☑ Instructions

Uppsala University

Department of Information Technology

Database Design I (1DL301)

2021-08-16

Instructions: Read through the complete exam and note below any unclear directives before you start solving the questions. Answer **all** questions.
The paper has two types of questions:

- If a **question is marked with ♥** you must **select ALL correct choices**. If you do not select all correct choices or you include any incorrect choice, your answer will be marked as incorrect.
- For all **other questions** you must **select only one choice** even if there are several correct choices. Your answer will be marked as correct if you select any of the correct choices. If you select an incorrect choice or select more than one choice, your answer will be marked as incorrect.

Please also answer questions: ♣ Q1, Q2 and Q3 which can be useful to us.

Grading. For each correct answer, you gain 1 point. A wrong answer does not generate negative points. To achieve a grade of 3, you must gain at least 14 points in the whole exam. To achieve a grade of 4, you must gain at least 17 points in the whole exam. To achieve a grade of 5, you must collect at least 21 points in the whole exam.

You can email us at **georgios.kalamatianos@it.uu.se** and **georgios.fakas@it.uu.se** for any emergency questions during the examination.

If you find any unclear directives, please note the question number below and explain what you think is unclear.

1 ♣ Question G1: When

General questions (useful for us)

When have you attended the course?

Select one alternative (no points awarded for this question):

- 2020
- 2019
- 2018
- Before 2018

Maximum marks: 0

2 ♣ Question G2: How many

General questions (useful for us)

How many lectures have you attended?

Select one alternative (no points awarded for this question):

- None or very few
- Around 25%
- Around 50%
- Around 75%
- Almost all

Maximum marks: 0

3 ♣ Question G3: Study program

General questions (useful for us)

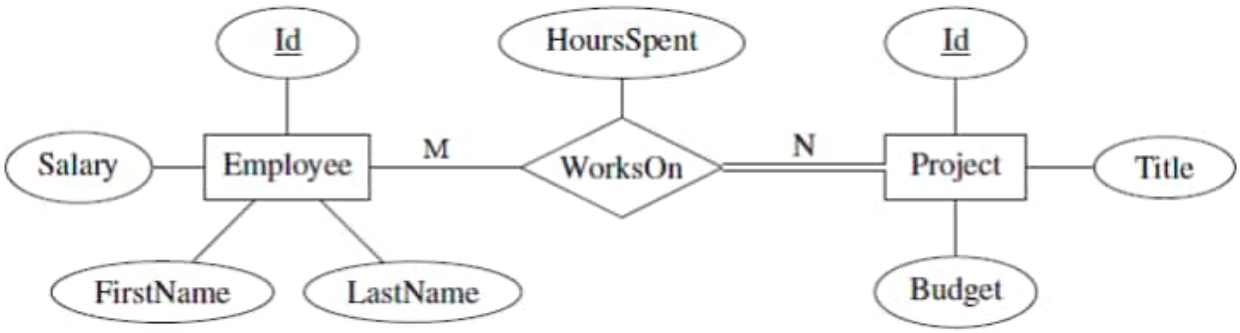
What is your study program?
Select one alternative (no points awarded for this question):

- F
- STS
- CS
- X
- IT
- None of the previous answers

Maximum marks: 0

4 ♥ Interpreting ER model

For the depicted ER model, select **all** statements that must hold!



♥ Select one or more alternatives:

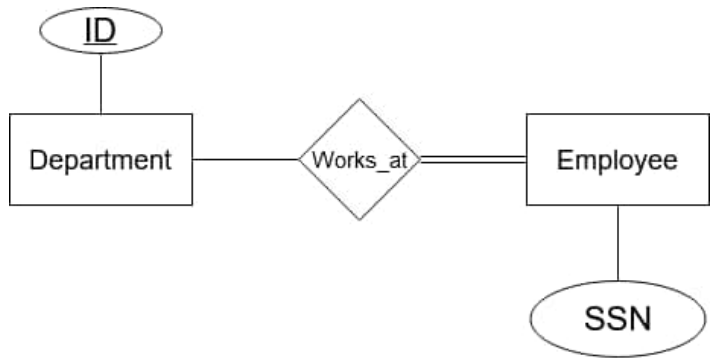
- [ ] There can be projects on which no employees work.
- [ ] It is possible that a project has only a single employee working on it.
- [ ] An employee must work on several projects.
- [ ] The ID for an employee must match the ID for some project.
- [ ] There can be employees who work on no projects.

