Databases

Worksheets

Assignment Project Exam Help

Peter McBrien

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

branch		
sortcode	bname	cash
56	'Wimbledon'	94340.45
34	'Goodge St'	8900.67
67	'Strand'	34005.00

movement			
mid	no	amount	tdate
1000	100	2300.00	5/1/1999
1001	101	4000.00	5/1/1999
1002	100	-223.45	8/1/1999
1004	107	-100.00	11/1/1999
1005	103	145.50	12/1/1999
1006	100	10.23	15/1/1999
1007	107	345.56	15/1/1999
1008	101	1230.00	15/1/1999
1009	119	5600.00	18/1/1999

account				
no	type	cname	rate?	sortcode
100	'current'	'McBrien, P.'	NULL	67
101	'deposit'	'McBrien, P.'	5.25	67
103	'current'	'Boyd, M.'	NULL	34
107	'current'	'Poulovassilis, A.'	NULL	56
119	'deposit'	'Poulovassilis, A.'	5.50	56
125	'current'	'Bailey, J.'	NULL	56

key branch(sortcode)
key branch(bname)
key movement(mid)
key account(no)
movement(no) $\stackrel{fk}{\Rightarrow}$ account(no)
account(sortcode) $\stackrel{fk}{\Rightarrow}$ branch(sortcode)

1. What is the result of the RA query $\pi_{\text{cname,sortcode}}\ \sigma_{\text{type}=\text{'deposit'}}$ account

2. What is the result of the RA query

$\pi_{\text{account.no,cname,mid}}\ \sigma_{\text{account.no}=\text{movement.no} \wedge \text{movement.amount}<0}$ account $\times$ movement

3. Write an RA query to return the scheme (cnam... mers that
have made a withdrawal from an account

4. Write an RA query to return the scheme (cname,bname) that lists the cname of all deposit
account holders, together with bname where the account is held.

5. Write an RA query to return the scheme (sortcode) that lists the sortcodes of branches that
either have less than £10,000 of cash, or that hold deposit accounts.

6. Write an RA query to return the scheme (sortcode) that lists those branches where no deposit
account is held

DB Worksheet 2: Derived Relational Algebra Operators

1. What is the result of the RA query $\pi_{\mathsf{bname,cname}}(\mathsf{branch} \bowtie \mathsf{account})$

2. What is the result of the RA query $\pi_{\mathsf{bname,cname}}(\mathsf{branch} \bowtie \mathsf{account} \bowtie \mathsf{movement})$

3. What is the result of the RA query $\pi_{\mathsf{sortcode,type}} \mathsf{account} \div \pi_{\mathsf{sortcode}} \mathsf{branch}$

4. Write an RA query returning the scheme (bname,mid) that lists branch names, together with mids that have occurred at that branch.

5. Write an RA que   ts rows of all branches
   that have at lea

6. Write an RA query returning the scheme (sortc   anches that
   have both have less than £100,000 of cash, and that hol

Consider the following queries over the `bank_branch` database.

1. Put a tick or cross to indicate if each equivalence holds:

Equivalence	Correct?
$\pi_{\text{no,type}}\, \sigma_{\text{type='deposit'}}\, \text{account} \equiv \sigma_{\text{type='deposit'}}\, \pi_{\text{no,type}}\, \text{account}$	
$\pi_{\text{no,type}}\, \sigma_{\text{type='deposit'}}\, \text{account} \equiv \sigma_{\text{type='deposit'}}\, \pi_{\text{type,no}}\, \text{account}$	
$\pi_{\text{type}}\, \sigma_{\text{type='deposit'}}\, \text{account} \equiv \sigma_{\text{type='deposit'}}\, \pi_{\text{no}}\, \text{account}$	
$\sigma_{\text{sortcode}=56}(\text{account} \times \text{movement}) \equiv \sigma_{\text{sortcode}=56}\, \text{account} \times \text{movement}$	
$\pi_{\text{sortcode}}(\text{account} \times \text{movement}) \equiv \pi_{\text{sortcode}}\, \text{account} \times \text{movement}$	
$\pi_{\text{no,type}}\, \sigma_{\text{type='deposit'}}\, \text{account} \equiv \pi_{\text{no,type}}\, \sigma_{\text{type}<>\text{'current'}}\, \text{account}$	

