This exam is open book and notes. You may use a laptop, but please do not access the Internet, and please keep your sound on mute so you do not disturb others.

This exam consists of 35 multiple-choice questions, so you have approximately 5 minutes per question. Questions count 3 points each, with 1 point deducted for wrong answers (nothing deducted if you choose not to answer a question). The maximum score is 105.

Print your name: ________________________________________

The Honor Code is an undertaking of the students, individually and collectively:

1. that they will not give or receive aid in examinations; that they will not give or receive unpermitted aid in class work, in the preparation of reports, or in any other work that is to be used by the instructor as the basis of grading;

2. that they will do their share and take an active part in seeing to it that others as well as themselves uphold the spirit and letter of the Honor Code.

The faculty on its part manifests its confidence in the honor of its students by refraining from proctoring examinations and from taking unusual and unreasonable precautions to prevent the forms of dishonesty mentioned above. The faculty will also avoid, as far as practicable, academic procedures that create temptations to violate the Honor Code.

While the faculty alone has the right and obligation to set academic requirements, the students and faculty will work together to establish optimal conditions for honorable academic work.

I acknowledge and accept the Honor Code.

Signed: ____________________________________________
In each of the first 16 questions, you are asked to compare two queries $Q_1$ and $Q_2$. You must tell whether the queries are:

1. The same [choice (a)], meaning that for every database the answers to the two queries are the same. That is, the same tuples are produced by each query, and a tuple is produced the same number of times by each query. The order in which tuples are produced is not to be considered.

2. Completely different [choice (d)], meaning that there are databases where $Q_1$ produces more of some particular tuple, and other databases where $Q_2$ produces more of some particular tuple. Note that the query producing the smaller number of copies of a tuple may produce zero copies of that tuple.

3. One is contained in the other but they are not the same [choice (b) or (c)]. For instance, $Q_1$ is contained in $Q_2$ if on every database, $Q_2$ produces at least as many copies of each tuple as $Q_1$ does. Note that it is possible $Q_2$ produces one or more copies of a tuple, while $Q_1$ produces none of that tuple.

General advice:

- Do not assume a query has a trivial syntactic error and therefore produces nothing.
- Relations mentioned in the queries may have attributes not mentioned, but their existence should not affect the answer.
- SQL relations may have NULL’s, although in other languages, you should assume no NULL’s unless stated otherwise.
- SQL queries should be assumed to be in standard SQL unless stated otherwise.
- In SQL and OQL, it is possible that there may be duplicate tuples, but in relational algebra or Datalog assume the relations are sets unless stated otherwise.

1. The following relational-algebra queries refer to a relation $R(a, b, c)$.

   $Q_1$: $\gamma_{a,b}(R)$
   $Q_2$: $\pi_{a,b}(R)$

   (a) $Q_1$ and $Q_2$ produce the same answer.
   (b) The answer to $Q_1$ is always contained in the answer to $Q_2$.
   (c) The answer to $Q_2$ is always contained in the answer to $Q_1$.
   (d) $Q_1$ and $Q_2$ produce different answers.
2. The following SQL queries refer to a relation $R(a,b,c)$.

$$Q_1: \text{SELECT } a, b \text{ FROM } R;$$
$$Q_2: \text{SELECT } a, b \text{ FROM } R \text{ GROUP BY } a, b;$$

(a) $Q_1$ and $Q_2$ produce the same answer.
(b) The answer to $Q_1$ is always contained in the answer to $Q_2$.
(c) The answer to $Q_2$ is always contained in the answer to $Q_1$.
(d) $Q_1$ and $Q_2$ produce different answers.

3. In the following Datalog programs, $A$, $B$, and $C$ are EDB predicates; $P$, $Q$, and $Answer$ are IDB predicates. The result of the query is the value of $Answer$ in the stratified model. Note that in $Q_2$ the second rule helps define $P$, not $Q$.

$$Q_1: \begin{align*}
P(x) &\leftarrow A(x) \\
Q(x) &\leftarrow B(x) \text{ AND NOT } P(x) \\
Answer(x) &\leftarrow C(x) \text{ AND NOT } Q(x)
\end{align*}$$
$$Q_2: \begin{align*}
P(x) &\leftarrow A(x) \\
P(x) &\leftarrow B(x) \\
Answer(x) &\leftarrow C(x) \text{ AND NOT } P(x)
\end{align*}$$

(a) $Q_1$ and $Q_2$ produce the same answer.
(b) The answer to $Q_1$ is always contained in the answer to $Q_2$.
(c) The answer to $Q_2$ is always contained in the answer to $Q_1$.
(d) $Q_1$ and $Q_2$ produce different answers.

