PL/SQL static code analysis: Predefined exceptions should not be overridden

1-2 minutes

Naming custom exceptions the same as predefined ones, while technically acceptable, is not a good practice.

Noncompliant Code Example

```
SET SERVEROUTPUT ON

DECLARE
  no_data_found EXCEPTION; -- Noncompliant, overrides an Oracle predefined exception

  d VARCHAR2(1);
BEGIN
  SELECT dummy INTO d FROM DUAL WHERE dummy = 'Y';
-- Will raise STANDARD.NO_DATA_FOUND
EXCEPTION
  WHEN NO_DATA_FOUND THEN
    DBMS_OUTPUT.PUT_LINE('No data found!'); -- Won't be executed, as NO_DATA_FOUND was overriden, confusing!
  WHEN OTHERS THEN
    DBMS_OUTPUT.PUT_LINE('Unknown error!'); -- *Will* be executed
```

```
END;
/
```

Compliant Solution

```
SET SERVEROUTPUT ON

DECLARE
  my_own_exception EXCEPTION; -- Compliant

  d VARCHAR2(1);
BEGIN
  SELECT dummy INTO d FROM DUAL WHERE dummy = 'Y';
EXCEPTION
  WHEN NO_DATA_FOUND THEN
    DBMS_OUTPUT.PUT_LINE('No data found!'); -- *Will* be
executed
  WHEN OTHERS THEN
    DBMS_OUTPUT.PUT_LINE('Unknown error!');
END;
/
```