

Assignment 3- Transaction Management and MYSQL Spatial Extension

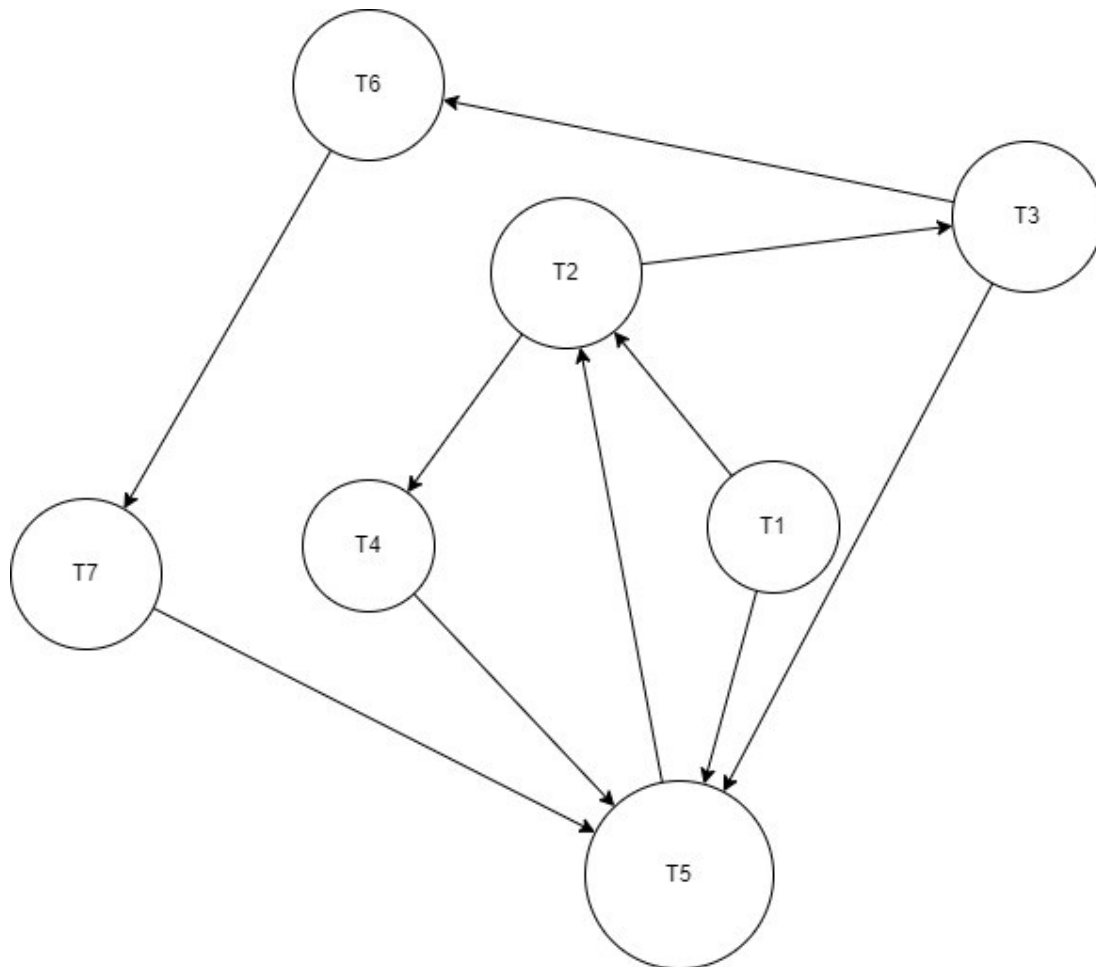
1.i) Explain the problem with the above described scenario involving multiple transactions running concurrently on the Student table.

T1 had a default read committed isolation level. T2 updated the student table data which was read by T1. T3 read the data which was modified by T2 which resulted in an unrepeatable read.

ii) Explain how you would solve the problem from (i) using your knowledge of transaction management.

