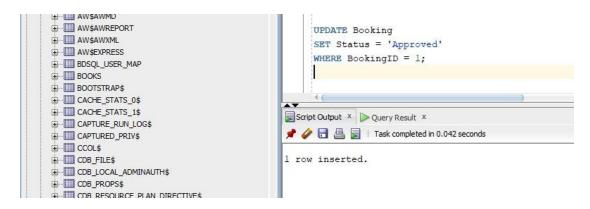
### Phase VI -: Database Interaction and Transactions

# **DML**

# **INSERT**

```
AW$AWREPORT
                                               INSERT INTO Appuser (Name, Email, Password, Role, DepartmentID)
AW$AWXML
AW$EXPRESS
                                               VALUES ('Alex Turner', 'alex.turner@university.edu', 'pwdl23', 'Student', 1);
BDSQL_USER_MAP
■ BOOKS
BOOTSTRAP$
CACHE_STATS_0$
CAPTURE RUN LOGS
                                          Script Output × Query Result ×
CAPTURED_PRIV$
                                          📌 🧽 🔡 🚇 🔋 | Task completed in 0.036 seconds
III CCOLS
CDB_FILE$
                                          ORA-00001: unique constraint (SYS.SYS_C008080) violated
CDB_LOCAL_ADMINAUTH$
CDB_PROPS$
CDB RESOURCE PLAN DIRECTIVE$
CDB_RESOURCE_PLAN$
                                          l row inserted.
CDB SERVICES
CDB_TS$
```

#### **UPDATE**



### DDL

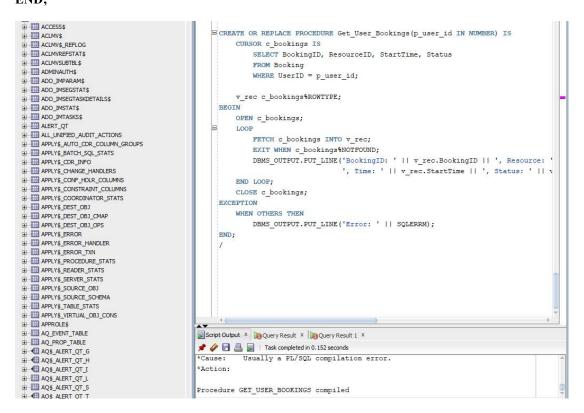
```
# EE APPLYS_ERROR
APPLYS ERROR HANDLER
                                               CREATE TABLE AuditLog (
APPLY$_ERROR_TXN
                                                   LogID NUMBER GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,
APPLY$_PROCEDURE_STATS
                                                    Operation VARCHAR2 (50),
APPLY$_READER_STATS
                                                    PerformedBy VARCHAR2 (100),
APPLYS SERVER STATS
APPLY$_SOURCE_OBJ
                                                    ActionDate TIMESTAMP DEFAULT CURRENT_TIMESTAMP
APPLY'S SOURCE SCHEMA
APPLY$_TABLE_STATS
APPLYS VIRTUAL OBJ CONS
APPROLE$
```

## **PROCEDURE**

```
CREATE OR REPLACE PROCEDURE Get_User_Bookings(p_user_id IN NUMBER) IS
CURSOR c_bookings IS
SELECT BookingID, ResourceID, StartTime, Status
FROM Booking
WHERE UserID = p_user_id;

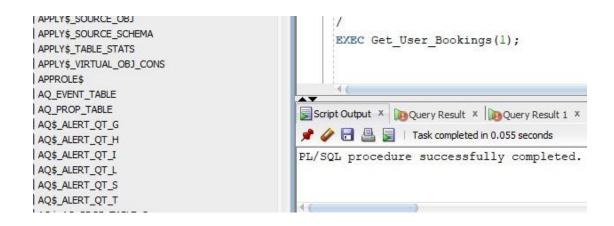
v_rec c_bookings%ROWTYPE;
BEGIN
OPEN c_bookings;
LOOP
FETCH c_bookings INTO v_rec;
EXIT WHEN c_bookings%NOTFOUND;
```

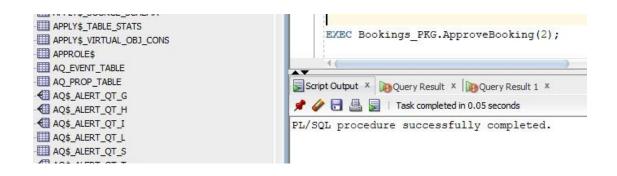
```
DBMS OUTPUT.PUT LINE('BookingID: ' || v rec.BookingID || ', Resource: ' ||
v rec.ResourceID ||
              ', Time: ' || v_rec.StartTime || ', Status: ' || v_rec.Status);
 END LOOP:
 CLOSE c bookings;
EXCEPTION
  WHEN OTHERS THEN
   DBMS OUTPUT.PUT LINE('Error: ' || SQLERRM);
END;
CREATE OR REPLACE FUNCTION Get Resource Bookings Count(p resource id IN
NUMBER)
RETURN NUMBER IS
 v count NUMBER;
BEGIN
 SELECT COUNT(*) INTO v count FROM Booking WHERE ResourceID = p resource id;
 RETURN v_count;
EXCEPTION
  WHEN OTHERS THEN
   RETURN-1;
END;
```



# **TESTING**

```
EXEC Get_User_Bookings(1);
-- Function test
SELECT Get_Resource_Bookings_Count(1) FROM dual;
-- Package test
EXEC Bookings_PKG.ApproveBooking(2);
SELECT Bookings PKG.GetTotalBookings(1) FROM dual;
```





#### **FUNCTION EXAMPLE**

```
# ## ALL_UNIFIED_AUDIT_ACTIONS
                                                           CREATE OR REPLACE FUNCTION Get Resource Bookings Count (p_resource_id IN NUMBER)

    APPLY$_AUTO_CDR_COLUMN_GROUPS
    APPLY$_BATCH_SQL_STATS
                                                            RETURN NUMBER IS
APPLY$_CDR_INFO

APPLY$_CHANGE_HANDLERS
                                                                 v_count NUMBER;
                                                            BEGIN
■ ■ APPLY'S CONF HDLR COLUMNS
                                                                 SELECT COUNT(*) INTO v_count

    APPLY$_CONSTRAINT_COLUMNS
    APPLY$_COORDINATOR_STATS
                                                                 FROM Booking
                                                                 WHERE ResourceID = p_resource_id;
APPLY$_DEST_OBJ CMAP
                                                                 RETURN v_count;
APPLYS DEST OBJ OPS
APPLYS_ERROR
APPLYS_ERROR_HANDLER
                                                            EXCEPTION
                                                                 WHEN OTHERS THEN
APPLY$ ERROR_TXN

APPLY$ PROCEDURE STATS
                                                                      RETURN -1:
APPLY$_READER_STATS
                                                            END;
APPLY$_SERVER_STATS

APPLY$_SOURCE_OBJ
APPLY$_SOURCE_SCHEMA
APPLY$_TABLE_STATS
                                                            # APPLY$_VIRTUAL_OBJ_CONS
```