****需求****   
根据IP找到对应的城市   
  
****原来的解决方案****   
oracle表（ip\_country）：   
  
  
  
  
查询IP对应的城市：   
  
1.把a.b.c.d这样格式的IP转为一个数字，例如为把210.21.224.34转为3524648994   
2. select city from ip\_country where ipstartdigital <= 3524648994 and 3524648994 <=ipenddigital   
  
****redis解决方案****   
  
我们先把上面的表简化一下：

1. id  city    min max
2. 1   P1  0   100
3. 2   P2  101 200
4. 3   P3  201 300
5. 4   P4  301 400

（注意：min/max组成的range之间不能有重叠）   
  
主要思路就是用hmset存储表的每一行，并为每一行建立一个id（作为key）   
然后把ip\_end按顺序从小到大存储在sorted set当中，score对应该行的id   
查询时，利用redis sorted set的范围查询特性，从sorted set中查询到id，再根据id去hmget   
  
****实验****

1. //存储表的每一行
2. 127.0.0.1:6379> hmset {ip}:1 city P1 min 0 max 100
3. OK
4. 127.0.0.1:6379> hmset {ip}:2 city P2 min 101 max 200
5. OK
6. 127.0.0.1:6379> hmset {ip}:3 city P3 min 201 max 300
7. OK
8. 127.0.0.1:6379> hmset {ip}:4 city P4 min 301 max 400
9. OK
11. //建立sorted set（score-member，例如score=100，member=1等等）
12. 127.0.0.1:6379> zadd {ip}:end.asc 100 1 200 2 300 3 400 4
13. (integer) 4
14. 127.0.0.1:6379> zrange {ip}:end.asc 0 -1
15. 1) "1"
16. 2) "2"
17. 3) "3"
18. 4) "4"
20. //查询对应的区间（score）
21. 127.0.0.1:6379> zrangebyscore {ip}:end.asc 90 +inf LIMIT 0 1
22. 1) "1"
23. 127.0.0.1:6379> zrangebyscore {ip}:end.asc 123 +inf LIMIT 0 1
24. 1) "2"
25. 127.0.0.1:6379> zrangebyscore {ip}:end.asc 100 +inf LIMIT 0 1
26. 1) "1"
27. //解释：
28. //zrangebyscore {ip}:end.asc 90 +inf LIMIT 0 1
29. //表示查找score大于等于90的第一个值。（+inf在Redis中表示正无穷大）
30. //该语句返回值member=1，与hmset当中的id对应，因此可以通过hmget查找城市了：
32. //查找城市
33. 127.0.0.1:6379> hmget {ip}:1 city
34. 1) "P1"