2. Simplify the following RA query to contain as few RA operators as possible

   $$\pi_{\text{no,type}}\, \sigma_{\text{sortcode}=56}\, \pi_{\text{no,type,sortcode}}\, \sigma_{\text{type='deposit'}}\, \text{account}$$

3. Rewrite the following RA query into the form $\pi_{...}\, \sigma_{...}\, R \times S$, the general form of which we will call a **project select product** (**PSP**) query.

   $$\sigma_{\text{account.no}=\text{movement.no}}(\pi_{\text{no,cname}}\, \text{account} \times \pi_{\text{mid,no}}\, \sigma_{\text{amount}>1000}\, \text{movement})$$

4. Rewrite the following RA to be a union between two PSP queries

   $$\sigma_{\text{account.no}=m}$$
   $$(\pi_{\text{mid}} \ldots)$$

5. Rewrite the following query to an equivalent form that minim... ber of tuples, and the number of attributes within those tuples, that are handled by the $\times$ operator.

   $$\pi_{\text{no,cname,tdate}}\, \sigma_{\text{amount}<0 \wedge \text{account.no}=\text{movement.no}}(\text{account} \times \text{movement})$$

branch		
sortcode	bname	cash
56	'Wimbledon'	94340.45
34	'Goodge St'	8900.67
67	'Strand'	34005.00

account				
no	type	cname	rate?	sortcode
100	'current'	'McBrien, P.'	NULL	67
101	'deposit'	'McBrien, P.'	5.25	67
103	'current'	'Boyd, M.'	NULL	34
107	'current'	'Poulovassilis, A.'	NULL	56
119	'deposit'	'Poulovassilis, A.'	5.50	56
125	'current'	'Bailey, J.'	NULL	56

movement			
mid	no	amount	tdate
1000	100	2300.00	5/1/1999
1001	101	4000.00	5/1/1999
1002	100	-223.45	8/1/1999
1004	107	-100.00	11/1/1999
1005	103	145.50	12/1/1999
1006	100	10.23	15/1/1999
1007	107	345.56	15/1/1999
1008	101	1230.00	15/1/1999
1009	119	5600.00	18/1/1999

key branch(sortcode)
key branch(bname)
key movement(mid)
key account(no)
movement(no) $\overset{fk}{\Rightarrow}$ account(no)
account(sortcode) $\overset{fk}{\Rightarrow}$ branch(sortcode)

1. Write a Datalog query returning the scheme (cname,bname) listing the cname of all deposit account holders, together with bname where the account is held.

2. Write a Datalog query returning the scheme (no,bname) listing the account number and branch name of all accounts that have no movements recorded for the account.

3. Write a Datalog query that returns the scheme   nt' numbers, defined as those accounts that have made a withdrawal or a deposit.

4. List what the following Datalog query returns, and explain its semantics.

query(CName) :-
        account(_, _, , CName, _, _),
        ¬sub_query(CName).
sub_query(CName) :-
        account(_, _, CName, _, _),
        account(_, Type, _, _, _),
        ¬account(_, Type, CName, _, _).

Consider the following fragment of the **bank_branch** database:

movement			
mid	no	amount	tdate
1000	100	2300.00	5/1/1999
1001	101	4000.00	5/1/1999
1002	100	-223.45	8/1/1999
1004	107	-100.00	11/1/1999
1005	103	145.50	12/1/1999
1006	100	10.23	15/1/1999
1007	107	345.56	15/1/1999
1008	101	1230.00	15/1/1999
1009	119	5600.00	18/1/1999

account				
no	type	cname	rate?	sortcode
100	'current'	'McBrien, P.'	NULL	67
101	'deposit'	'McBrien, P.'	5.25	67
103	'current'	'Boyd, M.'	NULL	34
107	'current'	'Poulovassilis, A.'	NULL	56
119	'deposit'	'Poulovassilis, A.'	5.50	56
125	'current'	'Bailey, J.'	NULL	56

$$movement(no) \overset{fk}{\Rightarrow} account(no)$$

1. Write the RA expression using project, select and product operators equivalent to the SQL query below.
   SELECT   account.cname, movement.amount
   FROM   account JOIN movement ON account.no=movement.no
   WHERE   account.rate>2.0
   AND   movement.amount>1000

