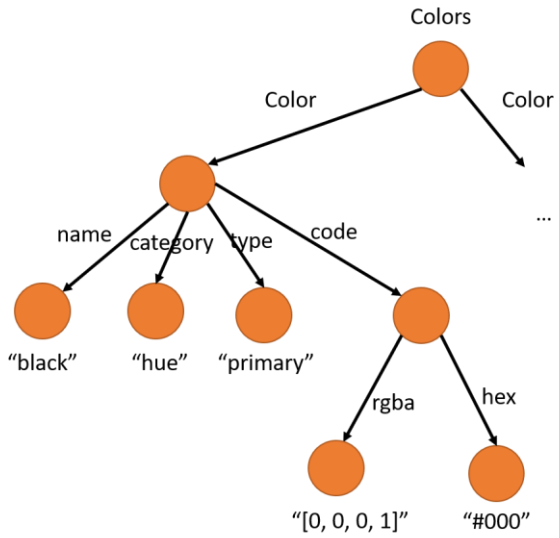
COMP0022: Database and information systems (PG – Term 1)

Non-relational databases exercises

1. Convert the following JSON object to an OEM-like tree structure.


```
{
  "colors": [
    {
      "color": "black",
      "category": "hue",
      "type": "primary",
      "code": {
        "rgba": [0,0,0,1],
        "hex": "#000"
      }
    },
    {
      "color": "white",
      "category": "value",
      "code": {
        "rgba": [255,255,255, 1],
        "hex": "#FFF"
      }
    },
    {
      "color": "red",
      "category": "hue",
      "type": "primary",
      "code": {
        "rgba": [255,0,0,1],
        "hex": "#F00"
      }
    },
    {
      "color": "green",
      "category": "hue",
      "type": "secondary",
      "code": {
        "rgba": [0,255,0,1],
        "hex": "#0F0"
      }
    },
  ]
}
```

Partial picture:



I altered the structure slightly to add a "name" field for the color. There are multiple ways to handle things like the rgba code.

2. Convert the JSON file to (a snippet of) equivalent XML.

```
<Colors>
 <Color name="black">
  <Category>Hue</Category>
  <Type>Primary</Type>
  <Code>
   <RGBA>
    <Redvalue>0</Redvalue>
    <Greenvalue>0</Greenvalue>
    <Bluevalue>0</Bluevalue>
    <Alphavalue>1</Alphavalue>
   </RGBA>
   <Hex>#000</Hex>
  </Code>
 </Color>
 <Color>
  ...
</Colors>
```

(There are many ways to do this, especially given that first-order children can also be implemented as attributes.)

3. Given the XML schema you created in the previous question, create an XML query that:

    a. Selects colors that belong to the category "hue".

doc("colors.xml")/Color[Category = "hue"]

OR

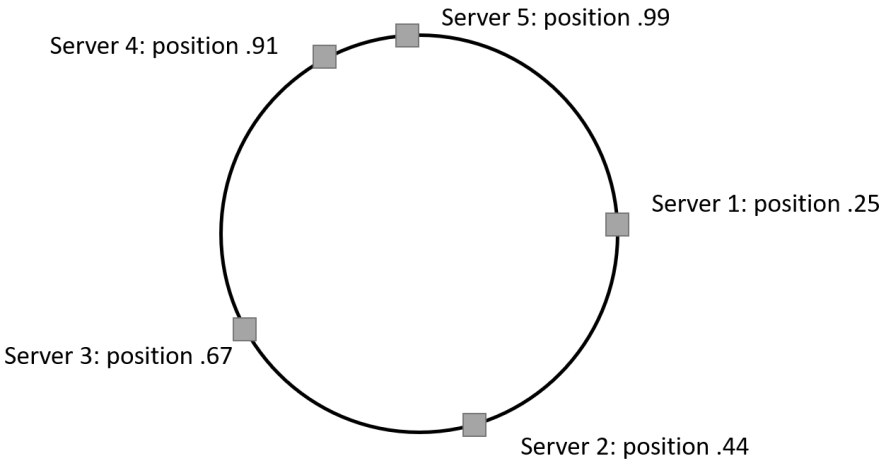
FOR $C in doc("colors.xml")/Color
WHERE $C/Category = "hue"
RETURN $C

(Many ways to do it. Also depends on the way you set up your schema. No worries about getting the exact syntax correct.)

b. Returns color names sorted by green content.

FOR $C in doc("colors.xml")/Color
ORDER BY $C//Greenvalue
RETURN $C/@name

4. For the following key-value sharding setup:



a. How would storage of keys be affected if server 2 were removed?

Keys with hash values from .25 to .44 which were previously being stored at server 2 would need to be copied to server 3.

b. How would storage of keys be affected if a new server went up at position .78?

Keys from .67 to .78 that were being stored at server 4 would be transferred to the new server and stored there.