1.

T1 -> R1(A) W1(C) C1
T2 -> R2(B) R2(C) C2
T3 -> W3(A) R3(A) W3(C) W3(B) R3(B) C3
T4 -> R4(C) W4(C) R4(A) W4(A) C4

a. View-serializable
   To satisfy the criteria for view serializability, T2 should occur before T1 and T1 should occur before T4. This violates the criteria. Therefore, the schedule is not view serializable.

b. Conflict-serializable
   The given schedule is not conflict serializable because there are two cycles : T1 -> T4 -> T1 and T1 -> T4 -> T2 -> T1.

c. Recoverable
   For each pair of transactions Ti and Tj , such that Tj reads the value written by Ti, and Ti is committed before Tj, then the schedule is called recoverable.
   The given schedule is recoverable because it follows the above property.

d. Cascadeless
   For each pair of transactions Ti and Tj , such that Tj reads the value written by Ti, and Ti is committed before Tj reads, then the schedule is called cascadeless. Every cascadeless schedule is also recoverable schedule. Since the above property is satisfied, there are no cascading rollbacks. Therefore, we can say that the above schedule is cascadeless.

e. Strict
   A schedule is said to be strict, if you have two transactions *T1* and *T2*, such that if a write operation of *T1* comes before a conflicting operation of *T2*(read or write), then the commit event of T1 will also come before that conflicting operation of T2.
   Since, the above property is satisfied, the given schedule is strict.

2.

T1 -> R1(C) W1(A) W1(B) C1
T2 -> R2(C) W2(C) W2(A) C2
T3 -> W3(A) W3(B) R3(B) C3
T4 -> W4(A) R4(B) R4(B) C4

a. View-serializable
   To satisfy the property of view serializability:
   T1 should occur before T2
   T3 should occur before T4
   T2 should occur at last since it write the final values of C and A.
   The above schedule is view serializable. The equivalent view serializable schedule is T3,T4,T1,T2

b. Conflict-serializable
   It is conflict serializable because there is no cycle amongst the 4 transactions.

c. Recoverable

The given schedule is recoverable because W3(B) occurs before R4(B) and T3 is committed before T4. It satisfies the property of recoverability.

d. Cascadeless

T3 and T4 do not follow the property of cacasdeless schedule. T3 is not committed before read of T4. Therefore this schedule is not cascadeless.

e. Strict

The given schedule is not strict because of T3 is not committed before W4(A).

3.

T1 -> R1(B) W1(A) C1
T2 -> R2(A) R2(B) C2
T3 -> W3(A) R3(C) C3
T4 -> R4(A) W4(A) R4(A) C4

a. View-serializable

It is not view serializable because T1 should occur before T4 and T4 should occur before T1.

b. Conflict-serializable

It is not conflict serializable because there is a cycle between T1 -> T4 and T4 -> T1.

c. Recoverable

It is recoverable because the following property of recoverability is satisfied:-
Wi(x)…Rj(x)…Ci…Cj

d. Cascadeless

It is cascadeless because T1 is committed before R4(A) . So it doesn't violate the property of cascadeless schedules.

e. Strict

It is not strict because T1 is not committed before W4(A)

4.

T1 -> W1(A) R1(B) C1
T2 -> R2(C) W2(B) C2
T3 -> W3(C) R3(A) R3(C) C3
T4 -> W4(C) R4(A) C4

a. View-serializable

It is not view serializable because T4 should occur before T2, T2 should occur before T1, T4 and T3 should occur after T1, which is not possible.

b. Conflict-serializable
Since there is a cycle between T1 -> T4 -> T2 -> T1, the given schedule is not conflict serializable.

c. Recoverable
It is not recoverable W2(B)…R1(B)…C1 i.e T1 is committed before T2.

d. Cascadeless
It is not cascadeless because T4 is not committed before R2(C).

e. Strict
It is not strict because T4 is not committed before R2(C).

5.

