1) Read the data into a Dataframe.

```
In [2]: rawDF = spark.read.json("iot_devices.json")
        rawDF.show()
```

```
+-------------+---------+----+----+-------------+---------+--------------------+--------+--------------+--------+------+------
---+-------+----+-------------+
|battery_level|c02_level|cca2|cca3|           cn|device_id|         device_name|humidity|            ip|latitude|   lcd|longit
ude|  scale|temp|    timestamp|
+-------------+---------+----+----+-------------+---------+--------------------+--------+--------------+--------+------+------
---+-------+----+-------------+
|            8|      868|  US| USA|United States|        1|meter-gauge-1xbYRYcj|      51|  68.161.225.1|    38.0| green|    -9
7.0|Celsius|  34|1458444054093|
|            7|     1473|  NO| NOR|       Norway|        2|   sensor-pad-2n2Pea|      70| 213.161.254.1|   62.47|   red|
6.15|Celsius|  11|1458444054119|
|            2|     1556|  IT| ITA|        Italy|        3| device-mac-36TWSKiT|      44|     88.36.5.1|   42.83|   red|    1
2.83|Celsius|  19|1458444054120|
|            6|     1080|  US| USA|United States|        4|   sensor-pad-4mzWkz|      32|  66.39.173.154|   44.06|yellow|  -12
1.32|Celsius|  28|1458444054121|
|            4|      931|  PH| PHL|  Philippines|        5|therm-stick-5gimp...|      62|    203.82.41.9|   14.58| green|   12
0.97|Celsius|  25|1458444054122|
|            3|     1210|  US| USA|United States|        6|sensor-pad-6al7RT...|      51| 204.116.105.67|   35.93|yellow|   -8
5.46|Celsius|  27|1458444054122|
|            3|     1129|  CN| CHN|        China|        7|meter-gauge-7GeDoanM|      26|  220.173.179.1|   22.82|yellow|   10
8.32|Celsius|  18|1458444054123|
|            0|     1536|  JP| JPN|        Japan|        8|sensor-pad-8xUD6p...|      35|  210.173.177.1|   35.69|   red|   13
9.69|Celsius|  27|1458444054123|
|            3|      807|  JP| JPN|        Japan|        9| device-mac-9GcjZ2pw|      85|  118.23.68.227|   35.69| green|   13
9.69|Celsius|  13|1458444054124|
|            7|     1470|  US| USA|United States|       10|sensor-pad-10Bsyw...|      56|208.109.163.218|   33.61|   red|  -11
1.89|Celsius|  26|1458444054125|
|            3|     1544|  IT| ITA|        Italy|       11|meter-gauge-11dlM...|      85|  88.213.191.34|   42.83|   red|    1
```

2. Convert the Dataframe into a temporary view called iot.

3. Count how many devices are there from each country and display the output.

```
rawDF.createOrReplaceTempView("iot_devices")
```

```
spark.sql("SELECT cn , COUNT(cn) FROM iot_devices GROUP BY cn").show(5, False)
```

```
+------------------+---------+
|cn                |count(cn)|
+------------------+---------+
|Russia            |5989     |
|Paraguay          |32       |
|Anguilla          |7        |
|Macao             |33       |
|U.S. Virgin Islands|51       |
+------------------+---------+
only showing top 5 rows
```

4.Display all the countries whose carbon dioxide level is more than 1400. Sort the output in descending order.

```
spark.sql("SELECT cn, c02_level FROM iot_devices WHERE c02_level > 1400 ORDER BY c02_level DESC").show(5, False)
```

```
+-------------+---------+
|cn           |c02_level|
+-------------+---------+
|China        |1599     |
|Sweden       |1599     |
|China        |1599     |
|United States|1599     |
|United States|1599     |
+-------------+---------+
only showing top 5 rows
```

5.Select all countries' devices with high-levels of C02 and group by cca3 and order by device_ids (Hint: For high CO2 level, the LCD status will be RED).

```
spark.sql("SELECT COUNT(device_id),cca3 from iot_devices where lcd == 'red' group by cca3 ORDER BY COUNT(device_id) DESC").show()
```

```
+----------------+----+
|count(device_id)|cca3|
+----------------+----+
|           17489| USA|
|            3616| CHN|
|            2942| KOR|
|            2935| JPN|
|            1966| DEU|
|            1660| GBR|
|            1564| CAN|
|            1508| RUS|
|            1353| FRA|
|             856| BRA|
|             769| AUS|
|             724| SWE|
|             713| ITA|
|             664| POL|
|             646| NLD|
|             586| ESP|
|             542| TWN|
|             446| IND|
|             399| NOR|
|             373| UKR|
+----------------+----+
```

6.find out all devices in countries whose batteries need replacements.

```
spark.sql("SELECT cn as country ,device_name , battery_level FROM iot_devices WHERE battery_level = 0 ").show(10,False)
```

```
+-----------------+-----------------------+-------------+
|country          |device_name            |battery_level|
+-----------------+-----------------------+-------------+
|Japan            |sensor-pad-8xUD6pzsQI  |0            |
|United States    |sensor-pad-12Y2kIm0o   |0            |
|United States    |meter-gauge-17zb8Fghhl |0            |
|Germany          |sensor-pad-448DeWGL    |0            |
|Canada           |sensor-pad-80TY4dWSMH  |0            |
|Republic of Korea|sensor-pad-92vxuq7e    |0            |
|United States    |sensor-pad-98mJQAfJpfW |0            |
|Japan            |meter-gauge-1075KSUDRjPa|0            |
|Australia        |device-mac-111WYtjxe1b |0            |
|United States    |sensor-pad-11663yUf    |0            |
+-----------------+-----------------------+-------------+
only showing top 10 rows
```