

PHASE 7

Create Audit Table

The screenshot shows the SQL Developer interface with the 'Query Builder' worksheet. The 'Tables (Filtered)' pane on the left shows the database schema. The main window displays the following SQL code:

```
CREATE TABLE Vote_Audit (  
  Audit_ID NUMBER GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,  
  Voter_ID INT,  
  Candidate_ID INT,  
  Election_ID INT,  
  Vote_Time TIMESTAMP,  
  Action_By VARCHAR2(50)  
);
```

The 'Script Output' pane at the bottom shows the execution results:

```
Task completed in 1.66 seconds  
  
Procedure CAST_VOTE compiled  
  
Function COUNT_VOTES compiled  
  
Vote successfully cast.  
  
PL/SQL procedure successfully completed.  
  
>>Query Run In:Query Result  
  
Table VOTE_AUDIT created.
```

An 'Activate Windows' watermark is visible in the bottom right corner.

Creation of Trigger to Log Vote to Audit Table

The screenshot shows the SQL Developer interface with the 'Query Builder' worksheet. The 'Tables (Filtered)' pane on the left shows the database schema. The main window displays the following SQL code:

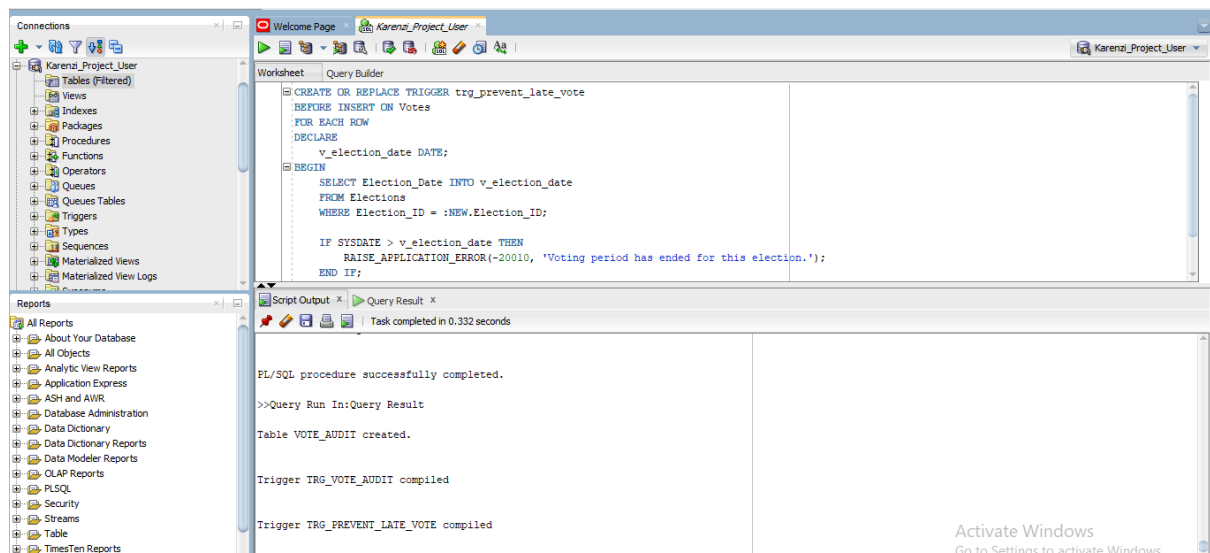
```
CREATE OR REPLACE TRIGGER trg_vote_audit  
AFTER INSERT ON Votes  
FOR EACH ROW  
BEGIN  
  INSERT INTO Vote_Audit (Voter_ID, Candidate_ID, Election_ID, Vote_Time, Action_By)  
  VALUES (:NEW.Voter_ID, :NEW.Candidate_ID, :NEW.Election_ID, :NEW.Timestamp, USER);  
END;
```

The 'Script Output' pane at the bottom shows the execution results:

