

它的作用是捕获这个promise对象的整个链式操作中由reject抛出的异常

```
function loadScript(src) {
  return new Promise((resolve, reject) => {
    let script = document.createElement('script');
    script.src = src;
    script.type = "text/javascript";
    document.body.appendChild(script);
    script.onload = () => {
      resolve(src)
    }
    script.onerror = (err) => {reject(err)}
  })
}

//
loadScript('/static/1.js').then(loadScript('/static/2.js')).then(loadScript('/static/3.js'))
loadScript('/static/1.js')
  .then(() => {
    return loadScript('/static/4.js')
  })
  .then(() => {
    loadScript('/static/3.js')
  })
  .catch((err) => {
    console.log(err);
  })
})
```