2. Modify your RA expression to use any other RA operators you have been shown to write a mimimal RA expression (*i.e.* one that uses as few operators as possible).

3. Write a minimal SQL ... pression
   π_{no} movement $- \pi_{no}$ account

4. Write a minimal SQL query equivalent to the RA expression $\pi_{no,type}$ account

5. Write a minimal SQL query equivalent to the RA expression π_{type} account

6. Write a minimal SQL query equivalent to the RA expression $\pi_{type,cname}$ account

7. Write a minimal SQL query equivalent to the RA expresss account $\ltimes$ movement

The following questions again use the **bank_branch** database:

branch		
sortcode	bname	cash
56	'Wimbledon'	94340.45
34	'Goodge St'	8900.67
67	'Strand'	34005.00

movement			
mid	no	amount	tdate
1000	100	2300.00	5/1/1999
1001	101	4000.00	5/1/1999
1002	100	-223.45	8/1/1999
1004	107	-100.00	11/1/1999
1005	103	145.50	12/1/1999
1006	100	10.23	15/1/1999
1007	107	345.56	15/1/1999
1008	101	1230.00	15/1/1999
1009	119	5600.00	18/1/1999

account				
no	type	cname	rate?	sortcode
100	'current'	'McBrien, P.'	NULL	67
101	'deposit'	'McBrien, P.'	5.25	67
103	'current'	'Boyd, M.'	NULL	34
107	'current'	'Poulovassilis, A.'	NULL	56
119	'deposit'	'Poulovassilis, A.'	5.50	56
125	'current'	'Bailey, J.'	NULL	56

key branch(sortcode)
key branch(bname)
key movement(mid)
key account(no)
movement(no) $\overset{fk}{\Rightarrow}$ account(no)
account(sortcode) $\overset{fk}{\Rightarrow}$ branch(sortcode)

1. Write an SQL query returning the scheme (mid,no,amount,tdate) listing all details of movements for accounts 100,101,103 and 107.

Assignment Project Exam Help

2. Write an SQL quer de of all branches without any d https://eduassistpro.github.io/

Add WeChat edu_assist_pro

3. Write an SQL query without using any negation (*i.e.* without the use of NOT or EXCEPT) returning the scheme (no) listing accounts with no movements on or before the 11-Jan-1999.

4. Write an SQL query returning the scheme (cname) listing customers that have every type of account that appears in account.

In a modified version of the **bank_branch** database, called **bank_branch_null**, there are the following two tables:

movement			
<u>mid</u>	no	amount	tdate
0999	119	45.00	null
1000	100	2300.00	5/1/1999
1001	101	4000.00	5/1/1999
1002	100	-223.45	8/1/1999
1004	107	-100.00	11/1/1999
1005	103	145.50	12/1/1999
1006	100	10.23	15/1/1999
1008	101	1230.00	15/1/1999
1009	119	5600.00	18/1/1999
1010	100	null	20/1/1999
1011	null	null	20/1/1999
1012	null	600.00	20/1/1999
1013	null	-46.00	20/1/1999

account				
<u>no</u>	type	cname	rate?	sortcode
100	'current'	'McBrien, P.'	null	67
101	'deposit'	'McBrien, P.'	5.25	67
119	'deposit'	'Poulovassilis, A.'	5.50	56
125	'current'	'Bailey, J.'	null	56