4. The following relational-algebra queries refer to relations $R(a,b)$ and $S(b,c)$.

$$Q_1: \pi_{a,b}(R \bowtie S)$$
$$Q_2: \pi_{b}(S)$$

(a) $Q_1$ and $Q_2$ produce the same answer.
(b) The answer to $Q_1$ is always contained in the answer to $Q_2$.
(c) The answer to $Q_2$ is always contained in the answer to $Q_1$.
(d) $Q_1$ and $Q_2$ produce different answers.

5. The following SQL queries refer to relations $R(a,b)$ and $S(b,c)$.

$$Q_1: \text{SELECT } b \text{ FROM } R \text{ WHERE } b \text{ IN (SELECT } b \text{ FROM } S);$$
$$Q_2: \text{SELECT } b \text{ FROM } S \text{ WHERE } b \text{ IN (SELECT } b \text{ FROM } R);$$

(a) $Q_1$ and $Q_2$ produce the same answer.
(b) The answer to $Q_1$ is always contained in the answer to $Q_2$.
(c) The answer to $Q_2$ is always contained in the answer to $Q_1$.
(d) $Q_1$ and $Q_2$ produce different answers.
6. The following SQL queries refer to a relation $R(a, b)$.

$$Q_1: \text{SELECT } a \text{ FROM } R \text{ WHERE } b \text{ LIKE } '%%%';$$
$$Q_2: \text{SELECT } a \text{ FROM } R \text{ WHERE } b \text{ LIKE } '%%' \text{ AND } b \text{ LIKE } '%%';$$

(a) $Q_1$ and $Q_2$ produce the same answer.
(b) The answer to $Q_1$ is always contained in the answer to $Q_2$.
(c) The answer to $Q_2$ is always contained in the answer to $Q_1$.
(d) $Q_1$ and $Q_2$ produce different answers.

7. In the following, the “queries” $Q_1$ and $Q_2$ are actually DTD’s, and their “answers” are the sets of documents that match them.

$$Q_1: \text{<!DOCTYPE A [}
\text{<!ELEMENT A(B, C)>}
\text{<!ELEMENT B(#PCDATA)>}
\text{<!ELEMENT C(#PCDATA)>}
\text{]}>$$
$$Q_2: \text{<!DOCTYPE A [}
\text{<!ELEMENT A(B?, C)>}
\text{<!ELEMENT B(#PCDATA)>}
\text{<!ELEMENT C(#PCDATA)>}
\text{]}>$$

(a) $Q_1$ and $Q_2$ produce the same answer.
(b) The answer to $Q_1$ is always contained in the answer to $Q_2$.
(c) The answer to $Q_2$ is always contained in the answer to $Q_1$.
(d) $Q_1$ and $Q_2$ produce different answers.

8. The following deletions followed by queries each start with the same relation $R(a, b)$. The results are what the queries return after deletion.

$$Q_1: \text{DELETE FROM } R \text{ rr WHERE EXISTS (}
\text{SELECT * FROM } R \text{ WHERE } a \text{ <> rr.a);}$$
$$\text{SELECT * FROM } R;$$
$$Q_2: \text{DELETE FROM } R \text{ WHERE a <> ALL (}
\text{SELECT } a \text{ FROM } R;)$$
$$\text{SELECT } * \text{ FROM } R;$$

(a) $Q_1$ and $Q_2$ produce the same answer.
(b) The answer to $Q_1$ is always contained in the answer to $Q_2$.
(c) The answer to $Q_2$ is always contained in the answer to $Q_1$.
(d) $Q_1$ and $Q_2$ produce different answers.
class A (extent As key name) {
    attribute string name;
    relationship Set<B> myBs inverse B::myA;
}

class B (extent Bs key name) {
    attribute string name;
    relationship A myA inverse A::myBs;
    relationship Set<C> myCs inverse C::myB;
}

class C (extent Cs key name) {
    attribute string name;
    relationship B myB inverse B::myCs;
}

Figure 1: ODL schema

9. The following OQL queries apply to the ODL schema of Fig. 1.

\[ Q_1: \text{SELECT c.name} \text{ FROM As a, a.myBs b, b.myCs c} \text{ WHERE a.name = "Joe"} \]
\[ Q_2: \text{SELECT c.name} \text{ FROM Cs c} \text{ WHERE c.myB.myA.name = "Joe"} \]

(a) \(Q_1\) and \(Q_2\) produce the same answer.
(b) The answer to \(Q_1\) is always contained in the answer to \(Q_2\).
(c) The answer to \(Q_2\) is always contained in the answer to \(Q_1\).
(d) \(Q_1\) and \(Q_2\) produce different answers.

