

Assignment No. 4

Assignment Title: Unnamed PL/SQL code block: Use of Control structure and Exception handling is mandatory.

PL/SQL Code:

```
DECLARE
    v_roll_no NUMBER;
    v_book_name VARCHAR2(100);
    v_date_of_issue DATE;
    v_fine_amt NUMBER;
    v_days_late NUMBER;

    -- Define custom exception
    fine_exception EXCEPTION;

    -- Exception handlers
    PRAGMA EXCEPTION_INIT(fine_exception, -20001);

BEGIN
    -- Accept Roll_no and Name of Book from the user
    v_roll_no := &roll_no;
    v_book_name := '&book_name';

    -- Get the date of issue and status
    SELECT Date_of_Issue, Status
    INTO v_date_of_issue, v_fine_amt
    FROM Borrower
    WHERE Roll_no = v_roll_no AND Name_of_Book = v_book_name;

    -- Calculate the number of days late
    v_days_late := TRUNC(SYSDATE - v_date_of_issue);

    IF v_days_late >= 15 AND v_days_late <= 30 THEN
        v_fine_amt := v_days_late * 5; -- Rs. 5 per day
    ELSIF v_days_late > 30 THEN
        v_fine_amt := v_days_late * 50; -- Rs. 50 per day
    END IF;

    -- Update the Status in the Borrower table
    UPDATE Borrower
    SET Status = 'R'
    WHERE Roll_no = v_roll_no AND Name_of_Book = v_book_name;
```

```

-- Insert fine details into the Fine table
INSERT INTO Fine(Roll_no, Date, Amt)
VALUES (v_roll_no, SYSDATE, v_fine_amt);

-- Commit the transaction
COMMIT;

DBMS_OUTPUT.PUT_LINE('Fine calculated and recorded successfully.');
```

EXCEPTION

```

    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.PUT_LINE('Borrower not found.');
```

WHEN fine_exception THEN

```

    DBMS_OUTPUT.PUT_LINE('Fine calculation error.');
```

WHEN OTHERS THEN

```

    DBMS_OUTPUT.PUT_LINE('An error occurred: ' || SQLERRM);
```

END;

/

Output:

Enter value for roll_no: 12345

Enter value for book_name: 'Sample Book'

Fine calculated and recorded successfully.