Home Work 4: Oracle SQL Class

If you see something in homework that needs more clarification please send me an email. Homework is meant to make you think, test examples of functions/concepts from the book, review the syntax, and practice.

# Research

In your own words give a brief definition of CRUD as it relates to databases.

In your own words give a brief definition of ACID as it relates to databases.

In your own words give a brief definition of Atomicity as it relates to databases.

In your own words give a brief definition of Drilling Down as it relates to databases.

In your own words give a brief definition of Granularity as it relates to databases.

# DECODE

Study closely (books, classroom files) the difference between decode and case. If decode will not work, explain why. Then try case. If case does not work explain why. (hint: use case not decode)

|  |  |
| --- | --- |
| **SCORES** | |
| **Student\_id** | **Score** |
| 2122 | 54 |
| 3211 | 92 |
| 411 | 80 |
| 2771 | 77 |
| 2098 | 64 |
| 7890 | 0 |

Create a table called scores.

Write a SQL statement using the decode function from Chapter 6 that gives a result set showing:

Sudent\_id

Score

Letter Grade

Where the letter grade is A if score 90-100; B if score 80-89;

C if score 70-79; D if score 60-69; INC if score 0-59;

# Transform null

Study carefully the three forms of null functions: NVL, NVL2, NULLIF.

Will nvl(shipdate, ‘Not Shipped’) work? If not why not. If yes use it.

Using Orders table in Books, write a SQL statement that returns “Not Shipped” if the shipdate is null. If none of the null functions work, what about decode or case?

hint:

select shipdate,

decode (

shipdate,

null , 'Not Shipped',

shipdate

)

# Transform date, number

Create a table called invoices using the appropriate data types:

inv\_id integer

inv\_date date

inv\_amt number (10,2)

acc\_no varchar2(25)

Copy the rows of data shown here into sql developer and place dingle ticks around each data value. Now write an insert statement that inserts each of the rows into the table in the correct data type.

Use:

to\_date to change string ‘3/15/2009’ to a date

to\_number to change string ‘147119’ to a number

|  |  |  |  |
| --- | --- | --- | --- |
| INVOICES | | | |
| Inv\_id | inv\_date | inv\_amt | Acct\_no |
| 701 | 3/15/2009 | 147119 | 0CODDA15 |
| 702 | 6/17/2010 | 275803 | CODDA12 |
| 703 | 10/18/2010 | 248414 | CODDA20 |
| 704 | 1/19/2009 | 169206 | CODDA18 |
| 705 | 9/18/2011 | 102680 | CODDA12 |
| 706 | 11/4/2010 | 179138 | CODDA17 |
| 707 | 12/13/2011 | 270723 | CODDA18 |
| 708 | 9/15/2010 | 130288 | CODDA13 |
| 709 | 3/21/2010 | 255003 | CODDA18 |
| 710 | 4/13/2009 | 254837 | CODDA19 |
| 711 | 6/30/2010 | 284695 | CODDA19 |
| 712 | 11/20/2010 | 297928 | CODDA19 |

# Case Statement

Use the Question 4 table invoices after inserting the data.

Create a SQL case statement that returns all table columns with one additional column that states

**Add Invoice to 2009** if the invoice date is in 2009

**Add Invoice to 2010** if the invoice date is in 2010

**Add Invoice to 2011** if the invoice date is in 2011

For row one the result set would look like:

Inv\_id inv\_date inv\_amt Acct\_no Instructions

701 3/15/2009 147119 0CODDA15 Add Invoice to 2009