To overcome this problem a serializable isolation level must be used to block unrepeatable reads

2.i) Produce a wait-for-graph for the above transaction scenario.

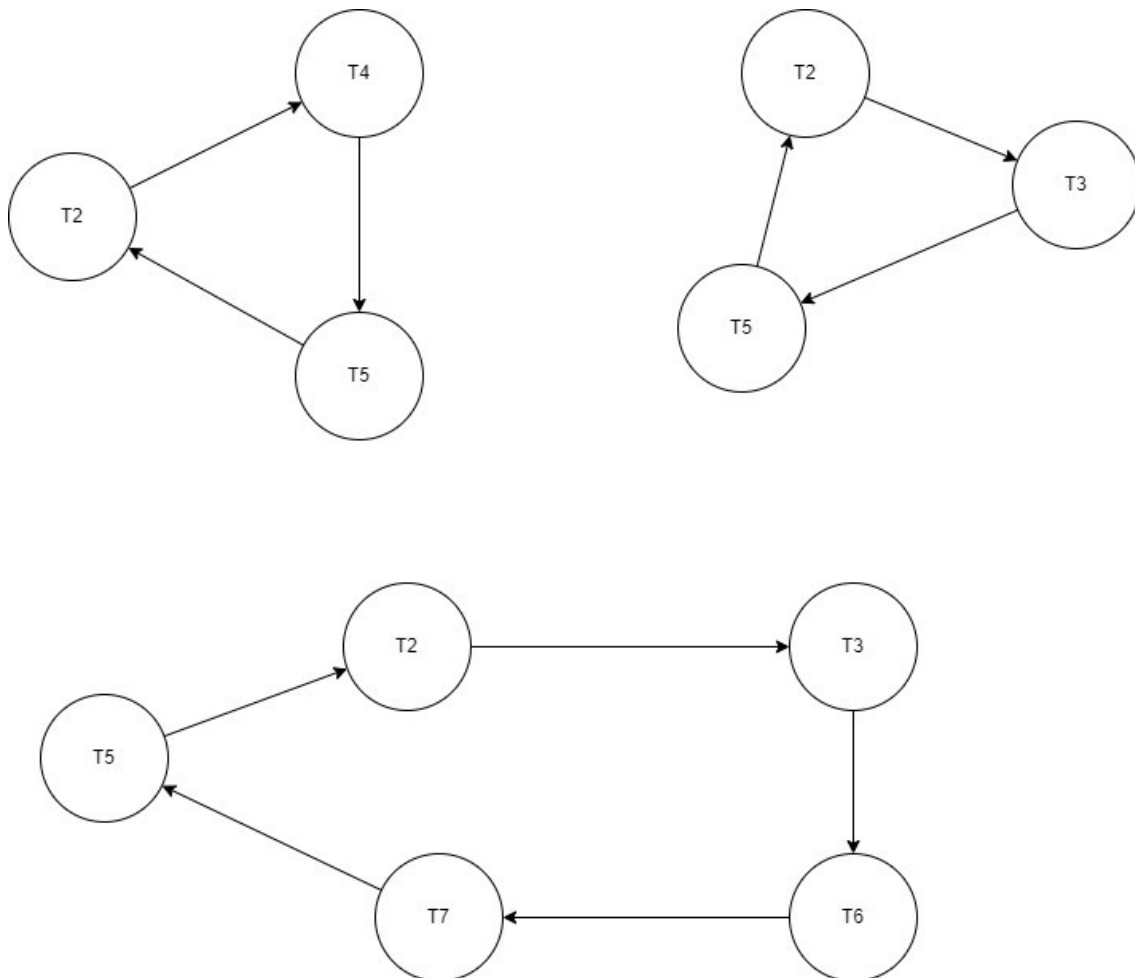


ii) Using the wait-for-graph from part (i) determine whether the transactions are in deadlock and explain how you detected deadlock. That is, you need to identify all the cycles in the WFG.

