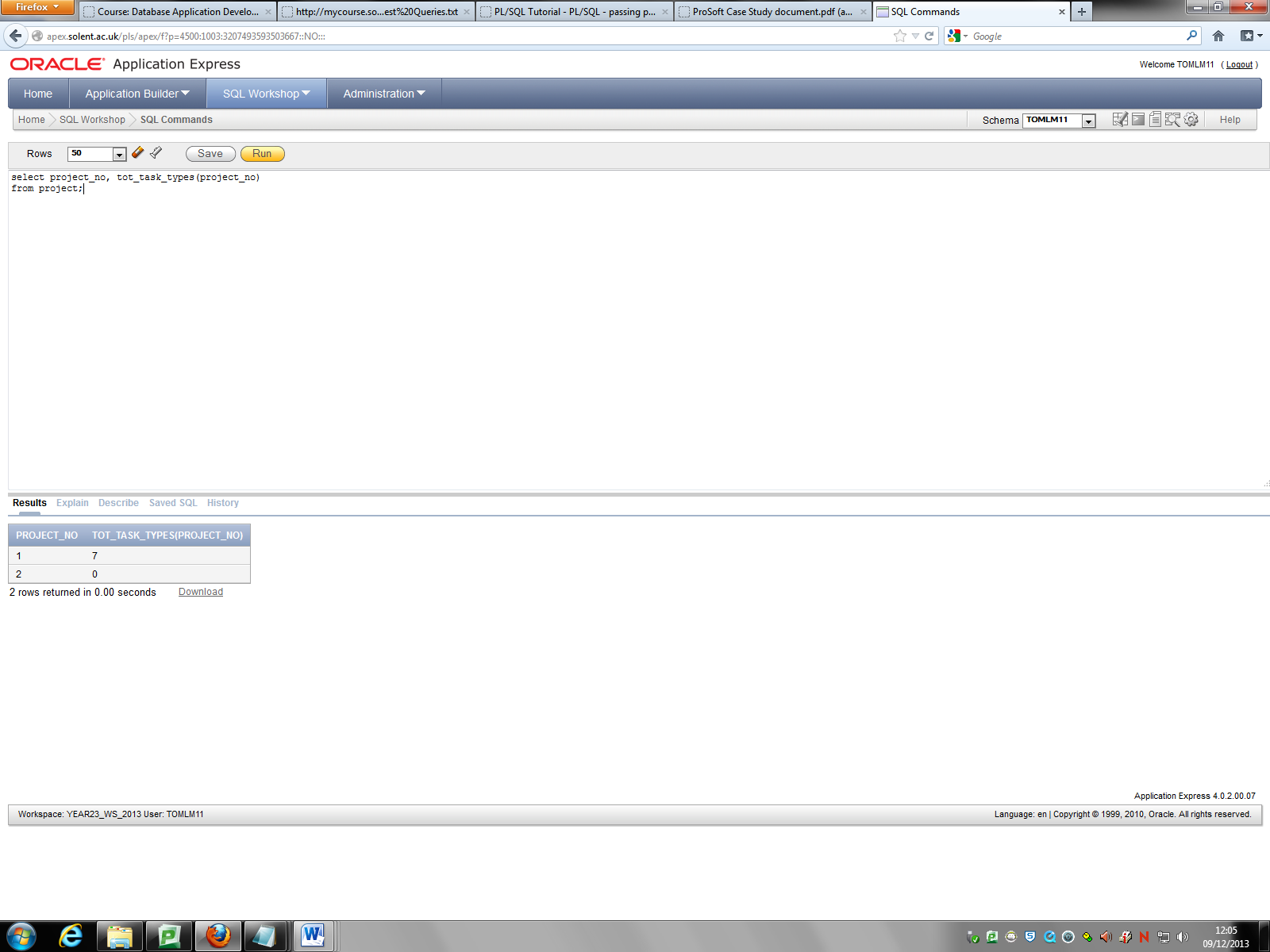
/

## Function Test Statement

SELECT project\_no, tot\_task\_types(project\_no)