1. Write an SQL query returning the scheme (mid) to find movements known not to have occurred on 5/1/1999

Assignment Project Exam Help

2. Write an SQL que   ave or might have
   occurred on 5/ https://eduassistpro.github.io/

Add WeChat edu_assist_pro

3. Write an SQL query returning the scheme (no,mid) that lists account numbers, and any movements mids that have or might have occurred on that account

4. What is the result of the following query:

```
SELECT    account.no
FROM      account
WHERE     account.no NOT IN (SELECT movement.no FROM movement)
```

5. Write an SQL query returning the scheme (no) that lists those account numbers that might not have any movements.

The following question uses the **bank_branch_null** database:

movement			
mid	no	amount	tdate
0999	119	45.00	null
1000	100	2300.00	5/1/1999
1001	101	4000.00	5/1/1999
1002	100	-223.45	8/1/1999
1004	107	-100.00	11/1/1999
1005	103	145.50	12/1/1999
1006	100	10.23	15/1/1999
1008	101	1230.00	15/1/1999
1009	119	5600.00	18/1/1999
1010	100	null	20/1/1999
1011	null	null	20/1/1999
1012	null	600.00	20/1/1999
1013	null	-46.00	20/1/1999

account				
no	type	cname	rate?	sortcode
100	'current'	'McBrien, P.'	null	67
101	'deposit'	'McBrien, P.'	5.25	67
119	'deposit'	'Poulovassilis, A.'	5.50	56
125	'current'	'Bailey, J.'	null	56

1. What is the result of

   SELECT account.no, movement.mid
   FROM account NATURAL LEFT JOIN movement

   Assignment Project Exam Help

2. What is the result of https://eduassistpro.github.io/

   SELECT account.no, movement.mid
   FROM account NATURAL FULL OUTER JOIN

   Add WeChat edu_assist_pro

3. Write an SQL query returning the scheme (no,cname,mid) that lists all account numbers in the database (including just those in movement), and lists any known mid or cname for each account. The result for the current data should be:

no	cname	mid
100	McBrien, P.	1000
100	McBrien, P.	1006
100	McBrien, P.	1010
100	McBrien, P.	1002
101	McBrien, P.	1008
101	McBrien, P.	1001
103	null	1005
107	null	1004
119	Poulovassilis, A.	999
119	Poulovassilis, A.	1009
125	Bailey, J.	null

The following questions should be written to run on the **bank_branch** database listed below.

branch		
sortcode	bname	cash
56	'Wimbledon'	94340.45
34	'Goodge St'	8900.67
67	'Strand'	34005.00

movement			
mid	no	amount	tdate
1000	100	2300.00	5/1/1999
1001	101	4000.00	5/1/1999
1002	100	-223.45	8/1/1999
1004	107	-100.00	11/1/1999
1005	103	145.50	12/1/1999
1006	100	10.23	15/1/1999
1007	107	345.56	15/1/1999
1008	101	1230.00	15/1/1999
1009	119	5600.00	18/1/1999

account				
no	type	cname	rate?	sortcode
100	'current'	'McBrien, P.'	NULL	67
101	'deposit'	'McBrien, P.'	5.25	67
103	'current'	'Boyd, M.'	NULL	34
107	'current'	'Poulovassilis, A.'	NULL	56
119	'deposit'	'Poulovassilis, A.'	5.50	56
125	'current'	'Bailey, J.'	NULL	56

key branch(sortcode)
key branch(bname)
key movement(mid)
key account(no)
movement(no) $\overset{fk}{\Rightarrow}$ account(no)
account(sortcode) $\overset{fk}{\Rightarrow}$ branch(sortcode)

1. Write an SQL query returning the scheme (no,balance,avg_trans) that lists for each account that has transactions the account no, the balance (computed as the sum of the movements for the account), and the average value of transactions on that account.

Assignment Project Exam Help

https://eduassistpro.github.io/

2. Alter the query for (1) so that it includes the customer name of e   t, and includes *all* accounts held at the bank in the listing of balances, even if the account has no movements.