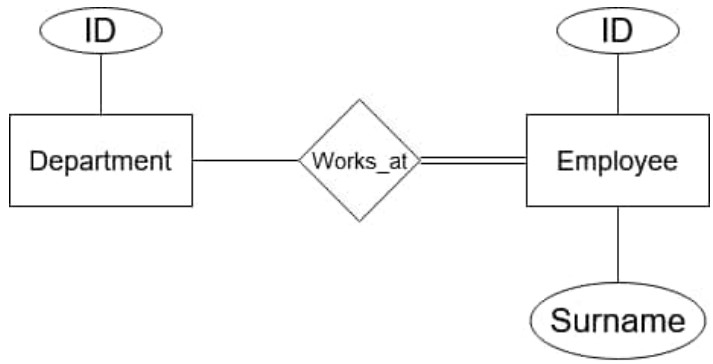
Maximum marks: 1

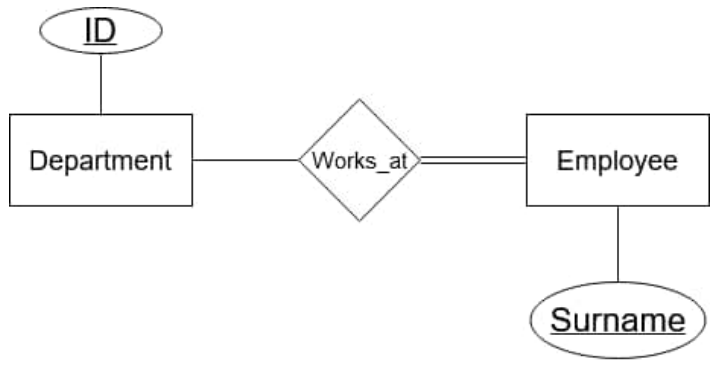
5 Department Employee ER

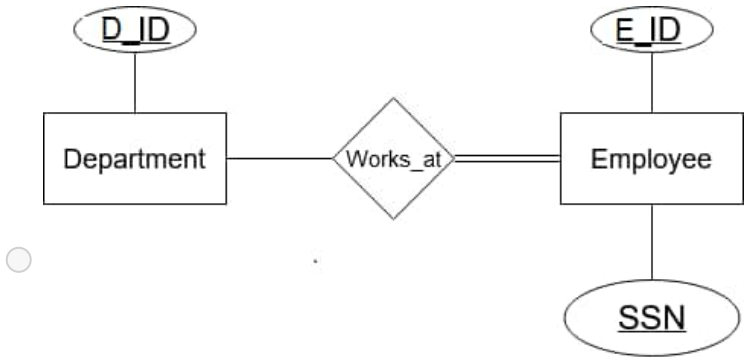
Consider a reasonable interpretation of the entities and attributes of the following ER-diagrams, where underlined attributes indicate candidate keys. Which of the following ER-diagrams is the most appropriate?

Select one alternative:





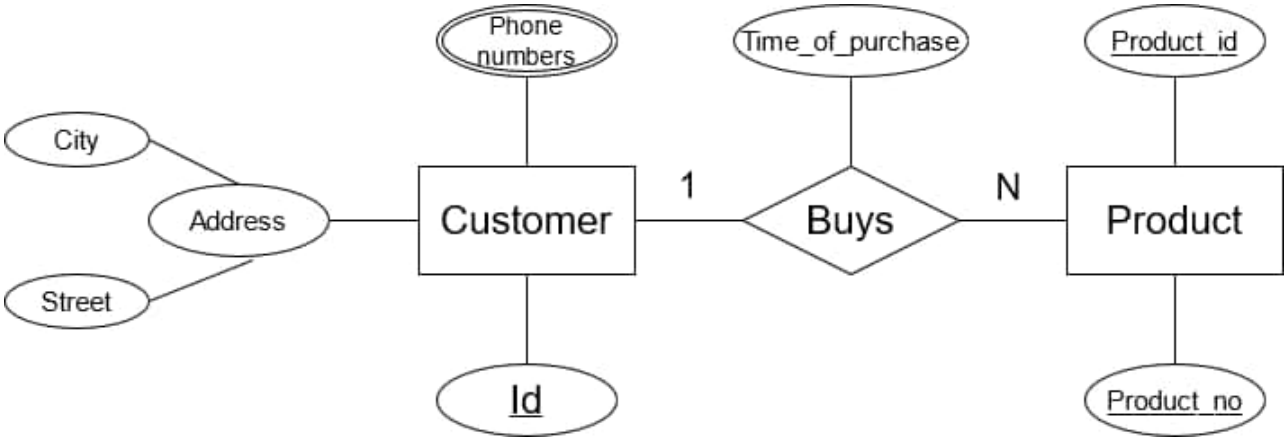




Maximum marks: 1

6 Mapping ER to relational model

Which of the following options is a correct mapping of the ER diagram to the relational model?



Select one alternative:

- CUSTOMER(Id, City, Street, Product_id, Product_no), with {Product_id, Product_no} $^{FK}\rightarrow$ PRODUCT({Product_id, Product_no}) PHONES(Phone_number, Id), with Id $^{FK}\rightarrow$ CUSTOMER(Id), PRODUCT(Product_id, Product_no, Time_of_purchase)

- CUSTOMER(Id, {Phone numbers}, Address(City, Street)), PRODUCT(Product_id, Product_no, Customer_id, Time_of_purchase), with Cutomer_id $^{FK}\rightarrow$ CUSTOMER(Id)

- CUSTOMER(Id, City, Street), PHONES(Phone_number, Id), with Id $^{FK}\rightarrow$ CUSTOMER(Id), PRODUCT(Product_id, Product_no, Time_of_purchase, Buyer_id), with Buyer_id $^{FK}\rightarrow$ CUSTOMER(Id)

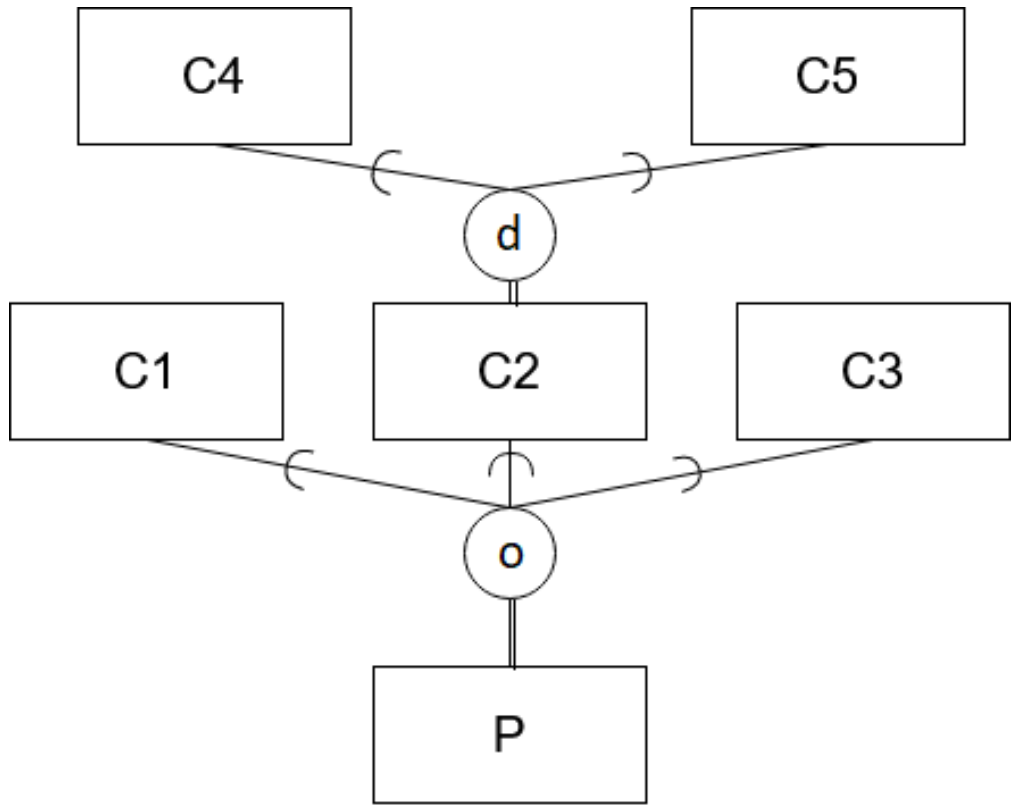
- CUSTOMER(Id, {Phone numbers}, Address, City, Street), BUYS(Id, Product_id, Time_of_purchase), with Id $^{FK}\rightarrow$ CUSTOMER(Id) and Product_id $^{FK}\rightarrow$ PRODUCT(Product_id), PRODUCT(Product_id, Product_no)

