

## 8.3 创建多个线程\_物联网 / 嵌入式工程师 - 慕课网

“ 慕课网慕课教程 8.3 创建多个线程涵盖海量编程基础技术教程，以图文图表的形式，把晦涩难懂的编程专业用语，以通俗易懂的方式呈现给用户。

- 创建多个线程时，一般由主线程统一创建，并等待释放资源或者分离线程，不要递归创建
- 具体示例如下:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <pthread.h>

void *do_thread_one(void *arg)
{
    printf("Thread one start.\n");
    pthread_exit(NULL);
}

void *do_thread_two(void *arg)
{
    printf("Thread two start.\n");
    pthread_exit(NULL);
}

int main(void)
{
    int err;
    pthread_t tid_one = 0, tid_two = 0;

    err = pthread_create(&tid_one, NULL, do_thread_one, NULL);
    if (err != 0){
        fprintf(stderr, "[ERROR] pthread_create : < %s >\n", strerror(err));
        exit(EXIT_FAILURE);
    }

    err = pthread_create(&tid_two, NULL, do_thread_two, NULL);
    if (err != 0){
        fprintf(stderr, "[ERROR] pthread_create : < %s >\n", strerror(err));
        exit(EXIT_FAILURE);
    }

    pthread_join(tid_one, NULL);
    pthread_join(tid_two, NULL);

    return 0;
}
```

### 注意

多个线程如果任务相同，则可以使用同一个线程执行函数

### 示例

创建多个线程，并等待两个子线程退出

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <pthread.h>
#include <unistd.h>
```

```
void *do_thread_one(void *arg)
{
    int i;
    for (i = 0; i < 5; i++){
        printf(" i = %d\n", i);
        sleep(1);
    }
    pthread_exit(NULL);
}

int main(void)
{
    int err, i;

    pthread_t tid[2] = {0};

    for (i = 0; i < 2; i++){
        err = pthread_create(&tid[i], NULL, do_thread_one, NULL);
        if (err != 0){
            fprintf(stderr, "[ERROR] pthread_create : < %s >\n", strerror(err));
            exit(EXIT_FAILURE);
        }
    }

    pthread_join(tid[0], NULL);
    pthread_join(tid[1], NULL);

    return 0;
}
```

---

全文完

本文由 简悦 SimpRead 优化，用以提升阅读体验

使用了 全新的简悦词法分析引擎 beta，[点击查看详细说明](#)

