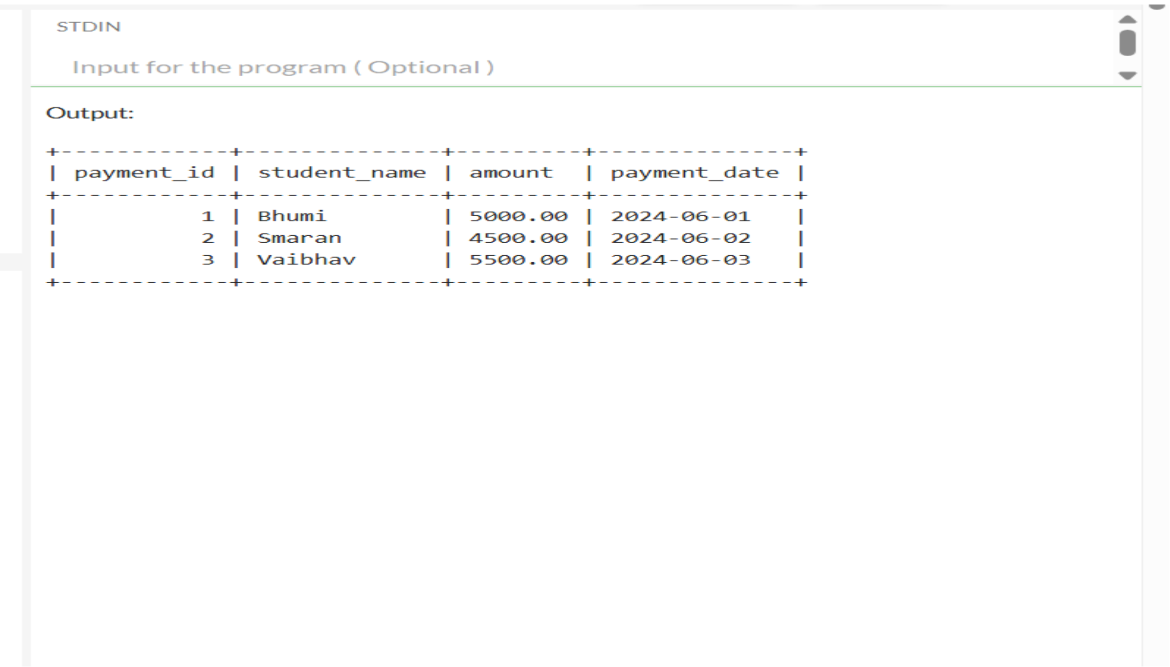
Name:	Bhumi
UID:	23BCS13453
Subject:	ADBMS
Section:	622-B

Part A: Insert Multiple Fee Payments in a Transaction

Ans:

Code:

```
-- Make sure table exists
CREATE TABLE IF NOT EXISTS FeePayments (
    payment_id INT PRIMARY KEY,
    student_name VARCHAR(100) NOT NULL,
    amount DECIMAL(10,2) CHECK (amount > 0),
    payment_date DATE NOT NULL
) ENGINE=InnoDB;

-- Start transaction
START TRANSACTION;

INSERT INTO FeePayments (payment_id, student_name, amount, payment_date)
VALUES
  (1, 'Bhumi', 5000.00, '2024-06-01'),
  (2, 'Smaran', 4500.00, '2024-06-02'),
  (3, 'Vaibhav', 5500.00, '2024-06-03');

COMMIT;

SELECT * FROM FeePayments;
```



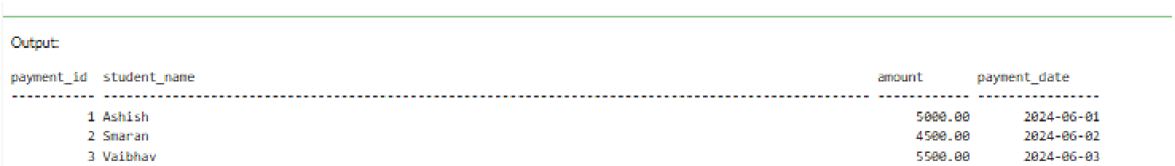
Output:

```
+------------+--------------+---------+--------------+
| payment_id | student_name | amount  | payment_date |
+------------+--------------+---------+--------------+
|          1 | Bhumi        | 5000.00 | 2024-06-01   |
|          2 | Smaran       | 4500.00 | 2024-06-02   |
|          3 | Vaibhav      | 5500.00 | 2024-06-03   |
+------------+--------------+---------+--------------+
```

