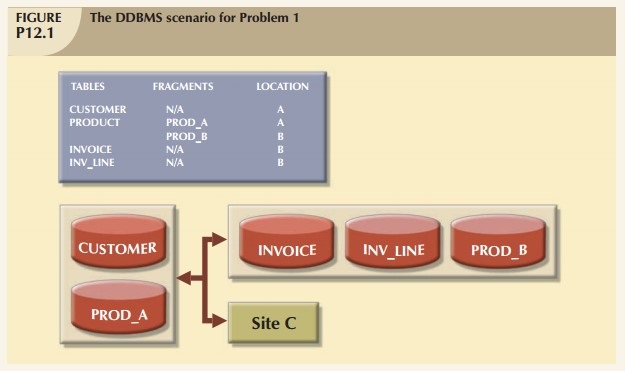
**Assignment-06**

**Distributed Database**

1. Given below is a Distributed database system. Specify the type of operation (remote request, remote transaction, distributed transaction, or distributed request) that the database must support to perform the following transactions. Also specify the reason in determining the type of database operation.



1. BEGIN;

SELECT prod\_code

FROM product

WHERE prod\_code = ‘A0555’;

COMMIT;

Ans: Distributed Request, it consists of 3 SQL statements in which, the second query accesses data at multiple sites. prod\_code A0555 refers to prod A or prod B in location A or location B.

1. BEGIN;

SELECT inv\_line\_num

FROM in\_line;

SELECT inv\_num

FROM invoice;

COMMIT;

Ans: Remote Transaction, among the four statements, both the SELECT queries are accessing data at single site (both in\_line and invoice are at location B)

1. BEGIN;

SELECT cus\_ID

FROM customer;

COMMIT;

Ans: Remote Transaction, the above SELECT query is accessing data from Customer table which is at single site (location A).

1. BEGIN;

UPDATE product

SET prod\_qoh = prod\_qoh – 1

WHERE prod\_num = ‘1111’;

COMMIT;

Ans: Distributed Request, the above UPDATE query requires data from multiple sites (Location A & Location B) as it has to locate prod\_num 1111 which maybe in prod A or prod B.

1. BEGIN;

INSERT INTO invoice(inv\_num, cus\_num, inv\_date, inv\_total)

VALUES (‘999333’, ‘10000’, ‘15-MAR-2014’, 250);

INSERT INTO inv\_line(inv\_num, prod\_num, line\_price)

VALUES (‘999333’, ‘1001’, 200);

COMMIT;

Ans: Remote Transaction, both the above insert statements are inserting values to tables at Location B (invoice and in\_line).

1. BEGIN;

UPDATE product

SET prod\_qoh = prod\_qoh – 1

WHERE prod\_num = ‘1111’;

UPDATE customer

SET cus\_bal = cus\_bal + 250

WHERE cus\_num = ‘10101’;

COMMIT;

Ans: Distributed Request, the whole transaction refers multiple locations as well as the individual queries are also referring multiple locations.

1. BEGIN;

SELECT cus\_name

FROM customer;

SELECT inv\_num

FROM invoice;

COMMIT;

Ans: Distributed Transaction, the complete transaction requires data from multiple locations. However, each SELECT query is accessing data from one location only.

1. Suppose we have the following product data for a store in the database for a company with 3 warehouses.

An example table is as shown below.

**PART**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PartNum** | **Description** | **On Hand** | **Class** | **Warehouse** | **Price** |
| AT94 | Iron | 50 | Home appliances | 3 | $524.95 |
| BV06 | Home Gym | 45 | Gym products | 2 | $ 794.95 |
| CD52 | Microwave Oven | 32 | Home appliances | 1 | $165.00 |
| DL71 | Cordless Drill | 21 | Home appliances | 3 | $129.95 |
| DR93 | Gas Range | 8 | Mechanical | 2 | $495.00 |
| DW11 | Washer | 12 | Mechanical | 3 | $399.00 |
| FD21 | Stand Mixer | 22 | Home appliances | 3 | $159.00 |
| KL62 | Dryer | 12 | Home appliances | 1 | $39.99 |
| KT03 | Dish washer | 8 | Home appliances | 3 | $595.00 |
| KV29 | Treadmill | 9 | Gym products | 2 | $1,390.00 |

1. Show how to fragment the above data horizontally by warehouse number. Call the fragments PART\_1, PART\_2 and PART\_3, and show the contents of each fragment.