There are three deadlocks in the transaction scenario:

1. T2-T4-T5
2. T2-T3-T5
3. T2-T3-T6-T7-T5

The wait-for-graph makes a closed unidirectional transactional cycle in case of these three transaction paths.



iii) Explain how you would resolve the deadlock if it is detected in part(ii)

Termination of T5 will resolve all the deadlocks of WFG

3.i) Your first task is to create a spatial database using the coordinates for each area on both floors. Implement your design as a MySQL database using the spatial datatypes available with MySQL. Refer to Exercise 6 of practical 9 as required. Once you have created your database, export it as an SQL script using the export tab on phpMyAdmin. Copy and paste the contents of this script into your submission document for this assignment

-- phpMyAdmin SQL Dump

-- version 4.9.5

-- <https://www.phpmyadmin.net/>

--

-- Host: mysql.abdn.ac.uk

-- Generation Time: Dec 13, 2021 at 04:42 PM

-- Server version: 5.5.68-MariaDB

-- PHP Version: 5.6.40

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";

SET AUTOCOMMIT = 0;

START TRANSACTION;

SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;

/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;

/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;

/*!40101 SET NAMES utf8mb4 */;

--

-- Database: `t10rc21_offblk`

--

-- -----

--


```
-- Table structure for table `GroundFloor`
```

```
CREATE TABLE `GroundFloor` (  
  `FID` int(12) NOT NULL,  
  `ParkingSpace` varchar(24) DEFAULT NULL,  
  `Gcoordinate` polygon NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
-- Dumping data for table `GroundFloor`
```

```
INSERT INTO `GroundFloor` (`FID`, `ParkingSpace`, `Gcoordinate`) VALUES
```

```
(2,                                                                                                     'A2',  
0x000000000010300000001000000050000000000000000f03f00000000000000000000000000  
040000000000000000000000000000004000000000000004000000000000f03f000000000000  
40000000000000f03f00000000000000),
```

[illegible][illegible]

(6, 'B2',
0x00000000010300000001000000050000000000000000f03f0000000000000400000000000000
04000000000000000400000000000000040000000000000104000000000000f03f00000000000010
40000000000000f03f000000000000040),

(7, 'B3',
0x00000000010300000001000000050000000000000000040000000000000040000000000000
840000000000000004000000000000084000000000000104000000000000040000000000001
04000000000000004000000000000040),

(8, 'B4',
0x000000000103000000010000000500000000000000084000000000000040000000000001
04000000000000004000000000000104000000000000104000000000000840000000000001
0400000000000008400000000000040),

(9, 'C1',
0x0000000001030000000100000005000000000000000000000000000000104000000000000f
03f00000000000010400000000000f03f0000000000018400000000000000000000000000184
00000000000000000000000000001040),

(10, 'C2',
0x000000000103000000010000000500000000000000f03f000000000001040000000000000
0400000000000010400000000000040000000000018400000000000f03f0000000000018
4000000000000f03f000000000001040),

(11, 'C3',
0x0000000001030000000100000005000000000000000400000000000001040000000000000
84000000000000104000000000000840000000000018400000000000040000000000001
84000000000000040000000000001040),

(12, 'C4',
0x00000000010300000001000000050000000000000840000000000001040000000000001
040000000000001040000000000010400000000001840000000000084000000000001
8400000000000084000000000001040),

(13, 'D1',
0x000000000103000000010000000500000000000000000000000000000018400000000000f
03f000000000001840000000000f03f0000000000020400000000000000000000000000204
00000000000000000000000000001840),

(14, 'D2',
0x0000000001030000000100000005000000000000f03f000000000001840000000000000
040000000000001840000000000004000000000002040000000000f03f0000000000020
400000000000f03f000000000001840),

(15, 'D3',
0x00000000010300000001000000050000000000000400000000000018400000000000000
8400000000000184000000000000840000000000020400000000000040000000000002
0400000000000040000000000001840),

