

## PHASE 6

### Creation of a Sequence for Auto-Generating Vote IDs

The screenshot displays the SQL Developer interface with the following components:

- Connections:** A tree view on the left showing the 'Karenzi\_Project\_User' connection.
- Worksheet:** The main area containing a SQL script in the 'Query Builder' tab:

```
CREATE SEQUENCE VOTES_seq  
START WITH 1003  
INCREMENT BY 1  
NOCACHE;
```
- Script Output:** A pane at the bottom showing the execution results:

```
1 row inserted.  
  
1 row inserted.  
  
Commit complete.  
  
Sequence VOTES_SEQ created.
```

The status bar at the bottom indicates 'Task completed in 0.553 seconds'.

## Creation of a Procedure to Cast a Vote and calling it

The image displays two screenshots of the SQL Developer interface, demonstrating the creation and execution of a PL/SQL procedure.

**Top Screenshot: Creating the Procedure**

The **Query Builder** window shows the following SQL code:

```
CREATE OR REPLACE PROCEDURE cast_vote (  
    p_voter_id    IN INT,  
    p_candidate_id IN INT,  
    p_election_id  IN INT  
) IS  
    v_eligible CHAR(1);  
    v_already_voted NUMBER := 0;  
BEGIN  
  
    SELECT Eligibility_Status INTO v_eligible  
    FROM Voters  
    WHERE Voter_ID = p_voter_id;
```

The **Script Output** window shows the execution results:

```
1 row inserted.  
  
1 row inserted.  
  
Commit complete.  
  
Sequence VOTES_SEQ created.  
  
Procedure CAST_VOTE compiled
```

**Bottom Screenshot: Calling the Procedure**

The **Query Builder** window shows the following SQL code:

```
SET SERVEROUTPUT ON;  
  
EXEC cast_vote(11, 102, 11);
```

The **Script Output** window shows the execution results:

```
Sequence VOTES_SEQ created.  
  
Procedure CAST_VOTE compiled  
  
Function COUNT_VOTES compiled  
  
Vote successfully cast.  
  
PL/SQL procedure successfully completed.
```

## Creation of a Function to Count Votes for a Candidate and calling it

The screenshot displays the Oracle SQL Developer interface with the 'Karezni\_Project\_User' schema selected. The 'Connections' pane on the left shows the database structure. The main workspace is divided into three panes: 'Worksheet', 'Query Builder', and 'Script Output'.

**Worksheet:** Contains the SQL code for creating a function named `count_votes`.

```
CREATE OR REPLACE FUNCTION count_votes (  
    p_candidate_id IN INT,  
    p_election_id IN INT  
) RETURN INT IS  
    v_count INT;  
BEGIN  
    SELECT COUNT(*) INTO v_count  
    FROM Votes  
    WHERE Candidate_ID = p_candidate_id AND Election_ID = p_election_id;  
    RETURN v_count;  
END;
```

**Script Output:** Shows the execution results of the SQL script.

```
1 row inserted.  
  
Commit complete.  
  
Sequence VOTES_SEQ created.  
  
Procedure CAST_VOTE compiled  
  
Function COUNT_VOTES compiled
```

**Worksheet:** Contains the SQL code for calling the `count_votes` function.

```
SET SERVEROUTPUT ON;  
  
SELECT count_votes(102, 11) AS total_votes FROM dual;
```

**Query Result:** Shows the result of the query.

TOTAL_VOTES
2

END.