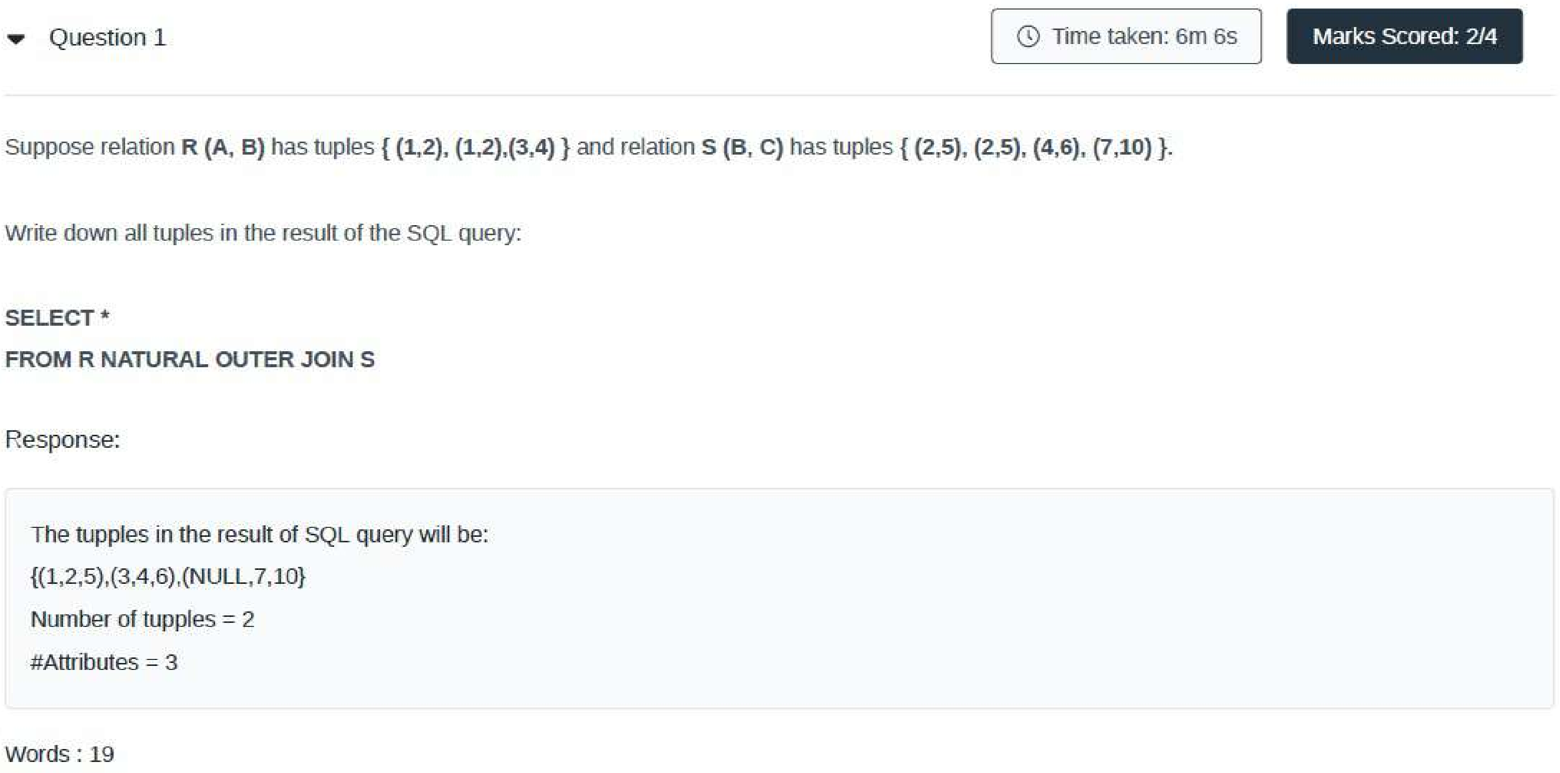


Suppose relation **R (A, B)** has tuples **{ (1,2), (1,2),(3,4) }** and relation **S (B, C)** has tuples **{ (2,5), (2,5), (4,6), (7,10) }**.