(16, 'D4',
0x0000000001030000000100000005000000000000084000000000001840000000000001

```
04000000000000001840000000000000104000000000000020400000000000008400000000000002
040000000000000008400000000000001840);
```

```
--
```

```
-- Indexes for dumped tables
```

```
--
```

```
--
```

```
-- Indexes for table `FirstFloor`
```

```
--
```

```
ALTER TABLE `FirstFloor`
  ADD PRIMARY KEY (`FID`);
```

```
--
```

```
-- Indexes for table `GroundFloor`
```

```
--
```

```
ALTER TABLE `GroundFloor`
  ADD PRIMARY KEY (`FID`);
```

```
COMMIT;
```

```
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
```

```
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
```

```
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

ii) You second task is to write an SQL query to list the parking spaces on the ground floor directly beneath the Office area on the first floor. Write this query in SQL script and include it in your submission document.

Showing rows 0 - 3 (4 total, Query took 0.0008 seconds.)

```
SELECT FID,ParkingSpace,AsText(Gcoordinate) AS PARKING FROM GroundFloor WHERE WITHIN(Gcoordinate,
ST_GeomFromText('Polygon((0 0,4 0,4 2,0 2,0 0))'))
```

FID	ParkingSpace	PARKING
1	A1	POLYGON((0 0,1 0,1 2,0 2,0 0))
2	A2	POLYGON((1 0,2 0,2 2,1 2,1 0))
3	A3	POLYGON((2 0,3 0,3 2,2 2,2 0))
4	A4	POLYGON((3 0,4 0,4 2,3 2,3 0))

iii)Your third task is to write an SQL query to list the parking slots which are visible from standing in the Hallway on the first floor and facing east. State any assumptions you make. Write this query in SQL script and include it in your submission document.

```
SELECT FID,ParkingSpace,AsText(Gcoordinate) AS PARKING FROM GroundFloor WHERE INTERSECTS(Gcoordinate,
ST_GeomFromText('Polygon((3 4,4 4,4 3,4 5,3 4))'))
```

FID	ParkingSpace	PARKING
7	B3	POLYGON((2 2,3 2,3 4,2 4,2 2))
8	B4	POLYGON((3 2,4 2,4 4,3 4,3 2))
11	C3	POLYGON((2 4,3 4,3 6,2 6,2 4))
12	C4	POLYGON((3 4,4 4,4 6,3 6,3 4))

Show query box

Showing rows 0 - 3 (4 total, Query took 0.0008 seconds.)

```
SELECT FID,ParkingSpace,AsText(Gcoordinate) AS PARKING FROM GroundFloor WHERE WITHIN(Gcoordinate,ST_GeomFromText('Polygon((0 0,4 0,4 2,0 2,0 0))'))
```

FID	ParkingSpace	PARKING
1	A1	POLYGON((0 0,1 0,1 2,0 2,0 0))
2	A2	POLYGON((1 0,2 0,2 2,1 2,1 0))
3	A3	POLYGON((2 0,3 0,3 2,2 2,2 0))
4	A4	POLYGON((3 0,4 0,4 2,3 2,3 0))

Show query box

Showing rows 0 - 3 (4 total, Query took 0.0006 seconds.)

```
SELECT FID,ParkingSpace,AsText(Gcoordinate) AS PARKING FROM GroundFloor WHERE INTERSECTS(Gcoordinate,ST_GeomFromText('Polygon((3 4,4 4,4 3,4 5,3 4))'))
```

FID	ParkingSpace	PARKING
7	B3	POLYGON((2 2,3 2,3 4,2 4,2 2))
8	B4	POLYGON((3 2,4 2,4 4,3 4,3 2))
11	C3	POLYGON((2 4,3 4,3 6,2 6,2 4))
12	C4	POLYGON((3 4,4 4,4 6,3 6,3 4))