Part B: Demonstrate ROLLBACK for Failed Payment Insertion

```
IF OBJECT_ID('FeePayments', 'U') IS NOT NULL
    DROP TABLE FeePayments;
GO

CREATE TABLE FeePayments (
    payment_id INT PRIMARY KEY,
    student_name VARCHAR(100) NOT NULL,
    amount DECIMAL(10,2) CHECK (amount > 0),
    payment_date DATE NOT NULL
);
GO
BEGIN TRANSACTION;

INSERT INTO FeePayments (payment_id, student_name, amount, payment_date)
VALUES
(1, 'Ashish', 5000.00, '2024-06-01'),
(2, 'Smaran', 4500.00, '2024-06-02'),
(3, 'Vaibhav', 5500.00, '2024-06-03');
```

```
COMMIT;
GO
SELECT * FROM FeePayments;
GO
BEGIN TRANSACTION;
INSERT INTO FeePayments (payment_id, student_name, amount, payment_date)
VALUES (4, 'Kiran', 4800.00, '2024-06-04');

-- Invalid insert (duplicate payment_id and negative amount)
INSERT INTO FeePayments (payment_id, student_name, amount, payment_date)
VALUES (1, 'Ashish', -500.00, '2024-06-05');
ROLLBACK;
GO
SELECT * FROM FeePayments;
```

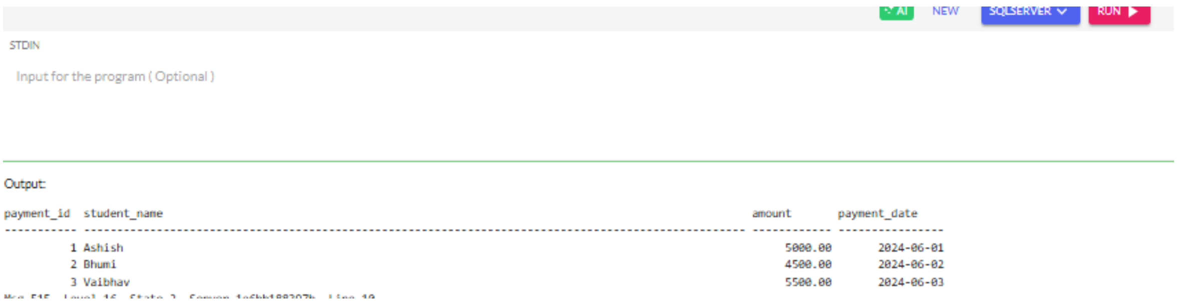


```
Output:

payment_id  student_name                                                                                         amount      payment_date
----------- ---------------------------------------------------------------------------------------------------- ----------- ----------------
          1 Ashish                                                                                                   5000.00       2024-06-01
          2 Smaran                                                                                                   4500.00       2024-06-02
          3 Vaibhav                                                                                                  5500.00       2024-06-03
```

Part C: Simulate Partial Failure and Ensure Consistent State

```
IF OBJECT_ID('FeePayments', 'U') IS NOT NULL

   DROP TABLE FeePayments;

GO


CREATE TABLE FeePayments (

   payment_id INT PRIMARY KEY,

   student_name VARCHAR(100) NOT NULL,

   amount DECIMAL(10,2) CHECK (amount > 0),

   payment_date DATE NOT NULL

);

GO

BEGIN TRANSACTION;


INSERT INTO FeePayments (payment_id, student_name, amount, payment_date)
```

```
VALUES
(1, 'Ashish', 5000.00, '2024-06-01'),
(2, 'Smaran', 4500.00, '2024-06-02'),
(3, 'Vaibhav', 5500.00, '2024-06-03');

COMMIT;
GO

-- Verify initial records
SELECT * FROM FeePayments;
GO
BEGIN TRANSACTION;

-- Valid insert
INSERT INTO FeePayments (payment_id, student_name, amount, payment_date)
VALUES (4, 'Kiran', 4800.00, '2024-06-04');

-- Invalid insert (student_name is NULL)
INSERT INTO FeePayments (payment_id, student_name, amount, payment_date)
VALUES (5, NULL, 4600.00, '2024-06-05');

-- Rollback entire transaction due to failure
ROLLBACK;
GO
SELECT * FROM FeePayments;
GO
```