Write down all tuples in the result of the SQL query:

SELECT *

FROM R NATURAL OUTER JOIN S

Response:

The tupples in the result of SQL query will be:

{(1,2,5),(3,4,6),(NULL,7,10}

Number of tupples = 2

#Attributes = 3

Words : 19

▼ Question 2

Not Attempted

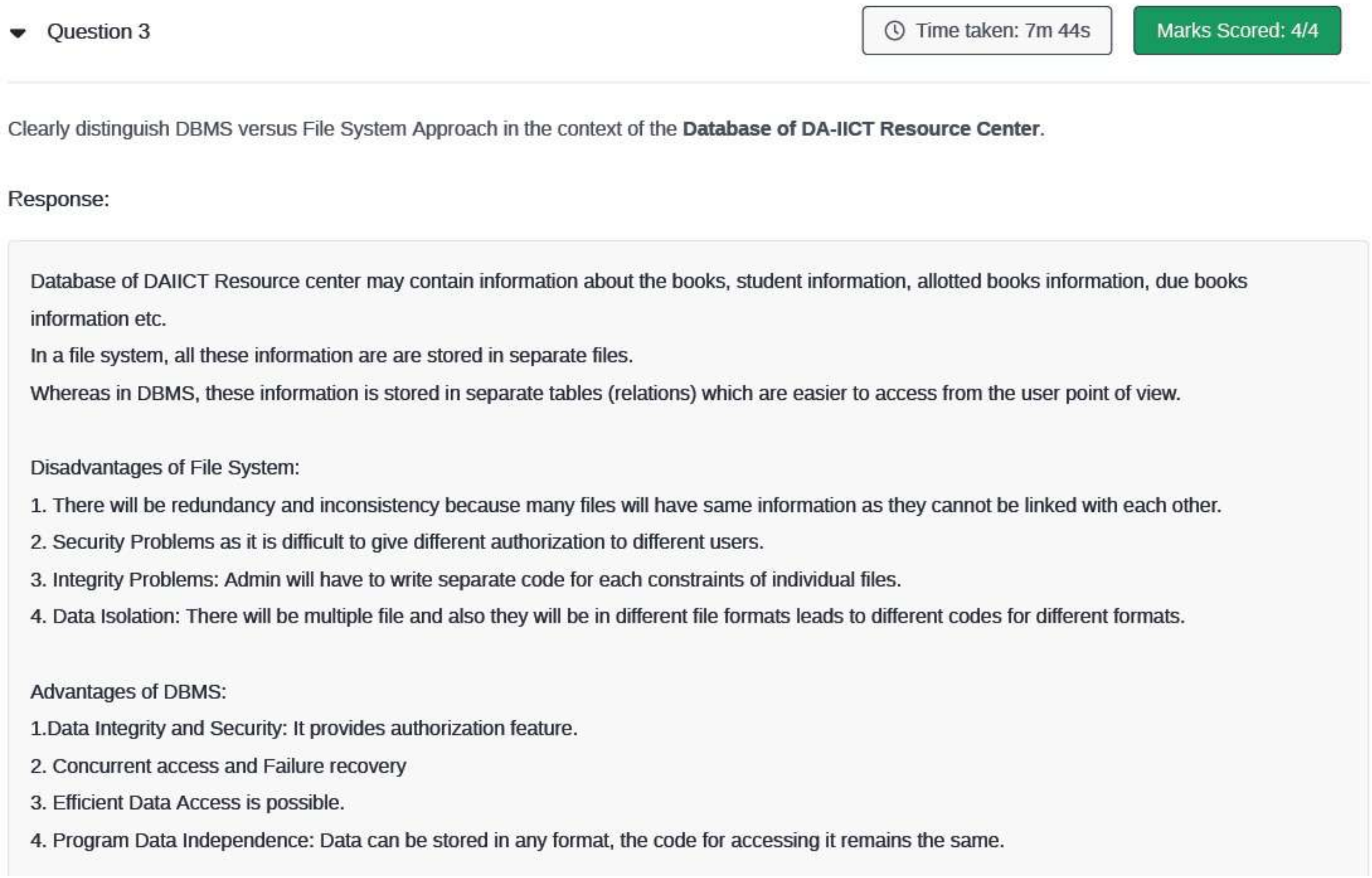
Define and Explain the following in the context of the Hierarchy in the E-R Model of **Database of DA-IICT Resource Center**:

a) Covering Constraint

b) Overlapping Constraint

Response:

Words : 0



Question 3

Time taken: 7m 44s

Marks Scored: 4/4

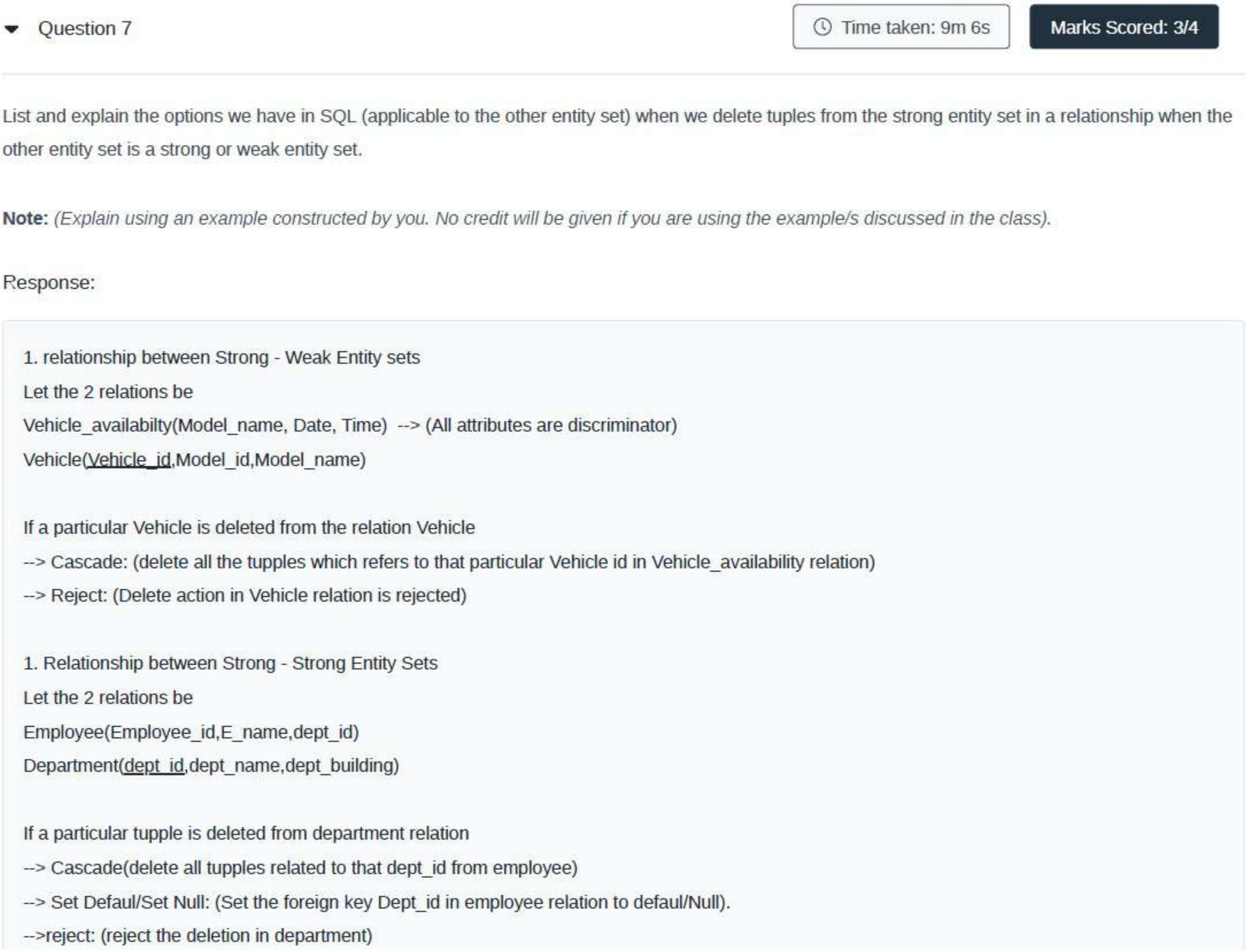
Clearly distinguish DBMS versus File System Approach in the context of the **Database of DA-IICT Resource Center**.

