|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 注入和读取Req,res  数据 | 注入方式 | 请求方式 | 注入到request对象 ctx.request ctx.request.xxx都返回对象类型 | koa中间件 | 读取request对象 ctx.request ctx.request.xxx都返回对象类型 | Koa注入body对象到response  注: ctx.response.body=== ctx.body===ctx.state ===response.data  ctx.response.body可以返回任何类型 | | Axios读取response |
| headers数据 | Ajax |  | **头文件:**  用于后端做程序员做判断 可跨域? 类型改? 已登录?  第一种附加头文件:扩展axios  axios.defaults.headers.common['Authorization'] = 'AUTH\_TOKEN'; axios.defaults.headers.post['Content-Type']='application/x-www-form-urlencoded';  第二种附加头文件:拦截器  axios.interceptors.request() | 自带 | ctx.request.header //等于ctx.req.headers | ctx.response.body = {goods:’data’,   code: 0,  msg:’msg’ } ctx.response.status=201 | ctx.state.code=0  ctx.state.msg=’msg’  ctx.state.goods=[]  ctx.status=201 | .then(response=>{response.data.msg}) |
| query[英]问号,字符串数据 | ajax | Get | axios.get(‘/a’,{params:{id:1}}) | 自带 | ctx.request.query |
| axios.get('/a?name=小李') //字符串不带引号,若带引号会错误连引号一起搜索 |
| Json数据 | ajax | Post | axios.post(‘/a’, {a:1}） //不可传送数组,例如axios.post(‘/a’, [1,2,3]) ,因为ctx.request.body必须是对象类型的 | koa-body | ctx.request.body  //这里不能省去request |
| delete | axios.delete('http://127.0.0.1:3000/api/request/', {data: {id: 1}}) |
| put | axios.put('http://localhost:3000/api/request',{a:1}) |
| patch | axios.patch('http://localhost:3000/api/request',{a:1}) |
| 表单对象FormData数据 | ajax |  | 第三种附加头文件:  axios.post(‘/a’,{img:Filetype(….)},{ headers:{ ‘Content-Type’:‘multipart/form-data’}}) | koa-body | ctx.request.files |
| 表单数据 | action(重绘) |  | <form action="http://localhost:3000/api/request-file/" method="post" enctype="multipart/form-data">  </form> | koa-body | ctx.request.body |
| Xml数据 | ajax |  | axios.post(‘/a’, xmlString,{ headers: {'Content-Type': 'text/xml'}}） | koa-bodyparser+  koa-xml-body | ctx.request.body |
| params 字符串数据 | ajax |  | axios.post(`/delCategory/${row.\_id}`) | 自带 | ctx.params  //params不是request的key,不是简写,也返回对象 |

|  |  |
| --- | --- |
| **request** | **response** |
|  |  |