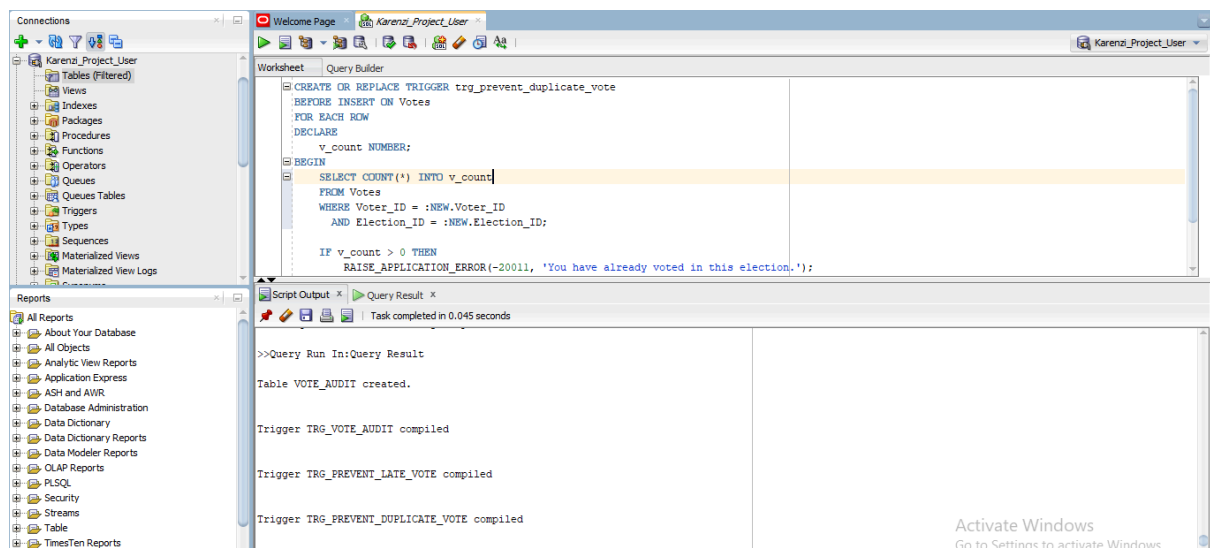
```
Task completed in 1.052 seconds  
  
Function COUNT_VOTES compiled  
  
Vote successfully cast.  
  
PL/SQL procedure successfully completed.  
  
>>Query Run In:Query Result  
  
Table VOTE_AUDIT created.  
  
Trigger TRG_VOTE_AUDIT compiled
```

An 'Activate Windows' watermark is visible in the bottom right corner.

Creation of Trigger to Prevent Voting After Deadline



Creation of Trigger to Prevent Double Voting (Extra Safety)



Testing Triggers

Connections

Karenzi_Project_User

Tables (Filtered)

Views

Indexes

Packages

Procedures

Functions

Operators

Queues

Queues Tables

Triggers

Types

Sequences

Materialized Views

Materialized View Logs

Custome...

Reports

All Reports

About Your Database

All Objects

Analytic View Reports

Application Express

ASH and AWR

Database Administration

Data Dictionary

Data Dictionary Reports

Data Modeler Reports

OLAP Reports

PL/SQL

Security

Streams

Table

Welcome Page

Karenzi_Project_User

Worksheet

Query Builder

INSERT INTO Votes (Vote_ID, Voter_ID, Candidate_ID, Election_ID, Timestamp)
VALUES (Votes_seq.NEXTVAL, 11, 101, 12, SYSTIMESTAMP);

SELECT * FROM Vote_Audit ORDER BY Audit_ID;

Script Output

Query Result

Query Result 1

Query Result 2

All Rows Fetched: 1 in 0.009 seconds

	AUDT_ID	VOTER_ID	CANDIDATE_ID	ELECTION_ID	VOTE_TIME	ACTION_BY
1	1	11	101	12-22-MAY-25 11.51.43.027000000	PM	KARENZI_VOTING

Activate Windows