**通过 Callable 和 Future 创建线程**

1. 创建 Callable 接口的实现类，并实现 call() 方法，该 call() 方法将作为线程执行体，并且有返回值。

2. 创建 Callable 实现类的实例，使用 FutureTask 类来包装 Callable 对象，该 FutureTask 对象封装了该 Callable 对象的 call() 方法的返回值。

3. 使用 FutureTask 对象作为 Thread 对象的 target 创建并启动新线程。

4. 调用 FutureTask 对象的 get() 方法来获得子线程执行结束后的返回值。

实例

public class CallableThreadTest implements Callable<Integer> {

public static void main(String[] args)

{

CallableThreadTest ctt = new CallableThreadTest();

FutureTask<Integer> ft = new FutureTask<>(ctt);

for(int i = 0;i < 100;i++)

{

System.out.println(Thread.currentThread().getName()+" 的循环变量i的值"+i);

if(i==20)

{

new Thread(ft,"有返回值的线程").start();

}

}

try

{

System.out.println("子线程的返回值："+ft.get());

} catch (InterruptedException e)

{

e.printStackTrace();

} catch (ExecutionException e)

{

e.printStackTrace();

}

}

@Override

public Integer call() throws Exception

{

int i = 0;

for(;i<100;i++)

{

System.out.println(Thread.currentThread().getName()+" "+i);

}

return i;

}

}