Custom::

ThreadPoolExecutor executorService=new ThreadPoolExecutor(csize,msize,kalive,

TimeUnit.SECONDS,queue,threadFactory,rejectionHandler);

public class RejectionHandler implements RejectedExecutionHandler {  
 @Override  
 public void rejectedExecution(Runnable r, ThreadPoolExecutor executor) {  
 //System.out.println(r+"is Rejected");  
 try{  
 executor.getQueue().put(r);  
 } catch (InterruptedException e) {  
 e.printStackTrace();  
 }  
 }  
}

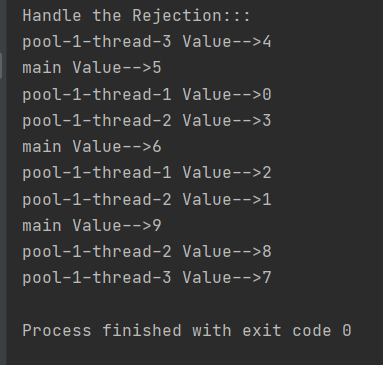
OutPut



CallerRunsPolicy::

ThreadPoolExecutor executorService=new ThreadPoolExecutor(csize,msize,kalive,TimeUnit.SECONDS,queue,threadFactory,new ThreadPoolExecutor.CallerRunsPolicy());

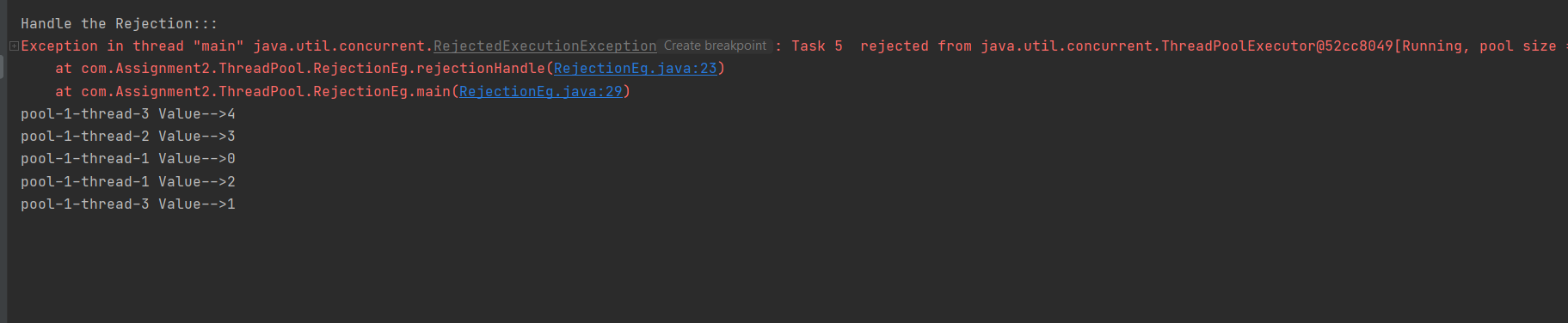
Output:



AbortPolicy::

ThreadPoolExecutor executorService=new ThreadPoolExecutor(csize,msize,kalive,TimeUnit.SECONDS,queue,threadFactory,new ThreadPoolExecutor.AbortPolicy());

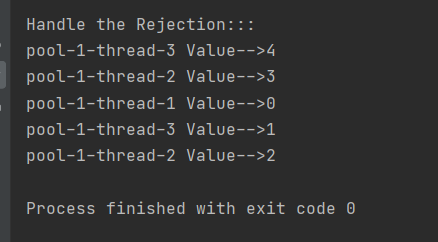
Output:



DiscardPolicy::

ThreadPoolExecutor executorService=new ThreadPoolExecutor (csize, msize,kalive, TimeUnit.SECONDS ,queue,threadFactory,new ThreadPoolExecutor.DiscardPolicy());

Output::



DiscardOldestPolicy::

ThreadPoolExecutor executorService=new ThreadPoolExecutor (csize, msize,kalive, TimeUnit.SECONDS ,queue,threadFactory,new ThreadPoolExecutor. DiscardOldestPolicy ());

Output::