Add WeChat edu_assist_pro

3. Write an SQL query returning the scheme (cname,current_balance,deposit_balance) that lists one row for each customer (*i.e.* each distinct cname), with a column for the net balance of all current accounts held by the customer, and a column for the net balance of all deposit accounts held by the customer.

4. Write an SQL query returning the scheme (no,cname,type,pc_cust_funds,pc_type_funds) that lists one row for each account, and for each account, lists the no, cname and type of the account, and in pc_cust_funds the percentage of the customer funds held in the account, and in pc_type_funds the percentage of the total funds in this particular type of account. For the current data this should result in:

no	cname	type	pc_cust_funds	pc_type_funds
100	McBrien, P.	current	28.52	84.22
101	McBrien, P.	deposit	71.48	48.29
103	Boyd, M.	current	100.00	5.87
107	Poulovassilis, A.	current	4.20	9.91
119	Poulovassilis, A.	deposit	95.80	51.71
125	Bailey, J.	current	NULL	0.00

The following text gives a description of a UoD, which you have been asked to build a relational database that stores data associated with the UoD.

The payroll system for *BIG Inc* records the salaries, status, joining date, name, and payroll number for all of the corporations 30,000 employees. Each employee works for one division, and each division has an account number for paying its staff. We identify divisions by their name, and record the address where the division's HQ is located.

For employees sent abroad by *BIG Inc*, we record the address, country and telephone number of the foreign tax office that will handle the employee. It is assumed that each country has one central tax office that we have to deal with. All other employees have their tax affairs dealt with by the Inland Revenue.

Draw an ER$^{\mathcal{KLMOS}}$ schema of the UoD

The following text gives a description of a UoD, which you have been asked to build a relational database that stores data associated with the UoD.

The customer and supplier database of *Big Inc* will hold all accounts of the company, divided into customer accounts and supplier accounts. All accounts have an account number, and one account manager assigned from the company's staff. *Big Inc* identifies staff by a sid, and records the staff member's name and room. The account managers have a limit on the number of accounts they can manage. Only certain staff members are permitted to be account managers.

For customer accounts we need to record a credit limit on the balance of the account, and the telephone number of the accounts department at the customer.

For supplier accounts we need to record which *Big Inc* products are supplied, and at what price.

Big Inc products are identified by the company standard part_no and all have a description. For some we record the colour. Some products have a record of the components, each component identified by a combination of part_no and component number, and again each has a description. Some products do not have a supplier.

Big Inc has purchased a copy of the Post Office address file, and associates every account to an address from this file. The address data includes street number, street name, town, county and post code, and uses a combination of street number and post code as a key.

Assignment Project Exam Help

Draw an ER$^{\mathcal{ADH}}$

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

Suppose a relation $R(A, B, C, D, E, F, G, H)$ has the FDs

$$S = \{AB \to DEH, BEF \to A, FGH \to C, D \to EG, EG \to BF, F \to BH\}$$

1. Rewrite S to an equivalent set of FDs which only have a single attribute on the RHS of each FD.

2. Consider each FD $X \to A$, and for each $B \in X$, consider if $X \to B$ from the other FDs. If so, replace $X \to A$ by $(X - B) \to A$ in S.

3. Consider each FD $X \to A$, and compute X^+ without using $X \to A$. If $A \subseteq X^+$, delete $X \to A$ since it is rundundant. This will give a minimal cover S_c of S.

4. Justify what are the minimal candidate keys of R constrained by S_c

1. Suppose relation $R(A, B, C, D, E)$ has the FDs $S = \{AB \to C, C \to DE, E \to A\}$. For each of the following decompositions, determine if the decomposition if lossless, and if not lossless, illustrate a dataset for R that fails to decompose in a lossless manner over the relations.