T1 -> R1(B) W1(B) C1
T2 -> R2(B) R2(C) R2(B) W2(A) W2(A) C2
T3 -> W3(B) R3(B) W3(C) R3(C) C3

a. View-serializable
For the schedule to be view serializable T1 and T2 should occur before T3 and T2 should occur before T3. Therefore, we can say that the given schedule is not view serializable.

b. Conflict-serializable
It is not conflict serializable because there is cycle between T2 -> T3 -> T2.

c. Recoverable
It is not recoverable because T2 reads B, which is written by T3 and T2 is committed before T3. This violated the property of recoverability.

d. Cascadeless
It is not cascadeless because T1 is not committed before write of T2.

e. Strict
It is not strict because T1 is not committed before write of T2.

6.

T1 -> R1(A) R1(B) C1
T2 -> R2(A) W2(B) R2(C) W2(C) R2(A) W2(C) R2(A) C2
T3 -> W3(C) R3(A) C3

a. View-serializable
Since the schedule is conflict serializable, it is also view serializable.
In equivalent view serializable schedule, T3 and T1 should occur before T2.
Equivalent view serializable schedule is T1,T3,T2 and T3,T1,T2.

b. Conflict-serializable

The schedule is conflict serializable because there are conflicts but there are no cycles in the schedule.

c. Recoverable
It recoverable because it follows the given property:-
Wi(x)…Rj(x)…Ci…Cj

d. Cascadeless
It is cascadeless because T3 is committed before T2 is committed.

e. Strict
It is strict because T3 is committed before T2 reads or writes C, which is initially written by T3.

7.

T1 -> W1(A) W1(C) C1
T2 -> W2(C) R2(A) W2(B) R2(C) W2(B) W2(A) C2
T3 -> W3(A) W3(A) R3(B) W3(A) R3(B) W3(B) R3(A) C3

a. View-serializable
The given schedule is view serializable and the equivalent view serializable schedule is T1, T3, T2.

b. Conflict-serializable
Since, there is cycle between T1 -> T3 -> T1. Therefore, the given schedule is not conflict serializable.

c. Recoverable
It recoverable because it follows the given property:-
T_j reads data item, previously written by T_i the commit operation of T_i appears before the read operation of T_j.

d. Cascadeless
It cascadeless because it follows the given property:-
Wi(x)…Ci..Rj(x)… Cj

e. Strict
It is not strict because W1(A)…W3(A)… To be strict, T1 should have been committed before write of A by T3.

8.

T1 -> W1(B) R1(B) C1
T2 -> R2(A) R2(B) C2
T3 -> R3(B) R3(A) W3(A) C3

a. View-serializable
To be view serializable following conditions need ti be satisfied:-
T1 occurs before T2
T2 occurs before T3
T3 occurs before T1

It is not possible. Therefore, the given schedule is not view serializable.

b. Conflict-serializable
   There is a cycle between T1 -> T2 -> T3 -> T1. Therefore, the given schedule is not conflict serializable.

c. Recoverable
   It is not recoverable because W1(B)..R2(B)..C2 i.e. T2 is committed before committing T1.

d. Cascadeless
   It is not cascadeless because T2 is committed before committing T1.

e. Strict
   It is not strict because T1 is not committed before R2(B).

2.a.

X-Path:

Company//Factory//Machine[@code>’K101’]/@id

JSON:

//Factory/*/Machine/*[code>’K101’]/id

2.b.

X-Path:

Company//ProductType[@title=’Carpet’][@size<=50]/Repairer[2]

JSON:

//key(“Product Type”)/*[title=”Carpet”][size<=50]/Repairer/*[1]

2.c.

X-Path:

Company//ProductType/Product[@started<’2015-12-31’]//Repairer[starts-with(text(),’Divya’)]

JSON:

//key(“Product Type”)/*[Repairer/*[starts-with(name,’Divya’)]]/Product/*[completed<=’2015-12-31’]

/started

2.d.

X-Path:

Company//Factory[contains(@address,’Main Street’)]/Machine[1][contains(@code,’01’)]

2.e.

X-Path:

Company//ProductType[starts-with(@description,'Shag')][not(Repairer[@qualification='expert'])[not (Repairer[@qualification=’intermediate’])]/@title