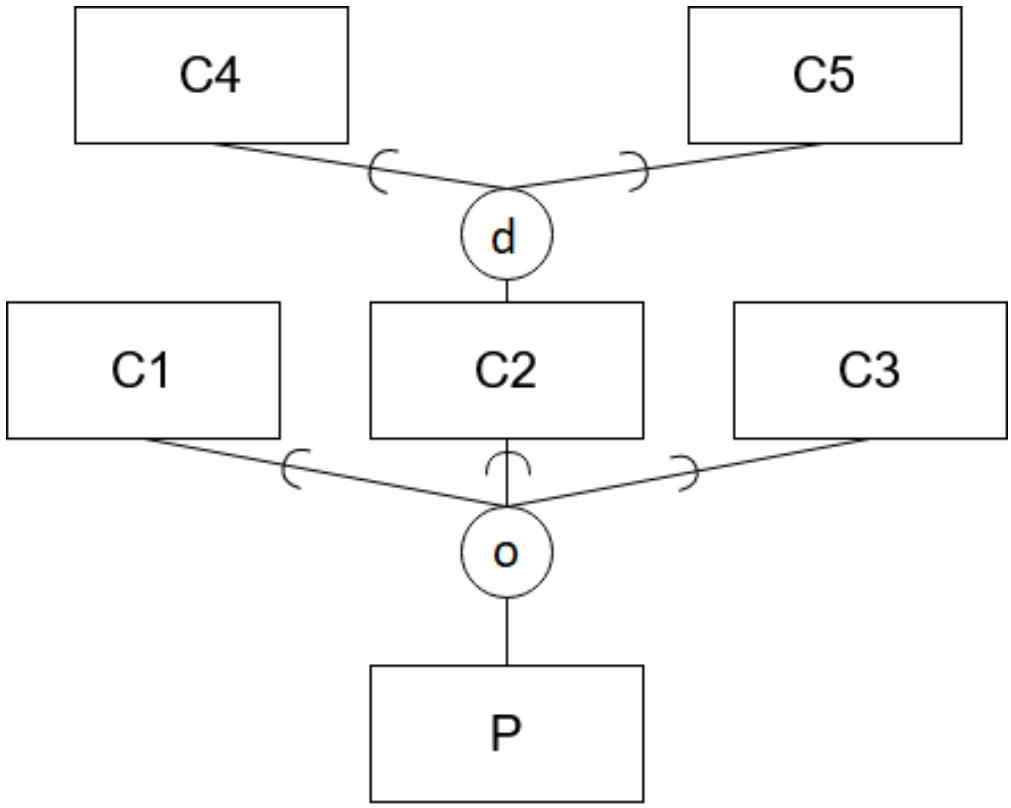
Maximum marks: 1

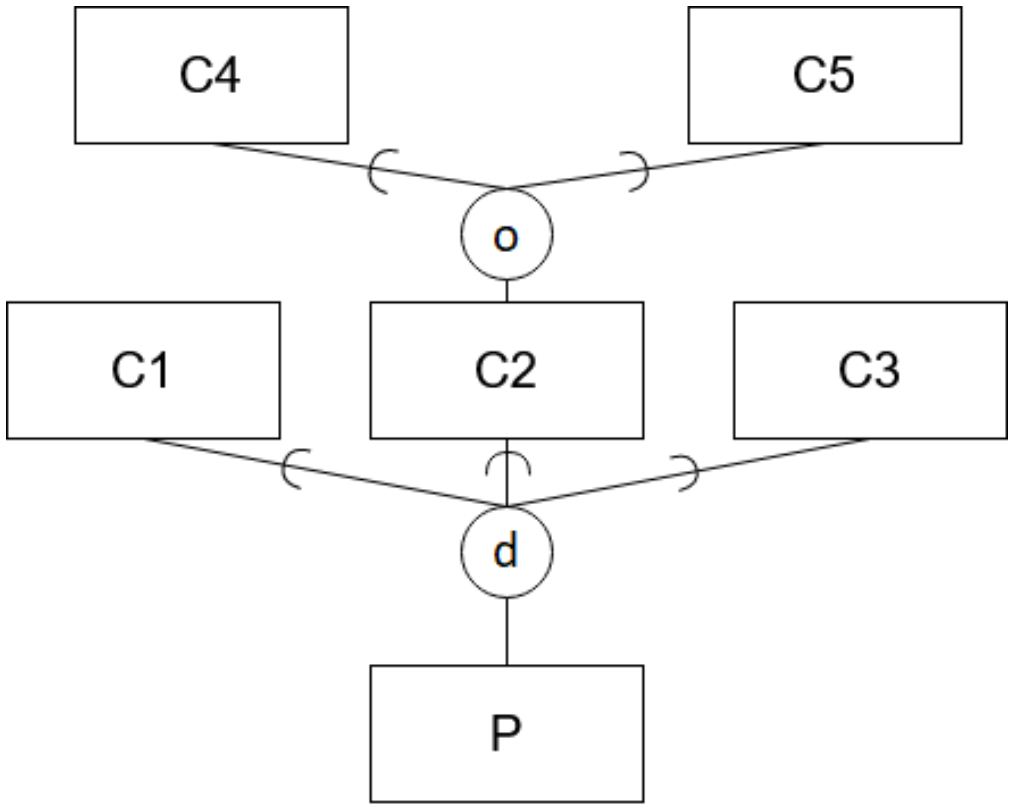
7 EER part 1

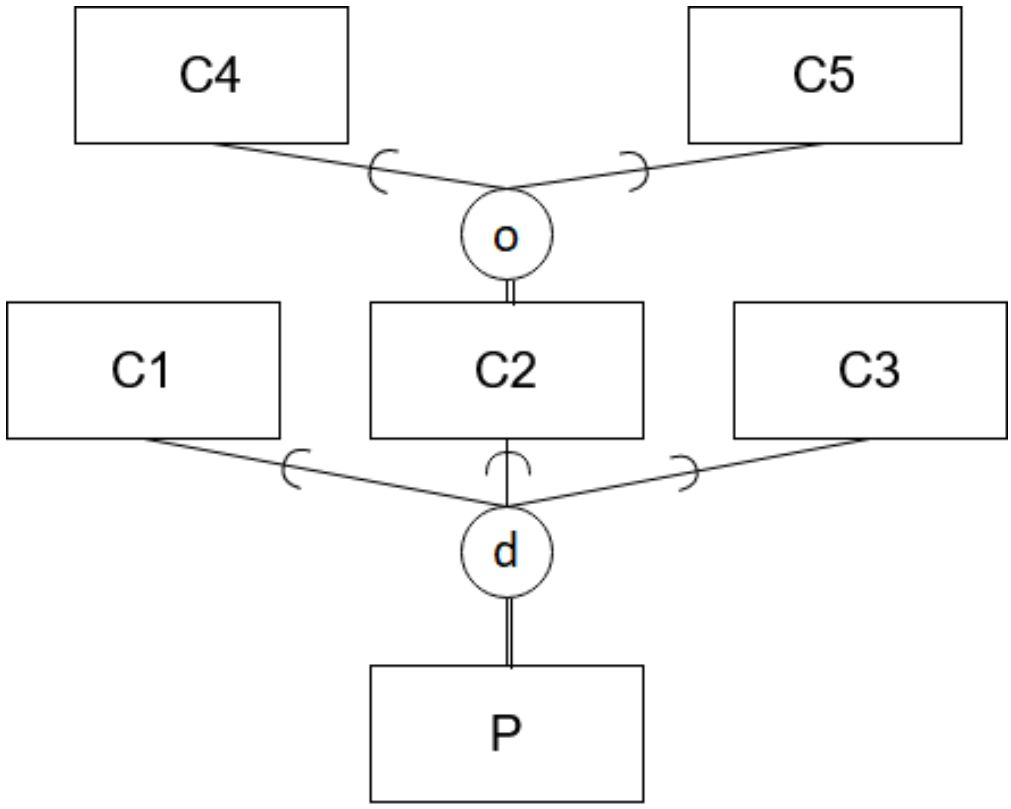
Entity type P can be *one or more of* the types C1, C2, C3 or possibly none of those. Which of the following EER diagrams corresponds to these specifications? (only a portion of the diagram has been visualized)!

Select one alternative:




C4
C5
d
C1
C2
C3
o
P

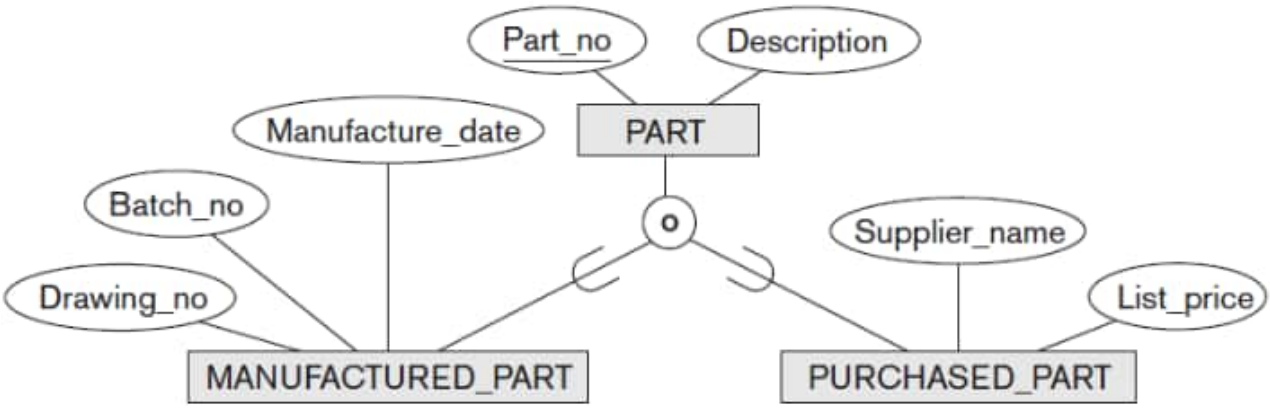

C4
C5
o
C1
C2
C3
d
P



Maximum marks: 1

8 EER to Relational

We want to convert the following diagram to the relational model. Choose the MOST appropriate answer from the following.



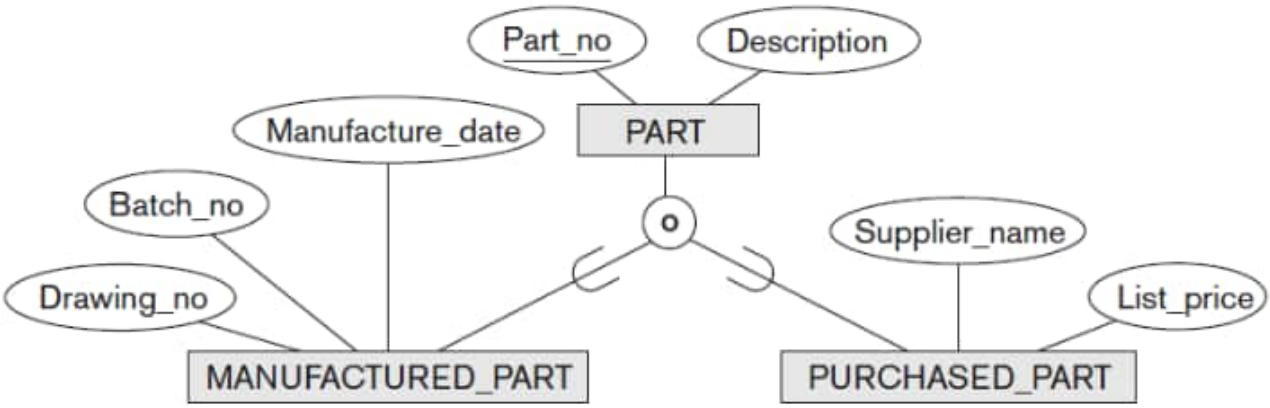
Select one alternative:

- PART(Part_no, Description, Manufacture_date, Batch_no, Drawing_no, Supplier_name, List_price, Manufactured_part_flag, Purchased_part_flag)
- MANUFACTURED_PART(Part_no, Manufacture_date, Batch_no, Drawing_no, Description), PURCHASED_PART(Part_no, Supplier_name, List_price, Description)
- PART(Part_no, Description, Manufacture_date, Batch_no, Drawing_no, Supplier_name, List_price, Type)
- PART(Part_no, Description), MANUFACTURED_PART(Manufacture_date, Batch_no, Drawing_no), PURCHASED_PART(Supplier_name, List_price)

Maximum marks: 1

9 ♥ EER Parts

Based on the EER model depicted below and the semantics of EER models, select all correct statements from the following.



