Evaluation

1) Write a query to find the root node.

```
SELECT id, product_name FROM product
WHERE parent_id IS NULL;
```

2) Write a query to find leaf node.

```
SELECT
   p.id, p.`product_name`
FROM
   product p
      LEFT JOIN
   product p2 ON p2.parent_id = p.id
WHERE
   p2.id IS NULL
```

3) Write a query to find non-leaf node.

```
SELECT p.id, p.`product_name`
FROM product p
LEFT JOIN product p2 ON p2.parent_id = p.id
 WHERE p2.id IS NOT NULL
```

4) Write a query to find the path of each node

```
WITH RECURSIVE item_path (id, `product_name`, path) AS
(
  SELECT id, `product_name`, `product_name` as path
    FROM product
    WHERE parent_id IS NULL
  UNION ALL
  SELECT p.id, p.`product_name`, CONCAT(product_path.path, ' > ',
p.`product_name`)
    FROM item_path AS product_path JOIN product AS p
      ON product_path.id = p.parent_id
)
SELECT * FROM item_path
ORDER BY path
```

5) Write a function to calculate node level. e.g. Electronics is at 0 level, Camera is on level 2 and iOs is on level 3.

```
WITH RECURSIVE item_path (id, `product_name`, node_level) AS
(
  SELECT id, `product_name`, 0 node_level
    FROM product
    WHERE parent_id IS NULL
  UNION ALL
  SELECT p.id, p.`product_name`,path.node_level + 1
    FROM item_path AS path JOIN product AS p
      ON path.id = p.parent_id
)
SELECT * FROM item_path
ORDER BY node_level
```

6) Write a procedure to get the immediate children.

```
DELIMITER $$

CREATE PROCEDURE get_children()

BEGIN

    SELECT id, product_name

FROM product

WHERE  parent_id = pid

END$$

DELIMITER ;
```

CREATE DEFINER=`root`@`localhost` PROCEDURE `get_children`(IN `pid` INT(10)) NOT DETERMINISTIC CONTAINS SQL SQL SECURITY DEFINER SELECT

```
    id, product_name

FROM

    product

WHERE

    parent_id = pid
```

```
SET @p0='2';

CALL `get_children`(@p0);
```