10. The following OQL queries apply to the ODL schema of Fig. 1.

\[ Q_1: \text{SELECT DISTINCT c.myB.myA} \text{ FROM Cs c} \]
\[ Q_2: \text{SELECT DISTINCT b.myA} \text{ FROM Bs b} \]

(a) \(Q_1\) and \(Q_2\) produce the same answer.
(b) The answer to \(Q_1\) is always contained in the answer to \(Q_2\).
(c) The answer to \(Q_2\) is always contained in the answer to \(Q_1\).
(d) \(Q_1\) and \(Q_2\) produce different answers.
11. The following SQL queries refer to a relation $R(a, b)$.

$$Q_1: \text{SELECT COUNT}(a) \text{ FROM } R;$$
$$Q_2: \text{SELECT COUNT}(*) \text{ FROM } R \text{ WHERE } a \text{ IS NOT NULL;}$$

(a) $Q_1$ and $Q_2$ produce the same answer.
(b) The answer to $Q_1$ is always contained in the answer to $Q_2$.
(c) The answer to $Q_2$ is always contained in the answer to $Q_1$.
(d) $Q_1$ and $Q_2$ produce different answers.

12. In the following Datalog programs, $Parent$ is an EDB predicate; $Answer$ and $Q$ are IDB predicates. The result of each query is the value of $Answer$. Hint: First, remember that cousins are individuals that have a common $nth$-generation ancestor for some $n \geq 1$. Then, notice that $Q$ represents “cousins once removed.” That is, $Q(x, y)$ is true if a parent of $x$ is a cousin of $y$ (or in the basis case, $y$ him/herself).

$$Q_1: \text{Answer}(x, y) \leftarrow Parent(x, z) \text{ AND } Parent(y, z)$$
$$\text{Answer}(x, y) \leftarrow Parent(x, z) \text{ AND } Answer(z, y)$$
$$\text{Answer}(x, y) \leftarrow Answer(x, z) \text{ AND } Parent(y, z)$$

$$Q_2: Q(x, y) \leftarrow Parent(x, y)$$
$$\text{Q}(x, y) \leftarrow Parent(x, z) \text{ AND } Answer(z, y)$$
$$\text{Answer}(x, y) \leftarrow Q(x, z) \text{ AND } Parent(y, z)$$

(a) $Q_1$ and $Q_2$ produce the same answer.
(b) The answer to $Q_1$ is always contained in the answer to $Q_2$.
(c) The answer to $Q_2$ is always contained in the answer to $Q_1$.
(d) $Q_1$ and $Q_2$ produce different answers.

13. The following queries apply to a relation $R(a, b, c)$.

$$Q_1: \text{SELECT r1.a}$$
$$\text{FROM R r1, R r2}$$
$$\text{WHERE r1.b = r2.c;}$$

$$Q_2: \text{SELECT a FROM R WHERE b IN}$$
$$\text{(SELECT c FROM R);}$$

(a) $Q_1$ and $Q_2$ produce the same answer.
(b) The answer to $Q_1$ is always contained in the answer to $Q_2$.
(c) The answer to $Q_2$ is always contained in the answer to $Q_1$.
(d) $Q_1$ and $Q_2$ produce different answers.
14. In the following, $R(a,b,c)$ is a relation with no NULL's, attribute $c$ is of type INTEGER, and $V(a,b,d)$ is the view defined by:

```sql
SELECT a, b, SUM(c) AS d
FROM R
GROUP BY a, b;
```

- $Q_1$: SELECT $b$, $d$ FROM $V$ WHERE $a=1$;
- $Q_2$: SELECT $b$, SUM($d$) FROM $V$ GROUP BY $a$, $b$ HAVING $a=1$;

(a) $Q_1$ and $Q_2$ produce the same answer.
(b) The answer to $Q_1$ is always contained in the answer to $Q_2$.
(c) The answer to $Q_2$ is always contained in the answer to $Q_1$.
(d) $Q_1$ and $Q_2$ produce different answers.