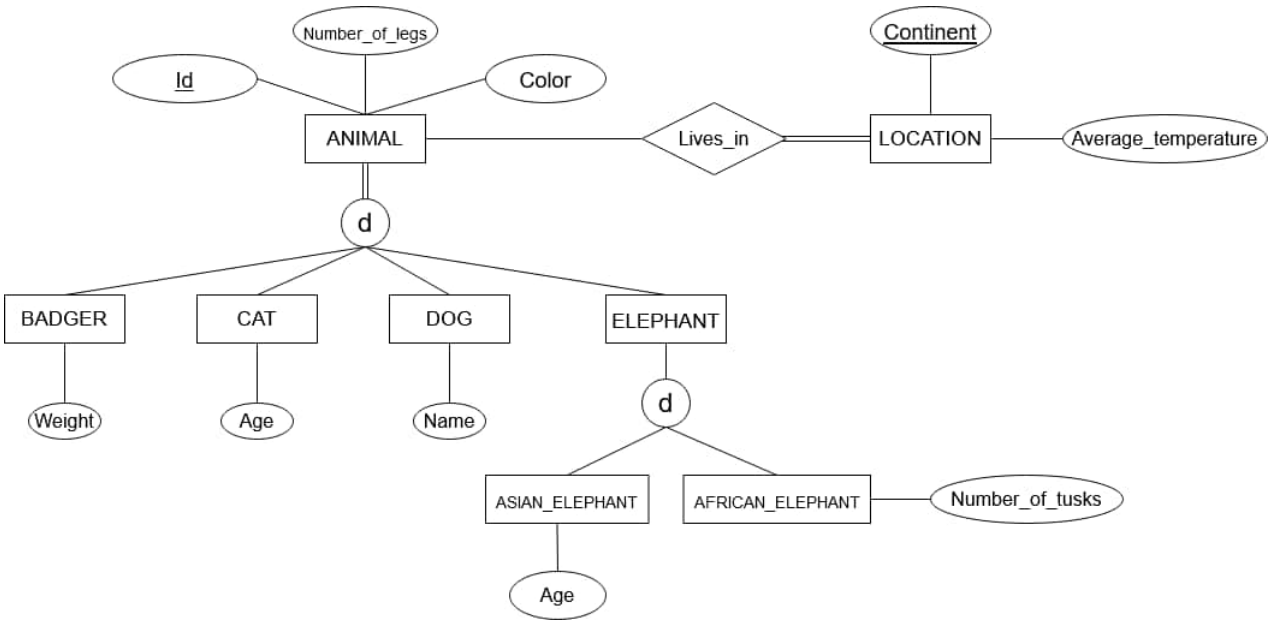
♥ Select one or more alternatives:

- [ ] PURCHASED_PART cannot inherit a primary key from PART.
- [ ] An entity type PART may be none of the indicated sub-classes
- [ ] All entity types have an attribute called “Description”
- [ ] An entity type PURCHASED_PART must also be a MANUFACTURED_PART

Maximum marks: 1

10 ♥ Interpreting EER

Select all true statements according to the following EER diagram:



♥ Select one or more alternatives:

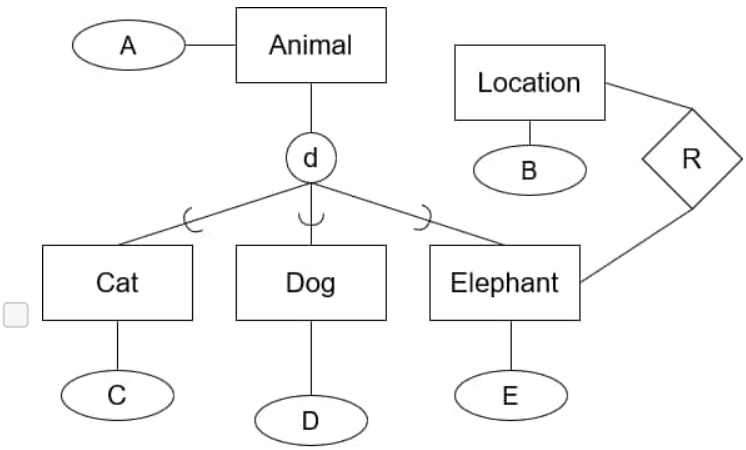
- [ ] There may be some subclass of ANIMAL that does not live in any continent.
- [ ] It is possible that an entity type ANIMAL is none of the entity types BADGER, CAT, DOG, or ELEPHANT.
- [ ] It is possible that an entity type ELEPHANT is none of the entity types ASIAN_ELEPHANT or AFRICAN_ELEPHANT
- [ ] The average temperature of the Continent Europe could be 90°C
- [ ] BADGER does not have an Id
- [ ] In every CONTINENT at least one of the following animals lives: BADGER, CAT, DOG, ELEPHANT, ASIAN_ELEPHANT, AFRICAN_ELEPHANT.

