

目 录

Qt页面相关

算法

python相关

Qt页面相关

删除布局中的所有子控件

```
1. void clearLayout(QLayout* layout)
2. {
3.     if (layout) {
4.         QLayoutItem* child = nullptr;
5.         while ((child = layout->takeAt(0)) != nullptr) {
6.             if (child->widget() != nullptr)
7.                 delete child->widget();
8.             delete child;
9.         }
10.    }
11. }
```

算法

快速排序

```

1. // 格式化后:
2. //backward --> forward <---
3. inline void moveNumber(vector<int>& nums,int& i,int& j,int baseNum,bool
   backward)
4. {
5.     while(i < j && backward ? (nums[i] < baseNum) :(nums[j] > baseNum)
6.         backward ? ++i:--j;
7.     if(i < j)
8.         nums[backward?j:i] = nums[backward?i:j];
9. }
10.
11. void sort(vector<int>& nums,int begin,int end)
12. {
13.     if(begin >= end)
14.         return;
15.
16.     const int baseNum = nums[begin];
17.     int i = begin;
18.     int j = end;
19.
20.     while(i < j)
21.         for(auto state:{false,true})
22.             moveNumber(nums,i,j,baseNum,state);
23.
24.     nums[i] = baseNum;
25.     sort(nums,begin,i - 1);
26.     sort(nums,i+1 ,end);
27. }
28.
29. // 默认解法:
30. void sort(vector<int>& nums,int begin,int end)
31. {
32.     if(begin >= end)
33.         return;
34.
35.     int baseNum = nums[begin];
36.     int i = begin;
37.     int j = end;
38.
39.     while(i < j)
40.     {
41.         while(i < j && nums[j] > baseNum)
42.             --j;
43.         if(i < j)
44.             nums[i] = nums[j];
45.

```

```
46.     while(i < j  && nums[i] < baseNum)
47.         ++i;
48.     if(i < j)
49.         nums[j] = nums[i];
50.     }
51.     nums[i] = baseNum;
52.     sort(nums, begin, i - 1);
53.     sort(nums, i+1, end);
54. }
```

python相关

获取**Dict**中的数据值

```
1. def get_dict_value(dict:Dict, key, default_value = None):  
2.     if key not in dict.keys():  
3.         return default_value  
4.     return dict[key]
```