Response:

Database of DAIICT Resource center may contain information about the books, student information, allotted books information, due books information etc.
In a file system, all these information are are stored in separate files.
Whereas in DBMS, these information is stored in separate tables (relations) which are easier to access from the user point of view.

Disadvantages of File System:
1. There will be redundancy and inconsistency because many files will have same information as they cannot be linked with each other.
2. Security Problems as it is difficult to give different authorization to different users.
3. Integrity Problems: Admin will have to write separate code for each constraints of individual files.
4. Data Isolation: There will be multiple file and also they will be in different file formats leads to different codes for different formats.

Advantages of DBMS:
1.Data Integrity and Security: It provides authorization feature.
2. Concurrent access and Failure recovery
3. Efficient Data Access is possible.
4. Program Data Independence: Data can be stored in any format, the code for accessing it remains the same.



List and explain the options we have in SQL (applicable to the other entity set) when we delete tuples from the strong entity set in a relationship when the other entity set is a strong or weak entity set.

Note: *(Explain using an example constructed by you. No credit will be given if you are using the example/s discussed in the class).*

Response:

1. relationship between Strong - Weak Entity sets

Let the 2 relations be

Vehicle_availabilty(Model_name, Date, Time) --> (All attributes are discriminator)

Vehicle(Vehicle_id,Model_id,Model_name)

If a particular Vehicle is deleted from the relation Vehicle

--> Cascade: (delete all the tupples which refers to that particular Vehicle id in Vehicle_availability relation)

--> Reject: (Delete action in Vehicle relation is rejected)

1. Relationship between Strong - Strong Entity Sets

Let the 2 relations be

Employee(Employee_id,E_name,dept_id)

