1. SQL and Relational Algebra queries

(f): Solution

$\rho(C, Customer)$
$\rho(P, Purchase)$
$\rho(R, Product)$
$\rho\left(paidMSRP(P.pid \rightarrow pid1, R.pid \rightarrow pid2), \sigma_{P.pid=R.pid \wedge P.price>=R.msrp}(P \times R)\right)$

$\rho\big(TEMP1, \pi_{cid}(paidMSRP)\big)$

$\rho\big(paid(C.cid \rightarrow cid1, P.cid \rightarrow cid2), \sigma_{C.cid=P.cid}(C \times P)\big)$

$\rho\big(TEMP2, \pi_{cid1}(paid)\big)$

$\rho(TEMP, TEMP2 - TEMP1)$
$\pi_{C.cid,C.cname}(TEMP \bowtie C)$

(g): Solution

$\rho(P1, Purchase\ \ P1)$
$\rho(P2, Purchase\ \ P2)$
$\rho(P3, Purchase\ \ P3)$
$\rho(P4, Purchase\ \ P4)$

$$\begin{aligned}\rho(Condition1, P1.cid = P2.cid\ \ AND\ \ P2.cid = P3.cid\ \ AND\ \ P3.cid \\ = P4.cid\ \ AND\ \ P1.pid = P2.pid\ \ AND\ \ P2.pid \\ = P3.pid\ \ AND\ \ P3.pid = P4.pid\ \ AND\ \ P1.date \\ \neq P2.date\ \ AND\ \ P1.date \neq P3.date\ \ AND\ \ P1.date \\ \neq P4.date\ \ AND\ \ P2.date \neq P3.date\ \ AND\ \ P2.date \\ \neq P4.date\ \ AND\ \ P3.date \neq P4.date)\end{aligned}$$

$$\begin{aligned}\rho(Condition2, P1.cid = P2.cid\ \ AND\ \ P2.cid = P3.cid\ \ AND\ \ P1.pid \\ = P2.pid\ \ AND\ \ P2.pid = P3.pid\ \ AND\ \ P1.date \\ \neq P2.date\ \ AND\ \ P1.date \neq P3.date\ \ AND\ \ P2.date \neq P3.date)\end{aligned}$$

$\rho\left(TEMP1, \big(\sigma_{Condition1}(P1 \times P2 \times P3 \times P4)\big)\right)$

$\rho\left(TEMP2, \big(\sigma_{Condition2}(P1 \times P2 \times P3)\big)\right)$

$\pi_{TEMP2.cid,TEMP2.pid(TEMP2-TEMP1)}$;

(h):

$\rho(P1, Purchase\ \ P1)$;
$\rho(P2, Purchase\ \ P2)$;

$\rho\left(TEMP, \pi_{P2.pid,P2.price}\left(\sigma_{P1.pid=P2.pid \wedge P1.price>P2.price}(P1 \times P2)\right)\right)$
$\pi_{P1.pid,P1.price}(P1) - TEMP;$

2. Bag Relational Algebra

We use the notation $t{:}k$ to mean that tuple t appears with multiplicity k. Then the formal definition of bag selection is as follows for an arbitrary bag R and selection condition C:

$$\sigma_C(R) = \{t_k{:}m_k | t_k{:}m_k \in R \ \ and \ \ C \ \ holds \ \ on \ \ t_k\}$$

(a). $\delta(\sigma_C(R)) \equiv \sigma_C(\delta(R))$ **true**

Solution:

LHS →RHS

First we will show $\delta(\sigma_C(R)) \subseteq \sigma_C(\delta(R))$ for arbitrary R.

Fix an arbitrary R and suppose $t \in \delta(\sigma_C(R))$. Then, from the definition of duplicate elimination operator δ and bag selection, we know that $t{:}k$ is in $\sigma_C(R)$ and it is also in R. Plus, C holds on t.

Let us now consider what happens to t in computing $\sigma_C(\delta(R))$. $t{:}k$ is in R. Based on the definition of duplicate elimination operator δ, we know that t is in $\delta(R)$. Because C holds on t, t appears in the result $\sigma_C(\delta(R))$.

RHS →LHS

Then we show $\sigma_C(\delta(R)) \subseteq \delta(\sigma_C(R))$ for arbitrary R.

Fix an arbitrary R and suppose $t \in \sigma_C(\delta(R))$. Then, from the definition of bag selection and duplicate elimination operator δ, we know that C holds on t and $t{:}k$ is in R.

