

Query1	Query2
<pre> FUNCTION USENESTEDSETS (name VARCHAR, tree NUMBER) -- input taxon name -- input source tree RETURN VARCHAR IS -- output taxon names outName VARCHAR(4000) := NULL; BEGIN -- hierarchical query using nestedSet left_Id and right_Id SELECT get_name_text (node.name_Id) INTO USENESTEDSETS.outName FROM node WHERE tree_Id = tree AND node.left_Id BETWEEN (SELECT left_Id FROM node WHERE name_Id = get_Id(USENESTEDSETS.inName) AND tree_Id = USENESTEDSETS.inTree) AND (SELECT right_Id FROM node WHERE name_Id = get_Id(USENESTEDSETS.inName) AND tree_Id = USENESTEDSETS.inTree); RETURN outName; END; </pre>	<pre> FUNCTION USEPATHS (name VARCHAR, tree NUMBER) -- input taxon name -- input source tree RETURN VARCHAR IS -- output taxon names outName VARCHAR(4000) := NULL; -- output path string outPath VARCHAR(4000) := NULL; BEGIN -- SQL statement for path string SELECT path INTO USEPATHS.outPath FROM node WHERE name_Id = get_Id(name); -- hierarchical query using path string SELECT get_name_text(node.name_Id) INTO USEPATHS.outName FROM node WHERE path LIKE USEPATHS.outPath% AND tree_Id = USEPATHS.input tree; RETURN outName; END; </pre>
Query3	Query4
<pre> FUNCTION USECONNECTBY (-- input taxon name inName VARCHAR, -- input source tree inTree NUMBER) -- outputs taxon names RETURN VARCHAR IS outName VARCHAR(4000) := NULL; BEGIN -- hierarchical query using connect by SELECT get_name_text (node.name_Id) INTO USECONNECTBY.outName FROM node WHERE tree_Id = inTree START WITH (get_Id(USECONNECTBY.inName) AND tree_Id = USECONNECTBY.inTree) CONNECT BY PRIOR parent_name_Id = name_Id; RETURN outName; END; </pre>	<pre> FUNCTION VERNACULAR2VALID (-- input vernacular name inName VARCHAR) -- outputs taxon names RETURN VARCHAR IS outName VARCHAR(4000) := NULL; BEGIN -- vernacular query SELECT get_name_text(valid_name_Id) INTO VERNACULAR2VALID.outName FROM vernacular WHERE name_Id IN (SELECT name_Id FROM name WHERE name_text = VERNACULAR2VALID.inName); RETURN outName; END; </pre>