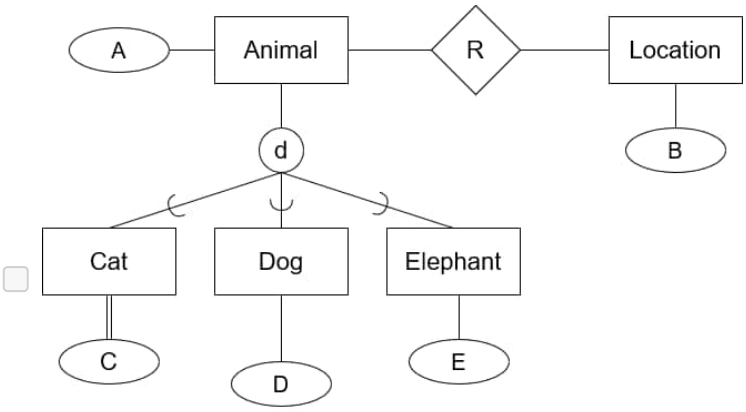
Maximum marks: 1

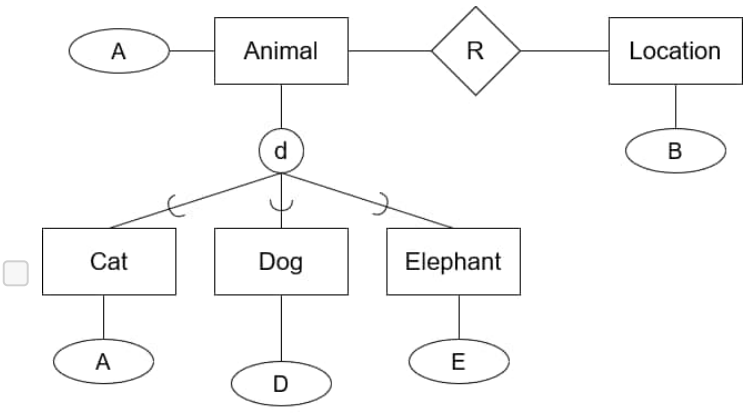
11 ♥ Valid EER

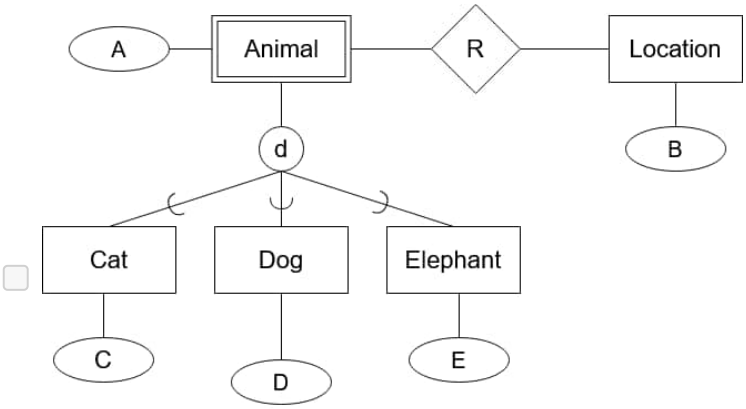
Which of the following are valid EER diagrams (disregard any real life, semantic interpretations of the model)?

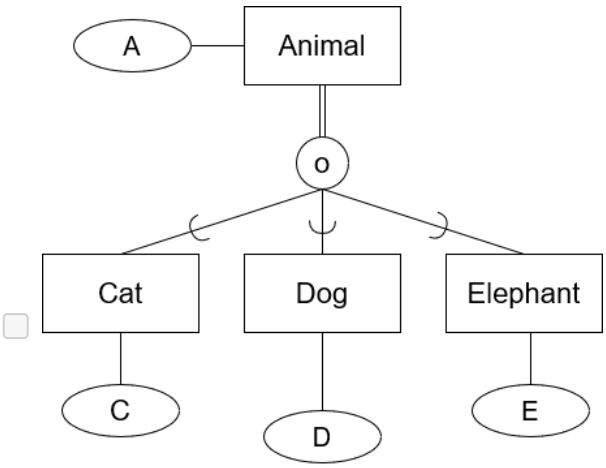
♥ Select one or more alternatives:











Maximum marks: 1

Consider the following populated database, with the two tables CAR and OWNER where VIN is the primary key for CAR and OwnerID is the primary key for OWNER. Also, CAR.OwnerID is a foreign key to OWNER.OwnerID.

CAR				
VIN	Year	Manufacturer	Color	OwnerID
1234	2001	Toyota	Red	1
4201	2017	Tesla	Silver	2
5678	2005	Ford	Black	3
9999	1999	Saab	Blue	1
1111	2010	Volvo	Green	NULL
2345	2015	Volvo	Gray	3

OWNER		
OwnerID	Name	Age
1	Mandy	19
2	George	45
3	Styrbjörn	39
4	Ingrid	80

Answer the questions on the right.

12 SQL

Which of the following SQL statements will return the average age of owners older than 20?

- SELECT Average_age
  FROM OWNER
  WHERE (Age > 20) AS Average_age;

- SELECT AVG(Age)
  FROM OWNER
  GROUP BY Age > 20;

- SELECT Age
  FROM OWNER
  WHERE Age IN AVG(Age);

- SELECT AVG(Age)
  FROM OWNER
  WHERE Age > 20;

Maximum marks: 1

13 SQL

Which of the following SQL statements will return the Manufacturer and Color of cars made before 2010?

SELECT Manufacturer, Color
FROM CAR
WHERE Year < 2010;

SELECT Manufacturer, Color
FROM CAR
WHERE Year IN (SELECT Car.Year < 2010);

SELECT Manufacturer, Color, Year < 2010
FROM CAR;

SELECT *
FROM CAR
WHERE Year < 2010;

Maximum marks: 1

14 SQL

Which of the following SQL statements will return the number of cars that are Volvos?

- SELECT COUNT(*)
  FROM CAR
  WHERE CAR.Manufacturer IN (CAR SELECT 'Volvo');

- SELECT COUNT('Volvo')
  FROM CAR.Manufacturer;

- SELECT COUNT(*)
  FROM CAR
  WHERE CAR.Manufacturer = 'Volvo';

- SELECT Number
  FROM COUNT(CAR.Manufacturer = 'Volvo') AS Number;

Maximum marks: 1

Consider the following populated database, with the two tables CAR and OWNER where VIN is the primary key for CAR and OwnerID is the primary key for OWNER. Also, CAR.OwnerID is a foreign key to OWNER.OwnerID.

CAR				
VIN	Year	Manufacturer	Color	OwnerID
1234	2001	Toyota	Red	1
4201	2017	Tesla	Silver	2
5678	2005	Ford	Black	3
9999	1999	Saab	Blue	1
1111	2010	Volvo	Green	NULL
2345	2015	Volvo	Gray	3

OWNER		
OwnerID	Name	Age
1	Mandy	19
2	George	45
3	Styrbjörn	39
4	Ingrid	80

Answer the questions on the right.