15. In the following relational algebra expressions, assume union and intersection are bag operators, and the relations $R(x)$, $S(x)$, and $T(x)$ are bags.

```sql
Q_1: R \cap (S \cup T)
Q_2: (R \cap S) \cup (R \cap T)
```

(a) $Q_1$ and $Q_2$ produce the same answer.
(b) The answer to $Q_1$ is always contained in the answer to $Q_2$.
(c) The answer to $Q_2$ is always contained in the answer to $Q_1$.
(d) $Q_1$ and $Q_2$ produce different answers.

16. In the following relational algebra expressions, again assume union and intersection are bag operators, and the relations $R(x)$, $S(x)$, and $T(x)$ are bags.

```sql
Q_1: R \cup (S \cap T)
Q_2: (R \cup S) \cap (R \cup T)
```

(a) $Q_1$ and $Q_2$ produce the same answer.
(b) The answer to $Q_1$ is always contained in the answer to $Q_2$.
(c) The answer to $Q_2$ is always contained in the answer to $Q_1$.
(d) $Q_1$ and $Q_2$ produce different answers.
17. Suppose we have relations $R(a, b)$ and $S(b, c)$, and the assertion:

```
CREATE ASSERTION Mystery CHECK (
    NOT EXISTS (SELECT * FROM R NATURAL JOIN S WHERE R.a = S.c)
);
```

Which of the following combinations of tuples are prohibited from existing at the same time?

(a) $(1, 1)$ in $R$ and $(2, 2)$ in $S$.
(b) $(1, 2)$ in $R$ and $(2, 1)$ in $S$.
(c) $(1, 2)$ in $R$ and $(1, 2)$ in $S$.
(d) $(1, 1)$ in $R$ and $(1, 2)$ in $S$.

![Figure 2: E/R diagram](image)

18. If we convert the E/R diagram of Fig. 2 to a relational database schema using the “E/R approach,” the number of relations we get is:

(a) 3   (b) 4   (c) 8   (d) 9

19. If we convert the E/R diagram of Fig. 2 to a relational database schema using the “object-oriented approach,” the number of relations we get is:

(a) 3   (b) 4   (c) 8   (d) 9
Figure 3: UML diagram

20. Consider the UML diagram of Fig. 3. Note that PK stands for “primary key” and indicates that a1 is the key for A and c1 is the key for C. If we convert this UML diagram to relations, then the most reasonable schema for association class D is:

(a) \( D(a1, a2, d1, d2, c1, c2) \)
(b) \( D(a1, c1) \)
(c) \( D(d1, d2) \)
(d) \( D(a1, d1, d2, c1) \)

21. If we implement the UML diagram of Fig. 3 in an object-relational database system, the most reasonable approach to handling C and E is:

(a) One relation with three attributes: c1, c2, and a third attribute whose type is objects that are \( (e1, e2) \) pairs.
(b) One relation with three attributes: c1, e2, and a third attribute whose type is objects that are \( (c1, e2) \) pairs.
(c) Three relations: \( C(c1, c2) \), \( E(e1, e2) \), and a third relation with attributes c1, e1, and e2, representing the association between C and E.
(d) One relation with attributes c1, e1, and e2.

22. In the following Datalog rule:

\[
\text{Answer}(x, y) \leftarrow P(x, y) \text{ AND } Q(y, x)
\]

The schemas of the EDB relations are \( P(a, b) \) and \( Q(c, d) \). Which of the following expressions of relational algebra does **not** produce the same relation as the value of Answer?

(a) \( P \cap \rho_{Q(a, b)}(\pi_{d, c}(Q)) \)
(b) \( \pi_{d, c}(\sigma_{a=c} \text{ AND } b=d(P \times Q)) \)
(c) \( P \bowtie \pi_{a,b}(\rho_{Q(a, c)}(Q)) \)
(d) \( \pi_{a,b}(P \downarrow_{a=d} \text{ AND } b=c \quad Q) \)
23. Consider the following trigger, which has applied to a relation \( R(a,b) \) from the time \( R \) was first created. You may assume \( R \) is not allowed to have NULL's anywhere, and no attempts to insert tuples with NULL's are ever made.

```
CREATE TRIGGER Mystery
BEFORE INSERT ON R
REFERENCING NEW ROW AS NewRow
FOR EACH ROW
WHEN (NewRow.a = ANY (SELECT a FROM R))
DELETE FROM R WHERE a = NewRow.a;
```

Which statement best describes the condition enforced by the trigger Mystery?