Let us now consider what happens to t in computing $\delta(\sigma_C(R))$. $t{:}k$ is in R and C holds on t, so based on the definition of bag selection, we know that $t{:}k$ is in $\sigma_C(R)$. According to the definition of duplicate elimination operator δ, t is in $\delta(\sigma_C(R))$. So we have $\delta(\sigma_C(R)) \equiv \sigma_C(\delta(R))$.

(b). $\delta(\pi_A(R)) \equiv \pi_A(\delta(R))$ **false**

Solution:

Suppose R:

a	b
1	1
1	2

$\delta(\pi_A(R))$:

a
1

$\pi_A(\delta(R))$:

a
1
1

Which means $\delta(\pi_A(R)) \neq \pi_A(\delta(R))$.

(c). $\delta(R \times S) \equiv \delta(R) \times \delta(S)$ **true**

Solution:

LHS →RHS

First we will show $\delta(R \times S) \subseteq \delta(R) \times \delta(S)$ for arbitrary R and S.

Fix an arbitrary R and an arbitrary S. Suppose $t_1: k_1 \in R$ and $t_2: k_2 \in S$. Let t denotes the tuple that is a concatenation of t_1 and t_2. Then from the definition of bag cross product, we know that $t: k_1 \times k_2 \in R \times S$. Based on the definition of duplicate elimination operator δ, $t \in \delta(R \times S)$. Let us now consider what happens to t in computing $\delta(R) \times \delta(S)$. Based on the definition of duplicate elimination operator δ, we have $t_1 \in \delta(R)$, $t_2 \in \delta(S)$. Because t is the tuple that contains all the attributes in both t_1 and t_2, we have $t \in \delta(R) \times \delta(S)$.

RHS→LHS

Then we show that $\delta(R) \times \delta(S) \in \delta(R \times S)$ for arbitrary R.

Fix an arbitrary R and an arbitrary S. Suppose $t_1: k_1 \in R$ and $t_2: k_2 \in S$. Based on the definition of duplicate elimination operator δ, we have $t_1 \in \delta(R)$, $t_2 \in \delta(S)$. Let t denotes the tuple that is a concatenation of t_1 and t_2, according to the definition of bag cross product, we have $t \in \delta(R) \times \delta(S)$.

Let us now consider what happens to t in computing $\delta(R \times S)$. Based on the definition of bag cross product, $t: k_1 \times k_2 \in R \times S$. According to the definition of duplicate elimination operator δ, $t \in \delta(R \times S)$. So we have $\delta(R \times S) \equiv \delta(R) \times \delta(S)$.

(d). $\delta(R \bowtie_C S) \equiv \delta(R) \bowtie_C \delta(S)$ **true**

Solution:

LHS→RHS

First we will show $\delta(R \bowtie_C S) \subseteq \delta(R) \bowtie_C \delta(S)$ for arbitrary R and S. We know that $R \bowtie_C S = \sigma_C(R \times S)$. Fix an arbitrary R and an arbitrary S. Suppose $t_1\!: k_1 \in R$ and $t_2\!: k_2 \in S$. Let t denotes the tuple that is a concatenation of t_1 and t_2. Then from the definition of bag cross product, we know that $t\!: k_1 \times k_2 \in R \times S$. Based on the definition of bag selection, we have $t\!: k_1 \times k_2 \in \sigma_C(R \times S)$ and C holds on t. According to the definition of duplicate elimination operator δ, $t \in \delta(R \bowtie_C S)$.

Let us now consider what happens to t in computing $\delta(R) \bowtie_C \delta(S)$. We know that $\delta(R) \bowtie_C \delta(S) = \sigma_C\big(\delta(R) \times \delta(S)\big)$. Based on the definition of duplicate elimination operator δ and the definition of bag cross product, t is in $\delta(R) \times \delta(S)$. Because C holds on t, $t \in \sigma_C\big(\delta(R) \times \delta(S)\big) = \delta(R) \bowtie_C \delta(S)$.

RHS→LHS

Then we show $\delta(R) \bowtie_C \delta(S) \subseteq \delta(R \bowtie_C S)$ for arbitrary R and S. Fix an arbitrary R and an arbitrary S. Suppose $t_1\!: k_1 \in R$ and $t_2\!: k_2 \in S$. Based on the definition of duplicate elimination operator δ, $t_1 \in \delta(R)$, $t_2 \in \delta(S)$. Let t denotes the tuple that is a concatenation of t_1 and t_2, according to the definition of bag cross product, we have $t \in \delta(R) \times \delta(S)$ and $t \in R \times S$. Suppose $t \in \sigma_C\big(\delta(R) \times \delta(S)\big)$, that is, C holds on t. Let us now consider what happens to t in computing $\delta(R \bowtie_C S)$. Because $t \in R \times S$ and C holds on t, $t \in \sigma_C(R \times S)$. Based on the definition of duplicate elimination operator δ, $t \in \delta(R \bowtie_C S)$. So we have $\delta(R \bowtie_C S) \equiv \delta(R) \bowtie_C \delta(S)$.