15 SQL

Which of the following SQL-statements will return a table where each row contains an owner ID and the most recent manufacturing year from which they own at least one car (several different cars can be made in the same year)? The table should contain the ID of all car owners and no IDs of people who do not own cars.

- SELECT OwnerID, Year
  FROM CAR
  WHERE OwnerID IS NOT NULL
  GROUP BY MAX(Year);

- SELECT OwnerID, MAX(Year)
  FROM CAR
  GROUP BY OwnerID;

- None of the other options.

- SELECT OwnerID, MAX(Year) AS Year_of_newest
  FROM CAR
  WHERE OwnerID IS NOT NULL
  GROUP BY OwnerID;

- SELECT OwnerID, Year
  FROM CAR
  WHERE YEAR IN (SELECT MAX(Year) FROM CAR);

Maximum marks: 1

16 SQL

Which of the following SQL-statements will return a list of unique names (i.e. no duplicates) of people that own a Toyota, a Saab, or a Tesla?

```
SELECT OWNER.Name
FROM OWNER, CAR
WHERE OWNER.OwnerID = CAR.OwnerID AND CAR.Manufacturer IN ('Toyota', 'Saab',
'Tesla') AND Owner.Name IS DISTINCT;
```

```
SELECT Names.Name
FROM
((SELECT Name
FROM OWNER, CAR
WHERE OWNER.OwnerID = CAR.OwnerID AND CAR.Manufacturer = 'Toyota')
UNION
(SELECT Name
FROM OWNER, CAR
WHERE OWNER.OwnerID = CAR.OwnerID AND CAR.Manufacturer = 'Saab')
UNION
(SELECT Name
FROM OWNER, CAR
WHERE OWNER.OwnerID = CAR.OwnerID AND CAR.Manufacturer = 'Tesla')) AS
Names;
```

```
SELECT DISTINCT(OWNER.Name)
FROM OWNER
WHERE OWNER.OwnerID = CAR.OwnerID AND CAR.Manufacturer IN ('Toyota', 'Saab',
'Tesla');
```

```
SELECT DISTINCT(OWNER.Name)
FROM OWNER, CAR
WHERE OWNER.OwnerID = CAR.OwnerID AND
FROM CAR
WHERE CAR.Manufacturer IN ('Toyota', 'Saab', 'Tesla');
```

Maximum marks: 1

17 SQL

Which of the following SQL-statements will return a list of OwnerIDs, of those that own more than one car?

- SELECT COUNT(CAR.OwnerID)
  FROM CAR, OWNER
  WHERE CAR.OwnerID > 1 AND CAR.OwnerId = OWNER.OwnerID
  GROUP BY CAR.OwnerID;

- SELECT OWNER.OwnerID
  FROM OWNER
  WHERE OWNER.OwnerID = CAR.OwnerID AND COUNT(OWNER.OwnerID) >= 2;

- SELECT Id
  FROM
  (SELECT OWNER.OwnerID AS Id, COUNT(*) AS NumCars
  FROM OWNER INNER JOIN CAR ON OWNER.OwnerID = CAR.OwnerID
  GROUP BY OWNER.OwnerID) AS T
  WHERE NumCars > 1;

- SELECT T.Id
  FROM
  (SELECT OWNER.OwnerID AS Id, COUNT(*) > 1 AS NumCars
  FROM OWNER, CAR) AS T;

Maximum marks: 1

18 SQL

Which of the following SQL-statements will return a list of manufacturers, without repetitions, who have made cars before 2000 or after 2005 in the database?

- SELECT DISTINCT Manufacturer
  WHERE Year < 2000 OR Year > 2005;

- SELECT Manufacturer
  FROM CAR
  WHERE Year < 2000 OR Year > 2005;

- (SELECT Manufacturer FROM CAR WHERE Year < 2000)
  UNION ALL
  (SELECT Manufacturer FROM CAR WHERE Year > 2005);

- (SELECT Manufacturer FROM CAR WHERE Year < 2000)
  UNION
  (SELECT Manufacturer FROM CAR WHERE Year > 2005);

Maximum marks: 1

19 SQL

We want to get a list of the names of owners of cars manufactured after 2004. Which of the following SQL-statements is correct?

- SELECT DISTINCT Name
  FROM CAR LEFT OUTER JOIN OWNER ON CAR.OwnerID = OWNER.OwnerID
  WHERE Year > 2004;

- SELECT DISTINCT Name
  FROM OWNER, CAR
  WHERE CAR.OwnerID = OWNER.OwnerID AND CAR.Year > 2004;

- SELECT DISTINCT Name
  FROM OWNER
  WHERE OWNER.OwnerID = CAR.OwnerID AND CAR.Year > 2004;

- SELECT DISTINCT Name
  FROM CAR
  WHERE Year > 2004;

Maximum marks: 1

20 ♥ Normal forms

Consider a relation R(A, B, C, D) where A is the primary key and the following full functional dependencies hold:

- $A \rightarrow B$
- $A \rightarrow C$
- $A \rightarrow D$
- $\{B, C\} \rightarrow D$

Which of the following hold?

♥ Select one or more alternatives:

- [ ] R is in 1NF but not in 2NF
- [ ] R is in 3NF but not in 2NF
- [ ] R is in 2NF but not in 1NF
- [ ] R is in 3NF
- [ ] R is in 2NF but not in 3NF

Maximum marks: 1

21 Giving read access

Alice wants to give a **read** access to a set of attributes A_1, A_2, …, A_n of table T to Bob. T contains more attributes than this set of attributes. Which of these options is most suitable?

Select one alternative:

- GRANT SELECT ON T TO Bob
- Create a view V with attributes A_1, A_2, …, A_n and then grant SELECT privileges on V to Bob.
- Temporarily grant access to table T to Bob and revoke it after it is no longer necessary.
- GRANT SELECT ON T(A_1, A_2, …, A_n) TO Bob
- Create a new table T_{new} with only the attributes A_1, A_2, …, A_n and grant select privileges to Bob.