(a) \( R \) is empty.
(b) \( R \) can have only one tuple.
(c) Attribute \( a \) is a key for \( R \).
(d) Attribute \( b \) is a foreign key, referencing \( R.a \).

24. Relation \( R(\{A,B,C,D,E\}) \) satisfies dependencies \( AB \rightarrow CD \) and \( E \rightarrow C \). Which dependency does \( R \) not necessarily satisfy?

(a) \( AB \rightarrow E \)    (b) \( AB \rightarrow C \)    (c) \( AB \rightarrow D \)    (d) \( E \rightarrow C \)

25. In the relation \( R(\{A,B,C,D,E\}) \) with functional dependencies \( AB \rightarrow C \), \( B \rightarrow D \), \( A \rightarrow E \), and \( DE \rightarrow C \), the number of keys is:

(a) 1  (b) 2  (c) 7  (d) 8

26. The relation \( R(\{A,B,C,D\}) \) with functional dependencies \( A \rightarrow B \), \( B \rightarrow A \), \( C \rightarrow D \), and \( D \rightarrow C \) is:

(a) Not in 3NF.
(b) In 3NF but not in BCNF.
(c) In BCNF but not in 4NF.
(d) In 4NF.
Transaction $T_1$:
\[
\begin{align*}
&\text{INSERT INTO S SELECT SUM(A) FROM R;} \\
&\text{INSERT INTO C SELECT COUNT(A) FROM R;} \\
&\text{COMMIT}
\end{align*}
\]

Transaction $T_2$:
\[
\begin{align*}
&\text{INSERT INTO R VALUES(1);} \\
&\text{COMMIT}
\end{align*}
\]

Figure 4: Two transactions on unary relations $R(A)$, $S(X)$, and $C(Y)$

27. This question refers to the transactions in Fig. 4. Suppose $T_1$ and $T_2$ are issued at approximately the same time, so that it is possible for their operations to become interleaved. Transaction $T_1$ runs at isolation level READ COMMITTED, while $T_2$ runs at isolation level SERIALIZABLE. Initially, relations $R(A)$, $S(X)$, and $C(Y)$ are empty. After both transactions complete, there is exactly one value in $S$ (call it $s$) and one value in $C$ (call it $c$). Which of the following is an invalid combination of values (i.e., one that could never arise)?

(a) $s = 0$, $c = 0$.
(b) $s = 0$, $c = 1$.
(c) $s = 1$, $c = 1$.
(d) None of the above.

28. This question also refers to the transactions in Fig. 4. Suppose $T_1$ and $T_2$ are issued at approximately the same time, so that it is possible for their operations to become interleaved. Transaction $T_1$ runs at isolation level SERIALIZABLE, while $T_2$ runs at isolation level READ COMMITTED. Initially, relations $R$, $S$, and $C$ are empty. After both transactions complete, there is exactly one value in $S$ (call it $s$) and one value in $C$ (call it $c$). Which of the following is an invalid combination of values (i.e., one that could never arise)?

(a) $s = 0$, $c = 0$.
(b) $s = 0$, $c = 1$.
(c) $s = 1$, $c = 1$.
(d) None of the above.
29. Consider a database with relation \( R(x, y) \) and users Alice, Bob, Carol, and Dave. Alice owns relation \( R \). The following sequence of operations takes place:

Alice: GRANT UPDATE\((x)\) ON \( R \) TO Bob WITH GRANT OPTION  
Alice: GRANT UPDATE\((y)\) ON \( R \) TO Carol WITH GRANT OPTION  
Alice: GRANT UPDATE ON \( R \) TO Dave WITH GRANT OPTION  
Bob: GRANT UPDATE\((x)\) ON \( R \) TO Carol WITH GRANT OPTION  
Carol: GRANT UPDATE\((y)\) ON \( R \) TO Dave WITH GRANT OPTION  
Dave: GRANT UPDATE\((y)\) ON \( R \) TO Bob  
Alice: REVOKE UPDATE\((y)\) ON \( R \) FROM Carol CASCADE  
Alice: REVOKE UPDATE\((x)\) ON \( R \) FROM Bob CASCADE

After these statements are executed, which of the following statements is true?