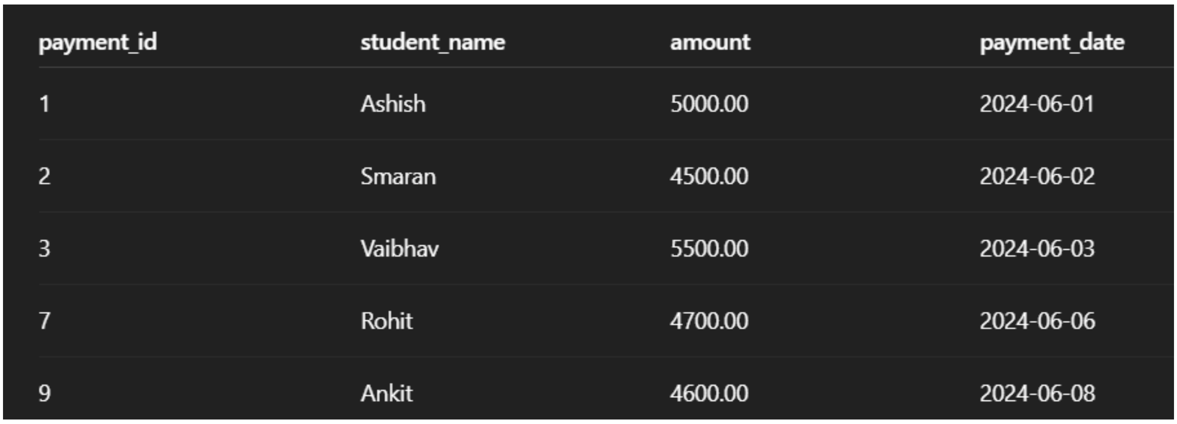


```
payment_id  student_name                                                                                          amount       payment_date
----------- ---------------------------------------------------------------------------------------------------- ------------ ----------------
          1 Ashish                                                                                                    5000.00       2024-06-01
          2 Bhumi                                                                                                     4500.00       2024-06-02
          3 Vaibhav                                                                                                   5500.00       2024-06-03
Msg 515, Level 16, State 2, Server 1e6bb188207b, Line 10
```

Part D: Verify ACID Compliance with Transaction Flow

```
CREATE TABLE FeePayments (
    payment_id INT PRIMARY KEY,
    student_name VARCHAR(100) NOT NULL,
    amount DECIMAL(10,2) CHECK (amount > 0),
    payment_date DATE NOT NULL
);
GO
BEGIN TRANSACTION;
INSERT INTO FeePayments (payment_id, student_name, amount, payment_date)
VALUES
(1, 'Ashish', 5000.00, '2024-06-01'),
(2, 'Smaran', 4500.00, '2024-06-02'),
(3, 'Vaibhav', 5500.00, '2024-06-03');
COMMIT;
GO
BEGIN TRANSACTION;
INSERT INTO FeePayments (payment_id, student_name, amount, payment_date)
VALUES (4, 'Kiran', 4800.00, '2024-06-04');  -- valid
INSERT INTO FeePayments (payment_id, student_name, amount, payment_date)
VALUES (1, 'Ashish', -500.00, '2024-06-05');  -- invalid: duplicate + negative
ROLLBACK;
GO
BEGIN TRANSACTION;
INSERT INTO FeePayments (payment_id, student_name, amount, payment_date)
VALUES (5, 'Neha', 5000.00, '2024-06-05');  -- valid
INSERT INTO FeePayments (payment_id, student_name, amount, payment_date)
VALUES (6, NULL, 4600.00, '2024-06-06');    -- invalid: student_name NULL
ROLLBACK;
GO
BEGIN TRANSACTION;
INSERT INTO FeePayments (payment_id, student_name, amount, payment_date)
VALUES (7, 'Rohit', 4700.00, '2024-06-06');
COMMIT;
GO
```

```
BEGIN TRANSACTION;
INSERT INTO FeePayments (payment_id, student_name, amount, payment_date)
VALUES (8, 'Simran', 4800.00, '2024-06-07');  -- valid
INSERT INTO FeePayments (payment_id, student_name, amount, payment_date)
VALUES (1, 'Ashish', 5000.00, '2024-06-01');  -- invalid
ROLLBACK;
GO
BEGIN TRANSACTION;
INSERT INTO FeePayments (payment_id, student_name, amount, payment_date)
VALUES (9, 'Ankit', 4600.00, '2024-06-08');
COMMIT;
GO
SELECT 'Final ACID-Compliant Table' AS Stage;
SELECT * FROM FeePayments ORDER BY payment_id;
GO
```



payment_id	student_name	amount	payment_date
1	Ashish	5000.00	2024-06-01
2	Smaran	4500.00	2024-06-02
3	Vaibhav	5500.00	2024-06-03
7	Rohit	4700.00	2024-06-06
9	Ankit	4600.00	2024-06-08