

type，是一个 index 中用来区分类似的数据的，类似的数据，但是可能有不同的 fields，而且有不同的属性来控制索引建立，分词器

field 的 value，在底层的 Lucene 中建立索引的时候，全部是 opaque bytes 类型，不区分类型的

Lucene 是没有 type 的概念的，在 document 中，实际上将 type 作为一个 document 的 field 来存储，即 type，es 通过，_type 来进行 type 的过滤和筛选

一个 index 中的多个 type，实际上是放在一起存储的，因此一个 index 下，不能有多个 type 重名，而类型或者其他设置不同的，因为那样是无法处理的

注：elasticsearch 在逐渐弱化 type 的概念，并且 elasticsearch 6.0.1 之后的版本 一个 index 下面只能放一个 type ! ! ! !

```
1 PUT /my_index
2 {
3   "mappings": {
4     "male": {
5       "properties": {
6         "name": {
7           "type": "text"
8         },
9         "sex": {
10          "type": "text"
11        },
12        "addr": {
13          "type": "text"
14        }
15      }
16    },
17    "female": {
18      "properties": {
19        "name": {
20          "type": "text"
21        },
22        "sex": {
23          "type": "text"
24        },
25        "age": {
26          "type": "integer"
27        }
28      }
29    }
30  }
31 }
```

这样建立的数据，在 es 的底层也是柔和在一起的，例如

```
1 POST my_index/male
2 {
3   "name":"daiwei",
4   "sex":"male",
5   "addr": "nanchang jiangxi China"
6 }
7
8 POST my_index/female
9 {
10  "name":"xiaowang",
11  "sex":"female",
12  "age":20
13 }
```

在底层存储的方式是：

```
1 {
2   "_type":"male"
3   "name":"daiwei",
4   "sex":"male",
5   "addr": "nanchang jiangxi China",
6   "age":
7 }
8
9 {
10  "_type":"female"
11  "name":"xiaowang",
12  "sex":"female",
13  "addr": "",
14  "age":20
15 }
```

因此在 es 6.x之前 将类似的结构 的 type 放在一个 index 下，这些 type 应该有多个 field 是相同的

假如说，你将两个type的field 完全不同放在一个 index 下，那么每条数据都至少有一半的 field 在底层的 Lucene中是空值，会产生严重的性能问题。