   (a) $R_1(A, B, C), R_2(C, D, E)$

   (b) $R_1(A, B, C), R_2(C, D), R_3(D, E)$

Assignment Project Exam Help

2. Suppose relatio $\to E, A \to F\}$. Give
   a lossless deco
   https://eduassistpro.github.io/

3. Suppose relation $R(A, B, C, D, E, F)$ has the FDs Add WeChat edu_assist_pro $E, F \to A\}$. Give
   a lossless decomposition of R into three relations

Suppose a relation $R(A, B, C, D, E, F, G, H)$ has the FDs

$$S_c = \{AB \to D, EF \to A, FG \to C, D \to EG, EG \to F, F \to BH\}$$

1. Decompose the relation into 3NF

2. Decompose the relation into BCNF

3. Determine if your decompositions in (1) and (2) preserve FDs, and if they do not, suggest how to amend your schema to preserve FDs.

BEGIN TRANSACTION rental_charge
    UPDATE directory
    SET charge=charge+17
COMMIT TRANSACTION rental_charge

BEGIN TRANSACTION total_charge
    SELECT SUM(charge)
    FROM directory
COMMIT TRANSACTION total_charge

BEGIN TRANSACTION transfer_charge
    UPDATE directory
    FROM charge=charge+100
    WHERE telephone=1000

    UPDATE directory
    SET charge=charge-100
    WHERE telephone=1002
COMMIT TRANSACTION transfer_charge

directory		
telephone	name	charge
1000	Adams	10.00
1001	Jones	120.25
1002	Black	344.00

rental_charge $H_1 = r_1[d_{1000}], w_1[d_{1000}], r_1[d_{1001}], w_1[d_{1001}], r_1[d_{1002}], w_1[d_{1002}]$

transfer_charge $H_2 = r_2[d_{1000}], w_2[d_{1000}], r_2[d_{1002}], w_2[d_{1002}]$

total_charge $H_3 = r_3[d_{1000}], r_3[d_{1001}], r_3[d_{1002}]$

For each of the following histories, identify if they are a concurrent execution of some pair of the above histories, and if so, then determine if any of the following three anomalies has occurred:

A lost update

B inconsistent an

C dirty read

1. $r_3[d_{1000}], r_1[d_{1000}], w_1[d_{1000}], r_1[d_{1001}], w_1[d_{1001}], r_1[d_{1002}], w_1[d_{1002}], c_1, r_3[d_{1001}], r_3[d_{1002}], c_3$

2. $r_1[d_{1000}], r_1[d_{1001}], r_1[d_{1002}], r_2[d_{1000}], w_2[d_{1000}], r_2[d_{1002}], w_2[d_{1002}], w_1[d_{1000}], w_1[d_{1001}], w_1[d_{1002}], c_1, c_2$

3. $r_1[d_{1000}], r_2[d_{1000}], w_1[d_{1000}], r_1[d_{1001}], w_1[d_{1001}], r_1[d_{1002}], w_2[d_{1000}], r_2[d_{1002}], w_1[d_{1002}], w_2[d_{1002}], c_1, c_2$

4. $r_3[d_{1000}], r_3[d_{1001}], r_2[d_{1000}], w_2[d_{1000}], r_3[d_{1002}], r_2[d_{1002}], w_2[d_{1002}], c_2, a_3$

5. $r_2[d_{1000}], w_2[d_{1000}], r_1[d_{1000}], r_2[d_{1002}], w_2[d_{1002}], a_2, w_1[d_{1000}], r_1[d_{1001}], w_1[d_{1001}], r_1[d_{1002}], w_1[d_{1002}], c_1$

$H_1 = r_1[o_1], w_1[o_1], w_1[o_2], w_1[o_3], c_1$

$H_2 = r_2[o_2], w_2[o_2], w_2[o_1], c_2$

$H_3 = r_3[o_1], w_3[o_1], w_3[o_2], c_3$

$H_x = r_1[o_1], w_1[o_1], r_2[o_2], w_2[o_2], w_2[o_1], c_2, w_1[o_2], r_3[o_1], w_3[o_1], w_3[o_2], c_3, w_1[o_3], c_1$

$H_y = r_3[o_1], w_3[o_1], r_1[o_1], w_1[o_1], w_3[o_2], c_3, w_1[o_2], r_2[o_2], w_2[o_2], w_2[o_1], c_2, w_1[o_3], c_1$

1. Determine what are the ordered conflicting pairs in H_x, and write them down in the form $rw_i[o] \rightarrow rw_j[o]$ (where the rw should be replaced by a r or w in your answer as appropriate). Use your answer to determine if H_x is CSR.

