Soumyajit Kundu 30/12/2020

Technical Task 2

1) Write a query to find the root node.

```
SELECT
   id, title
FROM
   category
WHERE
   parent_id IS NULL;
```

2) Write a query to find leaf node.

```
SELECT
   c1.id, c1.title
FROM
   category c1
      LEFT JOIN
   category c2 ON c2.parent_id = c1.id
WHERE
   c2.id IS NULL;
```

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

```
SELECT
  DISTINCT( c1.id), c1.title
FROM
   category c1
      INNER JOIN
   category c2 ON c2.parent_id = c1.id
WHERE
   c2.id IS NOT NULL
```

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

```
WITH RECURSIVE category_path (id, title, path) AS
(
  SELECT id, title, title as path
    FROM category
    WHERE parent_id IS NULL
  UNION ALL
  SELECT c.id, c.title, CONCAT(cp.path, ' > ', c.title)
    FROM category_path AS cp JOIN category AS c
      ON cp.id = c.parent_id
)
SELECT * FROM category_path
ORDER BY path;
```

5) Write a function to calculate node level. (Not done using function)

```
WITH RECURSIVE category_path (id, title, lvl) AS
(
  SELECT id, title, 0 lvl
    FROM category
    WHERE parent_id IS NULL
  UNION ALL
  SELECT c.id, c.title,cp.lvl + 1
    FROM category_path AS cp JOIN category AS c
      ON cp.id = c.parent_id
)
SELECT * FROM category_path
ORDER BY lvl;
```

6) Write a procedure to get the immediate children.

```
DELIMITER $$

CREATE PROCEDURE immediate_children

(IN con VARCHAR(20))

BEGIN

  SELECT id, title FROM category WHERE parent_id = 1;

END $$

DELIMITER ;
```