FROM project;

## Test Screenshot



# Task 1b

## The Function Code

-- Title:                          Function 2

-- Purpose:                    To calculate and return the total number of estimated hours involved in a given stage

When it was first issued

-- Language:                  Oracle PL/SQL

-- Author:                      Mark Tomlin

-- Date:                          13th December 2013

-- Version:                     1.0

CREATE OR REPLACE FUNCTION tot\_first\_est\_hours (s stage.stage\_id%TYPE)

RETURN NUMBER IS

run\_tot\_hours NUMBER:=0;

temp\_val NUMBER;

earliest\_date DATE;

BEGIN

SELECT MIN(th.change\_date)

INTO earliest\_date

FROM task\_history th,task t,stage s

WHERE s = t.stage\_id

AND t.task\_id = th.task\_id;

FOR loop\_count IN (SELECT th.prev\_est\_hrs,t.estimated\_hours

FROM stage s,task t,task\_history th

WHERE s = t.stage\_id

AND t.task\_id = th.task\_id ) LOOP

IF COUNT(th.prev\_est\_hrs)>0 THEN

SELECT th.prev\_est\_hrs

INTO temp\_val

FROM task\_history th

WHERE th.change\_date = earliest\_date;

run\_tot\_hours:=run\_tot\_hours+temp\_val;

ELSE

SELECT t.estimated\_hours

INTO temp\_val

FROM task t,task\_history th

WHERE t.task\_id = th.task\_id;

run\_tot\_hours:=run\_tot\_hours+temp\_val;

END IF;

END LOOP;

RETURN run\_tot\_hours;

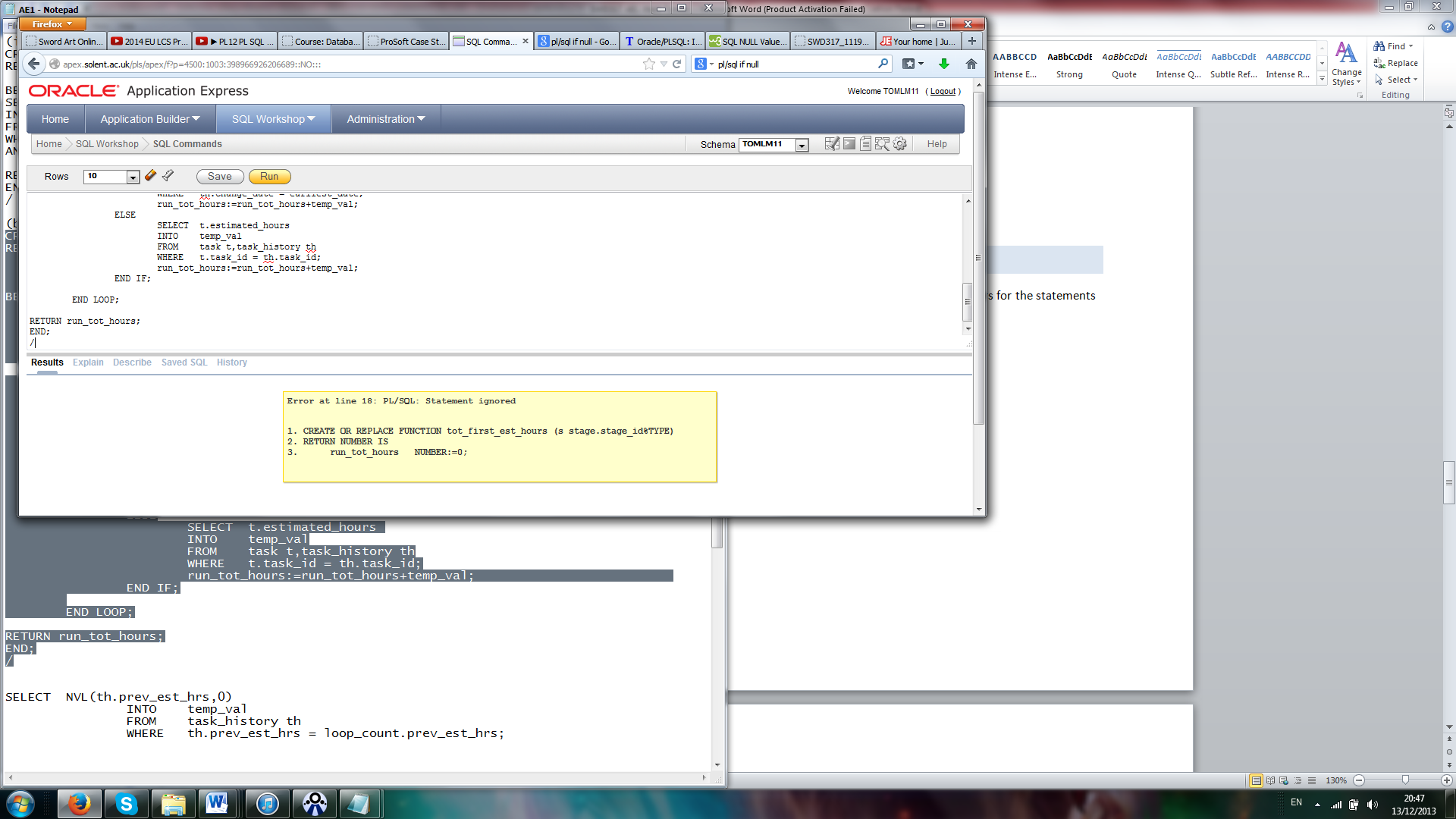
END;