PART\_1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PartNum** | **Description** | **On Hand** | **Class** | **Warehouse** | **Price** |
| CD52 | Microwave Oven | 32 | Home appliances | 1 | $165.00 |
| KL62 | Dryer | 12 | Home appliances | 1 | $39.99 |

PART\_2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PartNum** | **Description** | **On Hand** | **Class** | **Warehouse** | **Price** |
| BV06 | Home Gym | 45 | Gym products | 2 | $ 794.95 |
| DR93 | Gas Range | 8 | Mechanical | 2 | $495.00 |
| KV29 | Treadmill | 9 | Gym products | 2 | $1,390.00 |

PART\_3

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PartNum** | **Description** | **On Hand** | **Class** | **Warehouse** | **Price** |
| AT94 | Iron | 50 | Home appliances | 3 | $524.95 |
| DL71 | Cordless Drill | 21 | Home appliances | 3 | $129.95 |
| DW11 | Washer | 12 | Mechanical | 3 | $399.00 |
| FD21 | Stand Mixer | 22 | Home appliances | 3 | $159.00 |
| KT03 | Dish washer | 8 | Home appliances | 3 | $595.00 |

1. Next, you will fragment each fragment horizontally by class. Name each of the new fragments using the existing fragment name, followed by an underscore, followed by HOME\_APPLIANCES, GYM\_PRODUCTS and MECHANICAL.

PART\_1\_HOME\_APPLIANCES

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PartNum** | **Description** | **On Hand** | **Class** | **Warehouse** | **Price** |
| CD52 | Microwave Oven | 32 | Home appliances | 1 | $165.00 |
| KL62 | Dryer | 12 | Home appliances | 1 | $39.99 |

PART\_3\_HOME\_APPLIANCES

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PartNum** | **Description** | **On Hand** | **Class** | **Warehouse** | **Price** |
| AT94 | Iron | 50 | Home appliances | 3 | $524.95 |
| DL71 | Cordless Drill | 21 | Home appliances | 3 | $129.95 |
| FD21 | Stand Mixer | 22 | Home appliances | 3 | $159.00 |
| KT03 | Dish washer | 8 | Home appliances | 3 | $595.00 |

PART\_2\_ GYM\_PRODUCTS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PartNum** | **Description** | **On Hand** | **Class** | **Warehouse** | **Price** |
| BV06 | Home Gym | 45 | Gym products | 2 | $ 794.95 |
| KV29 | Treadmill | 9 | Gym products | 2 | $1,390.00 |

PART\_2\_ MECHANICAL

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PartNum** | **Description** | **On Hand** | **Class** | **Warehouse** | **Price** |
| DR93 | Gas Range | 8 | Mechanical | 2 | $495.00 |

PART\_3\_MECHANICAL

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PartNum** | **Description** | **On Hand** | **Class** | **Warehouse** | **Price** |
| DW11 | Washer | 12 | Mechanical | 3 | $399.00 |

1. Consider the following situation, adapted from Date. A simplified procurement (relational) database has the following three relations:

SUPPLIER (SUPPLIER\_NUMBER, CITY) 40,000 records stored in Detroit

PART (PART\_NUMBER, COLOR) 650,000 records stored in Chicago

SHIPMENT (SUPPLIER\_NUMBER, PART\_NUMBER) 1,500,000 records stored in Detroit

A query is made (in SQL) to list the supplier numbers for Cleveland suppliers of red parts:

SELECT SUPPLIER.SUPPLIER\_NUMBER

FROM SUPPLIER, SHIPMENT, PART

WHERE SUPPLIER.CITY = ‘Cleveland’

AND SHIPMENT.PART\_NUMBER = PART.PART\_NUMBER

AND SHIPMENT.SUPPLIER\_NUMBER = SUPPLIER.SUPPLIER\_NUMBER

AND PART.COLOR = ‘RED’;

Each record in each relation is 350 characters long. There are thirty red parts, a history of 200,000 shipments from Cleveland, and a negligible query computation time compared with communication time. Also, there is a very old communication system with a very slow data transmission time of 4,000 characters per second and two seconds access delay to send a message from one node to another. Consider operating time to execute the query as three seconds. Assume the time taken to transfer the query results as negligible.

Determine the time to process this remote query assuming the following strategy:

**Move SUPPLIER relation to Chicago; then move SHIPMENT relation to Chicago; process whole query at Chicago computer.**

Answer should be expressed in hours, with one decimal place.

Ans: 37.4 hours.

1,500,000 records (SHIPMENT) + 40,000 records (SUPPLIER) = 1,540,000 records.

1,540,000 records \* 350 characters per record = 539,000,000 characters.

539,000,000 characters / 4,000 characters per second = 134,750 seconds.

4 seconds access delay (2 sec for each query) and 6 seconds operating time (3 sec for each query).

Total Time = 134,750 seconds + 10 seconds =134,760 seconds.

**Note:** To submit the assignment, rename the solution document as LastName\_Assignment06 (.docx, .doc). Highlight your solutions in yellow back ground.