

# Elasticsearch 聚合类型习题

## 一、度量聚集 (Metric Aggregations)

### 题目 1: 最小值与最大值

在一个索引中, 字段 'temperature' 表示某城市的每日温度。编写 Elasticsearch 查询, 使用度量聚集分别计算最高温度和最低温度。

```
POST index/_search
{
  "query": {
    "match_all": {}
  },
  "size": 0,
  "aggs": {
    "max_tem": {
      "max": {
        "field": "temperature",
        "missing": 0
      }
    },
    "min_tem": {
      "min": {
        "field": "temperature",
        "missing": 0
      }
    }
  }
}
```

### 题目 2: 唯一值统计与百分比计算

在一个索引中，字段 ‘user\_id’ 表示用户 ID，‘spend’ 表示用户的消费金额。(1) 使用 ‘cardinality’ 度量聚集计算唯一用户数量；(2) 使用百分比分位数聚集计算消费金额的 50% 和 95% 分位数。

```
POST index/_search
{
  "query":{
    "match_all": {}
  },
  "size": 0,
  "aggs": {
    "cardinality_user_id": {
      "cardinality": {
        "field": "user_id",
        "missing": 0
      }
    },
    "percentiles_spend": {
      "percentiles": {
        "field": "spend",
        "missing": 0
      }
    }
  }
}
```

### 题目 3: 平均值计算

假设我们有一个包含销售记录的索引，其中每条记录包含字段 ‘price’ 表示商品价格，‘quantity’ 表示商品数量。编写 Elasticsearch 查询，使用度量聚集计算商品价格的平均值。

```
POST index/_search
{
  "query": {
    "match_all": {}
  },
  "size": 0,
  "aggs": {
    "avg_price": {
      "avg": {
        "field": "price",
        "missing": 0
      }
    }
  }
}
```

#### 题目 4: 唯一值统计

在一个索引中，字段 ‘user\_id’ 表示用户标识，可能存在重复值。编写 Elasticsearch 查询，使用度量聚集计算唯一用户数量。

```
POST index/_search
{
  "query": {
    "match_all": {}
  },
  "size": 0,
  "aggs": {
    "cardinality_id": {
      "cardinality": {
        "field": "user_id"
      }
    }
  }
}
```

```

    }
  }
}
}

```

## 二、桶聚集 (Bucket Aggregations)

### 题目 1: 按术语分组统计

一个索引中包含字段 ‘category’ 和 ‘sales’，分别表示商品类别和销量。编写 Elasticsearch 查询，使用 ‘terms’ 桶聚集统计每个类别的总销量，输出最大的 5 个。

```

POST index/_search
{
  "query": {
    "match_all": {}
  },
  "size": 0,
  "aggs": {
    "group_by_category": {
      "terms": {
        "field": "category",
        "size": 5,
        "order": {
          "sum_price": desc
        }
      },
      "aggs": {
        "sum_price": {
          "sum": {

```

```

        "field": sales
      }
    }
  }
}
}

```

## 题目 2: 时间分桶与范围分桶

(1) 一个索引中，字段 ‘order\_date’ 表示订单日期，‘revenue’ 表示订单收入。编写 Elasticsearch 查询，使用 ‘date\_histogram’ 桶聚集按季度分组统计每季度的收入；

```

POST index/_search
{
  "query": {
    "match_all": {}
  },
  "size": 0,
  "aggs": {
    "date": {
      "date_histogram": {
        "field": "order_date",
        "calendar_interval": "quarter"
      },
      "aggs": {
        "sum_revenue": {
          "sum": {
            "field": "revenue",

```

```

        "missing": 0
      }
    }
  }
}

```

(2) 字段 'age' 表示用户年龄，编写查询使用 'range' 桶聚集按年龄范围（如 18-25, 26-35）统计用户数量。

```

POST index/_search
{
  "query": {
    "match_all": {}
  },
  "size": 0,
  "aggs": {
    "agg_range": {
      "range": {
        "field": "age",
        "ranges": [
          {"from": 18, "to": 25},
          {"from": 26, "to": 35}
        ]
      },
    },
    "aggs": {
      "user_count": {
        "value_count": {

```

```

        "field": "age"
      }
    }
  }
}

```

### 三、管道聚集 (Pipeline Aggregations)

#### 题目 1: 最大桶与平均桶

在一个索引中, 字段 'sales\_date' 表示销售日期, 'sales' 表示每日销量。

(1) 使用 'max\_bucket' 管道聚集计算销量最高的一周;

```

POST index/_search
{
  "query": {
    "match_all": {}
  },
  "size": 0,
  "aggs": {
    "sum_month": {
      "date_histogram": {
        "field": "age",
        "fixed_interval": "7d"
      },
      "aggs": {
        "sales_sum": {
          "sum": {

```

```

        "field": "sales"
      }
    }
  },
  "max_bkt": {
    "max_bucket": {
      "buckets_path": "sum_month>sales_sum"
    }
  }
}
}

```

(2) 使用 ‘avg\_bucket’ 管道聚集计算每周平均销量。

```

POST index/_search
{
  "query": {
    "match_all": {}
  },
  "size": 0,
  "aggs": {
    "sum_month": {
      "date_histogram": {
        "field": "age",
        "fixed_interval": "7d"
      },
      "aggs": {
        "sales_sum": {

```



```
        "sum": {
            "field": "sales"
        }
    },
    "max_bkt": {
        "avg_bucket": {
            "buckets_path": "sum_month>sales_sum"
        }
    }
}
```