(a) Bob does not have the UPDATE\((y)\) ON \( R \) privilege.  
(b) Carol does not have the UPDATE\((x)\) ON \( R \) privilege.  
(c) Dave does not have the UPDATE\((y)\) ON \( R \) privilege.  
(d) None of the above

30. Consider the following Datalog program, with EDB predicates \( Red(x, y) \) and \( Blue(x, y) \):

\[
\begin{align*}
\text{P}(x,y) & \leftarrow \text{Red}(x,y) \\
\text{Q}(x,y) & \leftarrow \text{Blue}(x,y) \\
\text{P}(x,y) & \leftarrow \text{Red}(x,z) \land \text{Q}(z,y) \\
\text{Q}(x,y) & \leftarrow \text{Blue}(x,z) \land \text{P}(z,y)
\end{align*}
\]

Suppose \( Red = \{(1,2),(4,1)\} \) and \( Blue = \{(1,2),(2,3),(3,4)\} \). Then which of the following is not in \( P \)?

(a) (1,3)  
(b) (1,2)  
(c) (4,3)  
(d) (4,2)

31. Consider the following Datalog program \( D \) with EDB predicates \( R, S, \) and \( T \):

\[
\begin{align*}
\text{Q}(x) & \leftarrow \text{R}(x) \land \text{NOT} \text{S}(x) \\
\text{P}(x) & \leftarrow \text{T}(x) \land \text{NOT} \text{Q}(x)
\end{align*}
\]

Suppose \( R = \{(1),(2)\}, S = \{(1)\}, \) and \( T = \{(2),(3)\} \). Which of these is not a model for program \( D \)?

(a) \( Q = \{(2)\}, P = \{(3)\} \)
(b) \( Q = \{(2),(3)\}, P = \{} \)
(c) \( Q = \{(1),(2)\}, P = \{} \)
(d) \( Q = \{(1),(2),(3)\}, P = \{(2)\} \)
32. This question refers to the XML fragment below, which has no DTD associated with it:

```xml
<vals>
  <val1>four</val1>
  <val2>34</val2>
  <val3>02</val3>
</vals>
```

In XQuery, which of the following comparisons yields a truth value of TRUE?

(a) //val1 = 4
(b) //val2 = "034"
(c) //val3 > 1
(d) None of the above

```xml
<bib>
  <book isbn="1234">
    <title>TCP/IP Illustrated</title>
    <author>Stevens</author>
    <publisher>Addison-Wesley</publisher>
  </book>
</bib>
```

```xml
<bib>
  <book isbn="5678">
    <title>Advanced Unix Programming</title>
    <author>Stevens</author>
  </book>
</bib>
```

```xml
<bib>
  <book>
    <title>Data on the Web</title>
    <author>Abiteboul</author>
    <author>Buneman</author>
    <author>Suciu</author>
    <year>2000</year>
  </book>
</bib>
```

Figure 5: XML data contained in file books.xml

33. This question refers to the XML fragment in Figure 5. Which of the following XPath (or XQuery) expressions does not match any node?

(a) //book[@isbn]/year
(b) //book/author[2]
(c) //book[author="Stevens"]/publisher
(d) None of the above
CREATE TYPE EmpType AS (  
    name CHAR(50),  
    addr CHAR(50)  
);  
CREATE TYPE HierType AS (  
    emp REF(EmpType),  
    mgr REF(EmpType)  
);  
CREATE TABLE Employees OF EmpType (  
    REF IS empID SYSTEM GENERATED  
);  
CREATE TABLE Hierarchy OF HierType;

Figure 6: Types and tables for employee objects and a managerial hierarchy on those objects

34. In Fig. 6 is a series of declarations. These define two object types, EmpType and HierType; the intent is that HierType objects refer to pairs of employees such that the second is the manager of the first. We also see two tables, one to hold the employee objects, and the other to hold the hierarchy objects. We wish to query this database to find the manager of the employee(s) with name 'Joe'. Which substitution for xxx and yyy will make the query below conform to the SQL-99 standard?

    SELECT xxx  
    FROM Hierarchy hh  
    WHERE yyy = 'Joe';

(a) xxx = hh.mgr->name; yyy = hh.emp->name  
(b) xxx = hh->mgr->name; yyy = hh->emp->name  
(c) xxx = hh->mgr.name; yyy = hh.emp.name  
(d) xxx = hh.mgr.name; yyy = hh.emp.name

35. If we were to query the relation Employees from Fig. 6 with the query:

    SELECT * FROM Employees;

which of the following might be one of the tuples in the result?

(a) ('Joe', 'Maple St.')  
(b) Employees('Joe', 'Maple St.')  
(c) EmpType('Joe', 'Maple St.')  
(d) A long string used to represent the ID of an employee object internally.