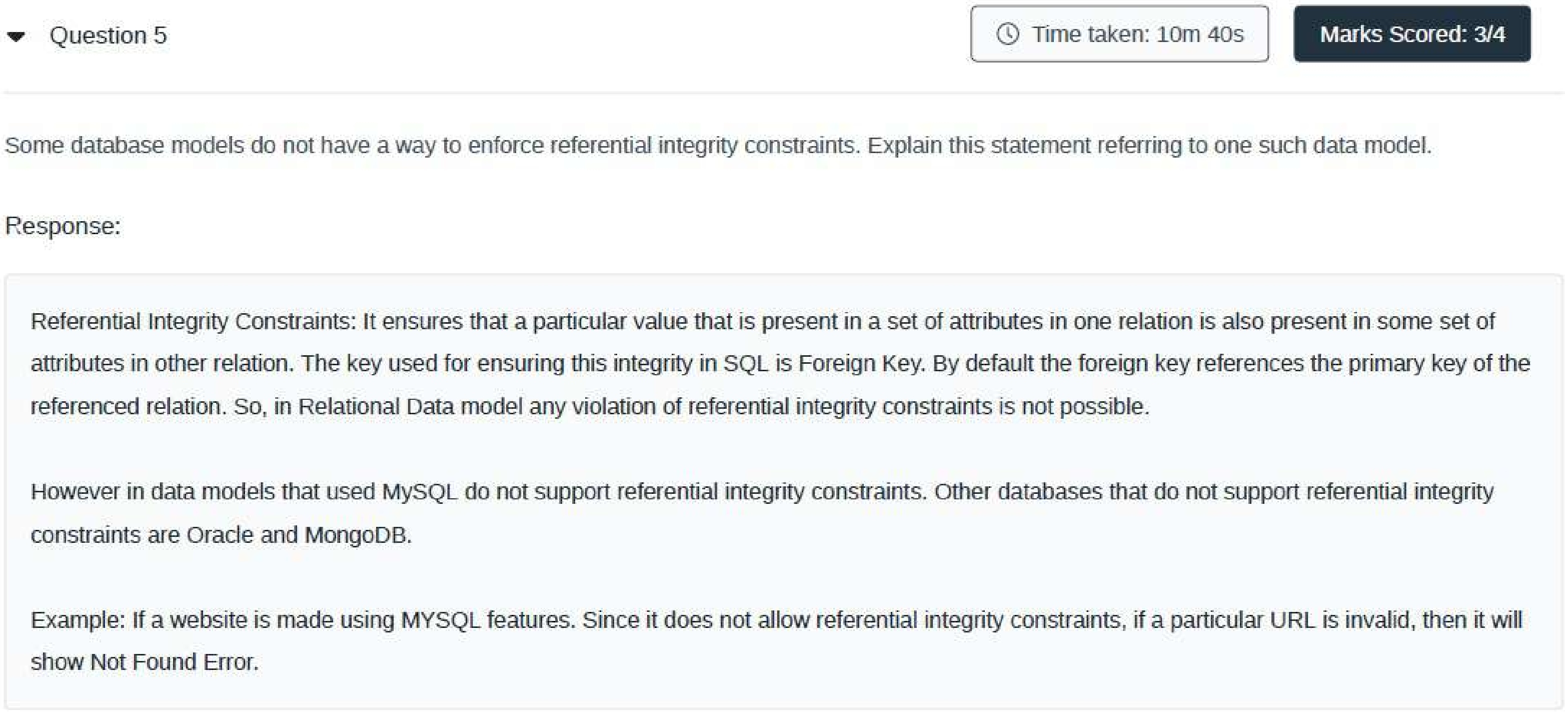
Department(dept_id,dept_name,dept_building)

If a particular tupple is deleted from department relation

--> Cascade(delete all tupples related to that dept_id from employee)

--> Set Defaul/Set Null: (Set the foreign key Dept_id in employee relation to defaul/Null).

-->reject: (reject the deletion in department)



