

S1

1-b d

2-d

3-b c

S2

1-

```
create table promotion(  
  pId INT PRIMARY KEY,  
  pName VARCHAR(50),  
  startDate DATE,  
  endDate DATE  
)
```

```
create table book(  
  bId INT PRIMARY KEY,  
  pId INT REFERENCES promotion(pId),  
  title VARCHAR(50),  
  stock INT,  
  price FLOAT,  
  bType VARCHAR(50)  
)
```

```
create table customer(  
  cId INT PRIMARY KEY,  
  cName VARCHAR(50),  
  phone VARCHAR(11),  
  email VARCHAR(100)  
)
```

```
create table buyingFromBookShop(  
  cId INT REFERENCES customer(cId),  
  bId INT REFERENCES book(bId),  
  PRIMARY KEY(cId, bId),  
  statusBuying VARCHAR(50)  
)
```

```
create table buyingOnline(  
  cId INT REFERENCES customer(cId),  
  pId INT REFERENCES promotion(pId),  
  PRIMARY KEY(cId, pId),  
  statusBuying VARCHAR(50)  
)
```

2-

```
using System;
using System.Data;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace exempluPractic
{
    public partial class Form1 : Form
    {
        SqlConnection connection;
        DataSet dataSet;
        SqlCommandBuilder commandBuilder;
        BindingSource bsParent, bsChild;
        SqlDataAdapter daParent, daChild;
        String parentTable = "book";
        String childTable = "promotion";
        String communColumnName = "pId";

        private void button1_Click(object sender, EventArgs e)
        {
            commandBuilder = new SqlCommandBuilder(daChild);
            commandBuilder.GetInsertCommand();

            daParent.Update(dataSet, parentTable);
            daChild.Update(dataSet, childTable);
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            connection = new SqlConnection(@"Data Source=DESKTOP-
PU9UT93\SQLEXPRESS;Initial Catalog=practicDBMS;Integrated Security=SSPI");
            dataSet = new DataSet();
        }
    }
}
```

```

        daChild = new SqlDataAdapter("SELECT * FROM " + childTable, connection);
        daParent = new SqlDataAdapter("SELECT * FROM " + parentTable, connection);

        commandBuilder = new SqlCommandBuilder(daParent);
        daChild.Fill(dataSet, childTable);
        daParent.Fill(dataSet, parentTable);

        DataRelation dataRelation = new DataRelation("FK_Parent_Child",
            dataSet.Tables[childTable].Columns[communColumnName],
            dataSet.Tables[parentTable].Columns[communColumnName]);
        dataSet.Relations.Add(dataRelation);

        bsChild = new BindingSource();
        bsChild.DataSource = dataSet;
        bsChild.DataMember = childTable;

        bsParent = new BindingSource();
        bsParent.DataSource = bsChild;
        bsParent.DataMember = "FK_Parent_Child";

        dgvPromotion.DataSource = bsChild;
        dgvBook.DataSource = bsParent;
    }

    public Form1()
    {
        InitializeComponent();
    }
}

```

3-

-----Non-repetable read 1------(T1)

```
SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED
```

```
BEGIN TRAN
```

```
    select * from promotion  
    WAITFOR DELAY '00:00:10'  
    select * from promotion
```

```
COMMIT TRAN
```

```
update promotion set pName = 'nonRepeatblereads' where pId = 1
```

-----Non-repetable read 2------(T2)

```
SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED
```

```
BEGIN TRAN
```

```
    update promotion set pName = 'newNonRepetableReads' where pId = 1
```

```
COMMIT TRAN
```

The non-repetable reads occurred because:

- The first process reading the value might get two different values, as the changed data (is done in T2) is read a second time because the second process changes the data

- To prevent this concurrency issue, we must increment the isolation level.