(e). $\delta(R \cup_B S) \equiv \delta(R) \cup_B \delta(S)$ **false**

Solution:

Suppose R:

a	b
1	2
1	2
3	4

S:

a	b
1	2
3	4
3	4

$\delta(R \cup_B S)$:

a	b
1	2
3	4

$\delta(R) \cup_B \delta(S)$:

a	b
1	2
1	2
3	4
3	4

Which means: $\delta(R \cup_B S) \neq \delta(R) \cup_B \delta(S)$

(f): $\delta(R \cap_B S) \equiv \delta(R) \cap_B \delta(S)$ **true**

Solution:

LHS→RHS

First we show $\delta(R \cap_B S) \subseteq \delta(R) \cap_B \delta(S)$ for arbitrary R and S. Suppose $t_1{:}k_1 \in R$ and $t_2{:}k_2 \in S$. Based on the definition of bag intersection, we have $t_1{:}k_1' \in R \cap_B S$ $(0 \leq k_1' \leq k_1)$, $t_2{:}k_2' \in R \cap_B S$ $(0 \leq k_2' \leq k_2)$. According to the definition of duplicate elimination operator δ, $t_1{:}k_1' \in \delta(R \cap_B S)$ $(0 \leq k_1' \leq 1)$, $t_2{:}k_2' \in \delta(R \cap_B S)$ $(0 \leq k_2' \leq 1)$.

Let us now consider what happens to t_1 and t_2 in computing $\delta(R) \cap_B \delta(S)$. Since that $t_1{:}k_1' \in R \cap_B S$ $(0 \leq k_1' \leq k_1)$, based on the definition of bag intersection, we have $t_1{:}k_1' \in R(0 \leq k_1' \leq k_1)$. For the same reason, $t_2{:}k_2' \in S$ $(0 \leq k_2' \leq k_2)$. According to the definition of duplicate elimination operator δ, we have $t_1{:}k_1' \in \delta(R)$ $(0 \leq k_1' \leq 1)$ and $t_2{:}k_2' \in \delta(S)$ $(0 \leq k_2' \leq 1)$. Finally, based on the definition of bag intersection, we have $t_1{:}\,k_1' \in \delta(R) \cap_B \delta(S)$ $(0 \le k_1' \le 1)$, $t_2{:}\,k_2' \in \delta(R) \cap_B \delta(S)$ $(0 \le k_2' \le 1)$.

RHS→LHS

Then we show $\delta(R) \cap_B \delta(S) \subseteq \delta(R \cap_B S)$ for arbitrary R and S. Suppose $t_1{:}\,k_1 \in R$ and $t_2{:}\,k_2 \in S$. Based on the definition of duplicate elimination operator δ, we have $t_1 \in \delta(R)$ and $t_2 \in \delta(S)$. Then according to the definition of bag intersection, we have $t_1{:}\,k_1' \in \delta(R) \cap_B \delta(S)$ $(0 \le k_1' \le 1)$ and $t_2{:}\,k_2' \in \delta(R) \cap_B \delta(S)$ $(0 \le k_2' \le 1)$. Let us now consider what happens to t_1 and t_2 in computing $\delta(R \cap_B S)$. Since that $t_1{:}\,k_1 \in R$ and $t_2{:}\,k_2 \in S$, based on the definition of bag intersection, we have $t_1{:}\,k_1' \in R \cap_B S$ $(0 \le k_1' \le k_1)$, $t_2{:}\,k_2' \in R \cap_B S$ $(0 \le k_2' \le k_2)$. Finally, according to the definition of duplicate elimination operator δ, we have $t_1{:}\,k_1' \in \delta(R \cap_B S)$ $(0 \le k_1' \le 1)$ and $t_2{:}\,k_2' \in \delta(R \cap_B S)$ $(0 \le k_2' \le 1)$. So $\delta(R \cap_B S) \equiv \delta(R) \cap_B \delta(S)$.

(g). $\delta(R -_B S) \equiv \delta(R) -_B \delta(S)$ false

Solution:

Suppose R:

a	b
1	2
1	2
3	4

S:

a	b
1	2
3	4
3	4

$\delta(R -_B S)$:

a	b
1	2

But $\delta(R) -_B \delta(S)$ has no element. Which means that the equivalence is false.