Question 5

Time taken: 10m 40s

Marks Scored: 3/4

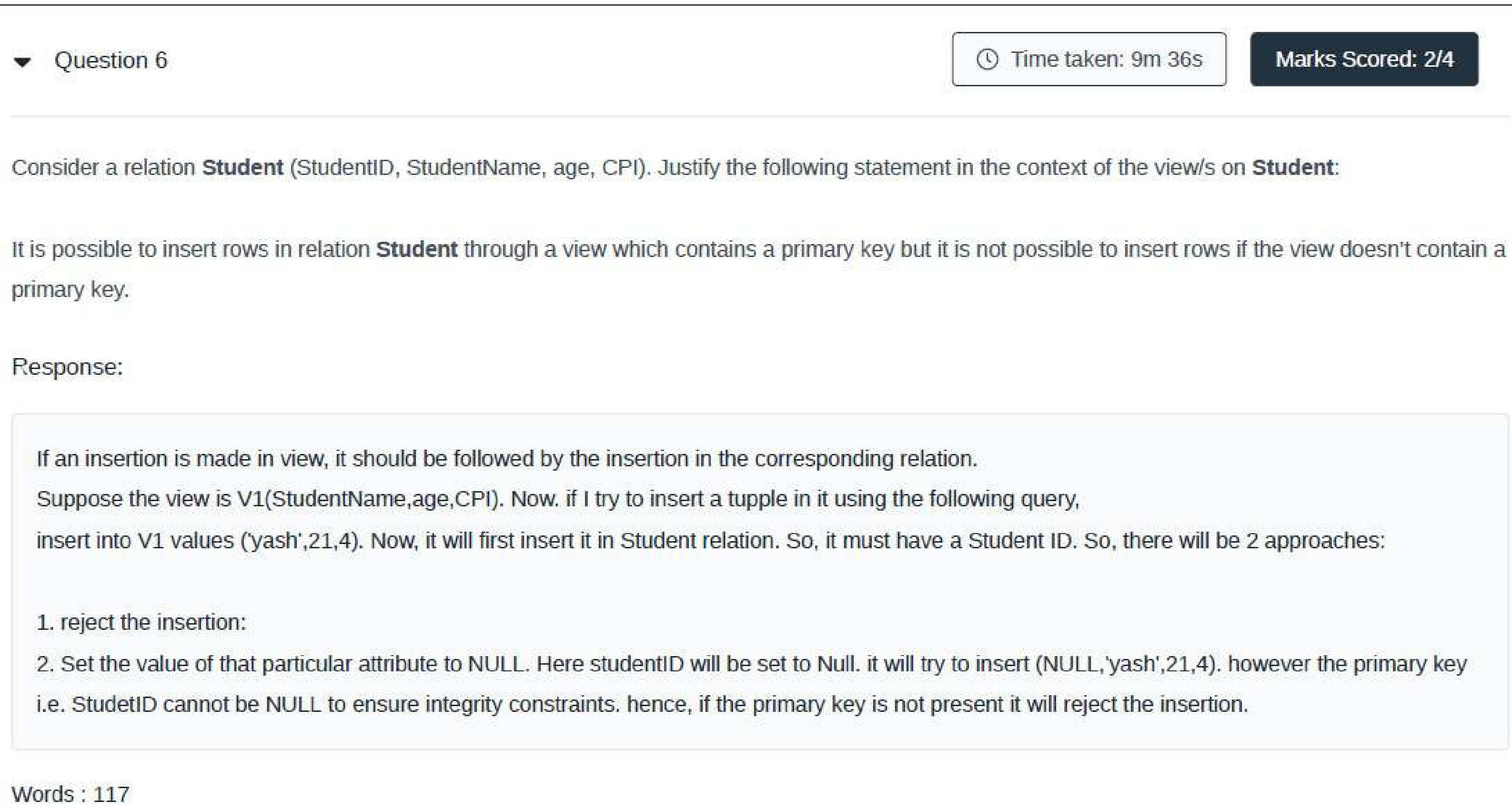
Some database models do not have a way to enforce referential integrity constraints. Explain this statement referring to one such data model.

Response:

Referential Integrity Constraints: It ensures that a particular value that is present in a set of attributes in one relation is also present in some set of attributes in other relation. The key used for ensuring this integrity in SQL is Foreign Key. By default the foreign key references the primary key of the referenced relation. So, in Relational Data model any violation of referential integrity constraints is not possible.

However in data models that used MySQL do not support referential integrity constraints. Other databases that do not support referential integrity constraints are Oracle and MongoDB.

Example: If a website is made using MYSQL features. Since it does not allow referential integrity constraints, if a particular URL is invalid, then it will show Not Found Error.



Question 6

Time taken: 9m 36s

Marks Scored: 2/4

Consider a relation **Student** (StudentID, StudentName, age, CPI). Justify the following statement in the context of the view/s on **Student**:

It is possible to insert rows in relation **Student** through a view which contains a primary key but it is not possible to insert rows if the view doesn't contain a primary key.

Response:

If an insertion is made in view, it should be followed by the insertion in the corresponding relation.
Suppose the view is V1(StudentName,age,CPI). Now. if I try to insert a tupple in it using the following query,
insert into V1 values ('yash',21,4). Now, it will first insert it in Student relation. So, it must have a Student ID. So, there will be 2 approaches:

1. reject the insertion:
2. Set the value of that particular attribute to NULL. Here studentID will be set to Null. it will try to insert (NULL,'yash',21,4). however the primary key i.e. StudetID cannot be NULL to ensure integrity constraints. hence, if the primary key is not present it will reject the insertion.

Words : 117