Maximum marks: 1

22 ♥ Functional Dependencies

The following table shows the current state of a relation, where C_1 is the **key** of the relation. Which of the following options are correct? (Select **all** correct choices)

C_1	C_2	C_3	C_4	C_5
1	1	George	Anna	42
2	1	George	Chris	42
3	1	George	Anna	42
4	2	John	Nek	2
5	2	John	Chris	1
6	3	Marina	Anna	3

♥ Select one or more alternatives:

- ☐ The functional dependency $\{C_3, C_4\} \rightarrow C_5$ may hold.
- ☐ We cannot be certain whether the functional dependency $C_2 \rightarrow C_3$ holds.
- ☐ The functional dependency $C_3 \rightarrow C_4$ may hold.
- ☐ The functional dependency $C_1 \rightarrow C_4$ holds.
- ☐ All attributes are functionally dependent on C_1.
- ☐ The functional dependency $C_3 \rightarrow C_5$ may hold.

Maximum marks: 1

Consider the database below. Where, the primary keys are underlined and the foreign key (FK) relationships are:

- PROJECT.Dnum is FK ref. DEPARTMENT.Dnumber;
- PROJECT.Plocation is FK ref. DEPT_LOCATIONS.Dlocation;
- DEPT_LOCATIONS.Dnumber is FK ref. DEPARTMENT.Dnumber;
- DEPARTMENT.Mgr_ssn is FK ref. EMPLOYEE.Ssn;
- EMPLOYEE.Super_ssn is FK ref. EMPLOYEE.Ssn;
- EMPLOYEE.Dno is FK ref. DEPARTMENT.Dnumber.

Suppose each of the following update operations is applied directly to this database:

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

PROJECT

Pname	Pnumber	Plocation	Dnum
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

DEPT_LOCATIONS

Dnumber	Dlocation
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

23 ♥ Constraint violations

Choose the correct statements regarding the integrity constraints (i.e. key constraints, entity and referential integrity constraint) of the following operation when applied to the database on the left.

INSERT INTO DEPARTMENT VALUES (NULL, 6, '123456789', '2021-01-13');

♥ Select one or more alternatives:

- [ ] No constraint violations.
- [ ] It violates the key constraint.
- [ ] It violates entity integrity
- [ ] It violates referential integrity.

Maximum marks: 1

Consider the database below. Where, the primary keys are underlined and the foreign key (FK) relationships are:

- PROJECT.Dnum is FK ref. DEPARTMENT.Dnumber;
- PROJECT.Plocation is FK ref. DEPT_LOCATIONS.Dlocation;
- DEPT_LOCATIONS.Dnumber is FK ref. DEPARTMENT.Dnumber;
- DEPARTMENT.Mgr_ssn is FK ref. EMPLOYEE.Ssn;
- EMPLOYEE.Super_ssn is FK ref. EMPLOYEE.Ssn;
- EMPLOYEE.Dno is FK ref. DEPARTMENT.Dnumber.

Suppose each of the following update operations is applied directly to this database:

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

PROJECT

Pname	Pnumber	Plocation	Dnum
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

DEPT_LOCATIONS

Dnumber	Dlocation
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

24 ♥ Constraint violations

Choose the correct statements regarding the integrity constraints (i.e. key constraints, entity and referential integrity constraint) of the following operation when applied on the database shown to the left.

DELETE FROM EMPLOYEE WHERE EMPLOYEE.SSN = '333445555'

♥ Select one or more alternatives:

- [ ] It violates the key constraint because employee with primary key 333445555 is deleted.
- [ ] It violates referential integrity because Super_ssn is NULL for the employee with Ssn 888665555.
- [ ] No constraint violations.
- [ ] It violates both key and referential integrity constraint.
- [ ] It violates referential integrity constraint because Mgr_ssn in the DEPARTMENT table has a value of 333445555 for one of its rows.
- [ ] It violates entity integrity constraint because Super_ssn is NULL for the employee with Ssn 888665555.

Maximum marks: 1

25 ♥ Candidate Keys

Consider a relation R(A, B, C, D, E, F) in 1NF where the following functional dependencies hold:

- {A, B} → {C}
- {C} → {D}
- {D, F} → {E}

Which of the following are candidate keys?

♥ Select one or more alternatives:

- [ ] None of the alternatives are candidate keys.
- [ ] {C}
- [ ] {C, D, E, F}
- [ ] {D, F}
- [ ] {A, B}

Maximum marks: 1

26 ♥ Normalization

Which of the following databases contains all the information contained in the following relation:

R(A, B, C, D, E, F) in 1NF (at least) where the following functional dependencies hold:

- {A, B} → {C, D, E, F}
- {C} → {A}
- {D} → {B}

and has all relations in 3NF (not showing the primary and foreign key constraints)?

♥ Select one or more alternatives

- ☐ R_1(A, B), R_2(C,D,E,F)
- ☐ R_1(A, B, E, F), R_2(C, D)
- ☐ None of these databases.
- ☐ R(A,B,C,D,E,F)
- ☐ R_1(A,B), R_2(C,D,E,F), R_3(C,A), R_4(D,B)

Maximum marks: 1

27 ♥ Serializability

For the following sets of transactions T_1, T_2, and T_3, which of the schedules are (conflict) serializable?:

♥ **Select one or more alternatives:**

T1	T2	T3
read_item(A);		
		read_item(C);
	read_item(A);	
A := A * 5;		
write_item(A);		
read_item(B);		
B := B – 10;		
write_item(B);		
	A := A + 10	
		C := C / 10;
		write_item(C)
	write_item(A);	
		read_item(A)
		A := A / 5;
		write_item(A)

T1	T2	T3
	read_item(A);	
	A := A + 10	
	write_item(A);	
read_item(A);		
A := A * 3;		
write_item(A);		
read_item(B);		
B := B – 5;		
write_item(B);		
		read_item(C);
		C := C / 5;
		write_item(C)
		read_item(A)
		A := A / 3;
		write_item(A)

T1	T2	T3
read_item(A);		
A := A * 5;		
write_item(A);		
	read_item(A);	
read_item(B);		
B := B – 10;		
write_item(B);		
	A := A + 10	
		read_item(C);
		C := C / 10;
		write_item(C)
	write_item(A);	
		read_item(A)
		A := A / 5;
		write_item(A)

T1	T2	T3
	read_item(A);	
	A := A + 10	
		read_item(C);
		C := C / 5;
		write_item(C)
		read_item(A)
read_item(A);		
A := A * 3;		
write_item(A);		
read_item(B);		
B := B – 5;		
write_item(B);		
	write_item(A);	
		A := A / 3;
		write_item(A)

Maximum marks: 1