/

## Testing

For this function I was able to understand the logic involved for it, however I kept getting errors for the statements I tried to run. I felt like my code needs a few adjustments to be functional but I have fell short of time for this.

Here is a screen shot of the error for this statement:



# Task 2

## The following requirments are fully supported by the given database design

**4.1 Maintaining Stages of a Project**

**4.2 Adding Tasks to a Stage**

**4.5 Employee Task Report**

**4.6 Monitoring Changes to Estimated Hours for a Task**

## The following are not supported by the given database design

**4.7 Recording of Hours Work Completed**

There is no table or data currently that records work hours actually done and also there is no data on whether or not the hours worked should be charged to the client.

**4.8 Weekly Timesheet Report**

There are no details of the work an employee has done in a given week.

**4.10 Management Reports**

There is no data on how many hours have been worked and whether or not they are chargeable.

**4.11 Invoices**

Can’t be produced because there is no data on how much work has been done. Also there is no data to hold previous invoices for new ones to get information about work done between the previous.

**4.12 Employee Experience Report**

There is currently no work done information stored for employees or records of any previous work they’ve done.

## The following are partially supported by the given database design

**4.3 Estimates**

Estimates can be calculated but a client ready document can‘t be produced because there are no client address details available.

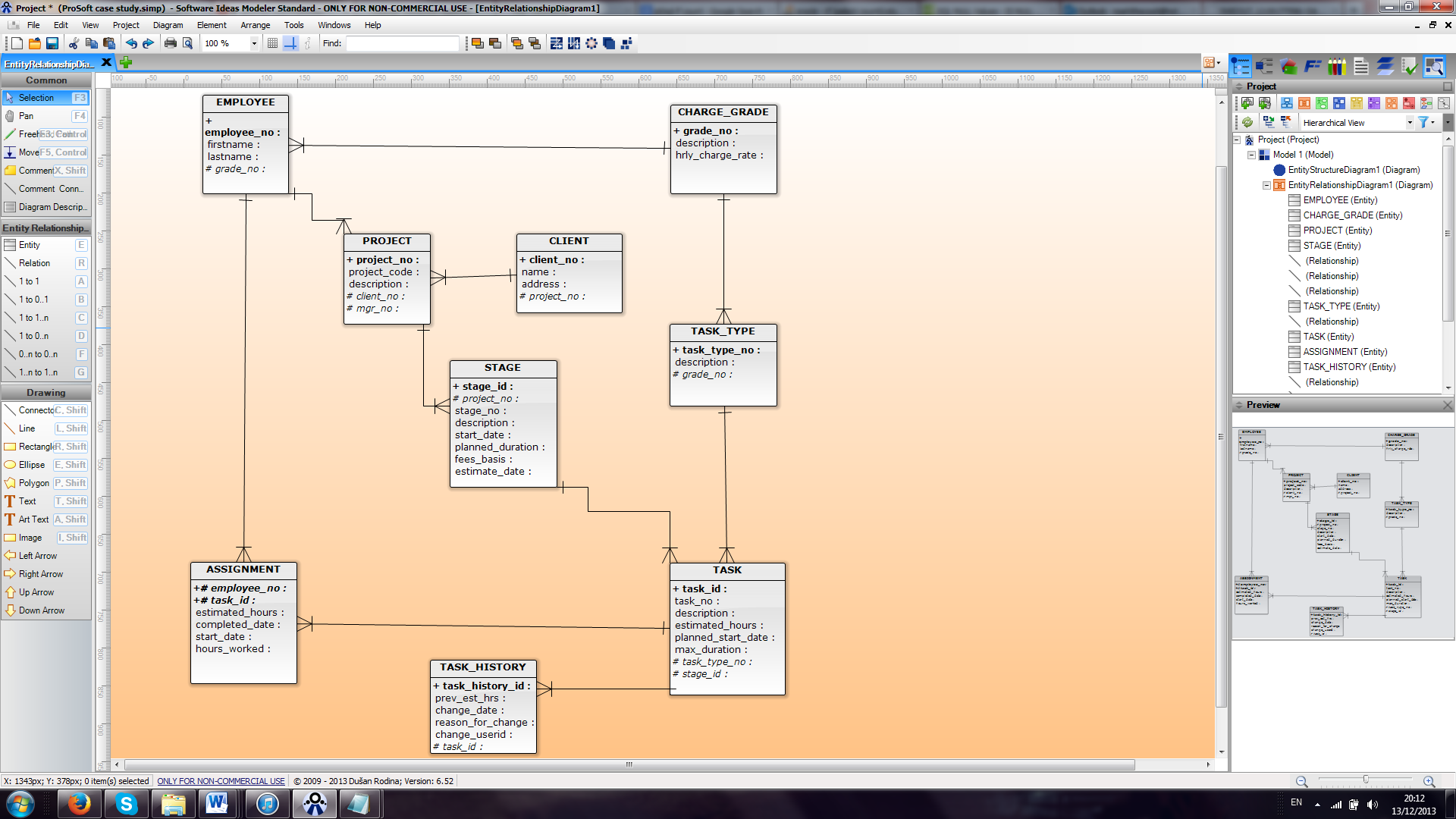
**4.4 Task Assignments**

The current database doesn’t limit the amount of assignments given to employees from a task.

**4.9 Project Report**

There is no way to get an accurate count of the number of hours worked to a date for a given task.

# Task 3



For this new entity relationship diagram (ERD) I have added a new table called ‘Client’. Having a client table will eliminate some problems for requirements 4.3 and 4.7

The second change I have done is extending the Assignment table by adding the data fields ‘start\_date’ and ‘hours\_worked’. I contemplated making a new table for ‘work\_done’ but it felt redundant because it would have to link with the Task and Employee tables anyway. So I felt extending the Assignment was the best way. This will help with ProSoft requirements 4.7,4.8,4.9,4.10,4.11 and 4.12