***Creating Views***

**Practice 11 Solutions**

1. Create a view called EMPLOYEES\_VU based on the employee numbers, employee names, and

department numbers from the EMPLOYEES table. Change the heading for the employee name to

EMPLOYEE.

**CREATE OR REPLACE VIEW employees\_vu AS**

**SELECT employee\_id, last\_name employee, department\_id**

**FROM employees;**

2. Display the contents of the EMPLOYEES\_VU view.

**SELECT \***

**FROM employees\_vu;**

3. Select the view name and text from the USER\_VIEWS data dictionary view.

**Note:** Another view already exists. The EMP\_DETAILS\_VIEW was created as part of your schema.

**Note:** To see more contents of a LONG column, use the iSQL\*Plus command SET LONG *n*, where

*n* is the value of the number of characters of the LONG column that you want to see.

**SET LONG 600**

**SELECT view\_name, text**

**FROM user**\_**views;**

4. Using your EMPLOYEES\_VU view, enter a query to display all employee names and department

numbers.

**SELECT employee, department\_id**

**FROM employees\_vu;**

5. Create a view named DEPT50 that contains the employee numbers, employee last names, and

department numbers for all employees in department 50. Label the view columns

EMPNO, EMPLOYEE, and DEPTNO. Do not allow an employee to be reassigned to another

department through the view.

**CREATE VIEW dept50 AS**

**SELECT employee\_id empno, last\_name employee,**

**department\_id deptno**

**FROM employees**

**WHERE department\_id = 50**

**WITH CHECK OPTION CONSTRAINT emp\_dept\_50;**

6. Display the structure and contents of the DEPT50 view.

**DESCRIBE dept50;**

**SELECT \***

**FROM dept50;**

7. Attempt to reassign Matos to department 80.

**UPDATE dept50**

**SET deptno = 80**

**WHERE employee = 'Matos';**

If you have time, complete the following exercise:

8. Create a view called SALARY\_VU based on the employee last names, department names, salaries,

and salary grades for all employees. Use the EMPLOYEES, DEPARTMENTS, and JOB\_GRADES

tables. Label the columns Employee, Department, Salary, and Grade, respectively.

**CREATE OR REPLACE VIEW salary\_vu**

**AS**

**SELECT e.last\_name "Employee",**

**d.department\_name "Department",**

**e.salary "Salary",**

**j.grade\_level "Grades"**

**FROM employees e,**

**departments d,**

**job\_grades j**

**WHERE e.department\_id = d.department\_id**

**AND e.salary BETWEEN j.lowest\_sal and j.highest\_sal;**