# MySQL各个数据类型创建为索引所占用的字节数

意义：

KEY\_LEN代表的是数据存储时候的KEY的大小。**越小的key\_len 代表越小的存储空间 和更高的查询使用效率**

在判断复合索引时也可以通过执行计划的KEY\_LEN 来判断使用索引的效率

用varchar做主键的危害：

如用BINGINT 类型为主键 KEY的长度为8字节 ，在此表建立列为int型非空的二级索引，此时叶子节点大小为8+4=**12字节**

如果主键为VARCHAR（32）类型主键 KEY的长度为32\*3+2=98字节，在此表建立列为int型非空的二级索引，此时叶子节点大小为98+4=**102字节**

**两者对比不难看出有进10倍的差距，理论上效能也差距近10倍**

问题：

未解决大家在优化SQL语句时，需要计算key\_len的问题，测试了常用的数据类型在不同字符集下所占字节数。

各个数据类型的key\_len长度：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **数据类型** | **null/not null** | **字符集** | **key\_len** | **计算方式** |
| int | not null | - | 4 | 4 |
| int | null | - | 5 | 4+1 |
| bigint | not null | - | 8 | 8 |
| bigint | null | - | 9 | 8+1 |
| double | not null | - | 8 | 8 |
| double | null | - | 9 | 8+1 |
| float | not null | - | 4 | 4 |
| float | null | - | 5 | 4+1 |
| char（20） | not null | utf8 | 60 | 3\*20 |
| char（20） | null | utf8 | 61 | 3\*20+1 |
| char（20） | not null | gbk | 40 | 2\*20 |
| char（20） | null | gbk | 41 | 2\*20+1 |
| char（20） | not null | utf8mb4 | 80 | 4\*20 |
| char（20） | null | utf8mb4 | 81 | 4\*20+1 |
| char（20） | not null | latin1 | 20 | 1\*20 |
| char（20） | null | latin1 | 21 | 1\*20+1 |
| varchar（20） | not null | utf8 | 62 | 3\*20+2 |
| varchar（20） | null | utf8 | 63 | 3\*20+2+1 |
| varchar（20） | not null | gbk | 42 | 2\*20+2 |
| varchar（20） | null | gbk | 43 | 2\*20+2+1 |
| varchar（20） | not null | utf8mb4 | 82 | 4\*20+2 |
| varchar（20） | null | utf8mb4 | 83 | 4\*20+2+1 |
| varchar（20） | not null | latin1 | 22 | 1\*20+2 |
| varchar（20） | null | latin1 | 23 | 1\*20+2+1 |
| text(name(4)) | null | utf8 | 15 | 3\*4+2+1 |
| text(name(4)) | not null | utf8 | 14 | 3\*4+2 |
| text(name(4)) | null | gbk | 11 | 2\*4+2+1 |
| text(name(4)) | not null | gbk | 10 | 2\*4+2 |
| date | null | - | 4 | 3+1 |
| date | not null | - | 3 | 3 |
| time | null | - | 4 | 3+1 |
| time | not null | - | 3 | 3 |
| datetime | null | - | 6 | 5+1 |
| datetime | not null | - | 5 | 5 |
| timestamp | null | - | 4 | 4 |
| timestamp | not null | - | 4 | 4 |

## 总结 ：

**key\_len的长度计算公式：**  
**varchr(10)变长字段且允许NULL    =  10 \* ( character set：utf8=3,gbk=2,latin1=1)+1(NULL)+2(变长字段)  
varchr(10)变长字段且不允许NULL =  10 \*( character set：utf8=3,gbk=2,latin1=1)+2(变长字段)  
  
char(10)固定字段且允许NULL        =  10 \* ( character set：utf8=3,gbk=2,latin1=1)+1(NULL)  
char(10)固定字段且不允许NULL        =  10 \* ( character set：utf8=3,gbk=2,latin1=1)**