Assignment Project Exam Help

https://eduassistpro.github.io/

2. Determine the conflicting pairs in H_y, and use your answer to determine if H_y is CSR.

Add WeChat edu_assist_pro

$H_w = r_2[o_1], r_2[o_2], w_2[o_2], r_1[o_2], w_2[o_1], r_2[o_3], c_2, c_1$

$H_x = r_2[o_1], r_2[o_2], w_2[o_1], w_2[o_2], w_1[o_1], w_1[o_2], c_1, r_2[o_3], c_2$

$H_y = r_2[o_1], r_2[o_2], w_2[o_2], r_1[o_2], w_2[o_1], c_1, r_2[o_3], c_2$

$H_z = r_2[o_1], w_1[o_1], r_2[o_2], w_2[o_2], r_2[o_3], c_2, r_1[o_2], w_1[o_2], w_1[o_3], c_1$

1. Fill in for each history the list of pairs of operations where one transaction has read from another transaction ($w_i[o] \to r_j[o]$), and where one transaction has overwritten another transaction ($w_i[o] \to w_j[o]$).

H_w	H_x	H_y	H_z

2. Determine if H_w is RC, ACA, or ST.

3. Determine if H_x is RC, ACA, or ST.

4. Determine if H_y is RC, ACA, or ST.

5. Determine if H_z is RC, ACA, or ST.

You are told that three transactions H_1, H_2, H_3 perform the following sequence of operations:

$H_1 : w_1[o_1], r_1[o_2], r_1[o_4]$

$H_2 : r_2[o_3], r_2[o_2], r_2[o_1]$

$H_3 : r_3[o_4], w_3[o_4], r_3[o_3], w_3[o_3]$

1. Write down all the possible conflict pairs from H_1, H_2, H_3 in the form $rw_i \rightarrow rw_j$.

Assignment Project Exam Help

2. Write a concurre   lying all three
   transaction   tion might have to wait
   for another, an

https://eduassistpro.github.io/

Add WeChat edu_assist_pro

3. Draw the WFG for your answer in (2).

4. How would you resolve the deadlock?

The log belong is kept by a DM that is using cache consistent checkpointing and where the scheduler is using strict executions.

LOG	b_7
UNDO	$w_7[b_{67}, \text{cash}=34005.25]$
REDO	$w_7[b_{67}, \text{cash}=37005.25]$
LOG	b_2
UNDO	$w_2[b_{34}, \text{cash}=10900.67]$
REDO	$w_2[b_{34}, \text{cash}=8900.67]$
LOG	b_6
UNDO	$w_6[a_{101}, \text{rate}=5.25]$
REDO	$w_6[a_{101}, \text{rate}=6.00]$
LOG	b_1
UNDO	$w_1[b_{56}, \text{cash}=94340.45]$
REDO	$w_1[b_{56}, \text{cash}=84340.45]$
CP	$\{1, 2, 6\}$
UNDO	$w_6[a_{119}, \text{rate}=5.50]$
REDO	$w_6[a_{119}, \text{rate}=6.00]$
LOG	c_6
UNDO	$w_2[b_{67}, \text{cash}=34005.00]$
REDO	$w_2[b_{67}, \text{cash}=36005.25]$
LOG	c_2
UNDO	$w_1[b_{34}, \text{cash}=8900.67]$
REDO	$w_9[b_{67}, \text{cash}=20000.00]$
LOG	c_9

You have to recover the database after a system failure using the

1. In the backward scan for undos, what will be the set of committed transactions, the set of uncommitted transactions, and list of undo action(s) be when you reach the *cp* record?

2. Which actions(s) must you undo *before* the *cp* record?

3. Which actions(s) must you redo?

4. Can you devise a more efficient algorithm that minimises the number of redo and undo actions performed?