1. Ajax：Asynchronous JavaScript And XML，异步的JS和XML。可以无刷新获取数据
   1. 优缺点：无需刷新花去数据、可以根据用户事件更新部分页面内容；没有浏览历史不能回退、存在跨域问题（同源）、SEO不友好
   2. 跨域解决方案：JSONP（见计算机网络笔记）、CORS（见服务端笔记）、代理服务器（见Vue笔记）
   3. ajax中若url是满足同源策略的，可以省略域名，只写路径
   4. 使用：
      1. const xhr = new XMLHttpRequest() 创建对象
      2. xhr.open('Get', 'http://127.0.0.1:8000/server') 初始化 设置请求方法和ur
      3. xhr.send() 发送
      4. xhr.onreadystatechange = function ()｛ 事件绑定，处理服务器返回的结果
         1. if（xhr.readyState === 4）{ 判断服务端是否返回了所有的结果
            1. if（xhr.status >= 200 && xhr.status < 300）{ 判断响应状态码是否为成功（2xx）

。。。

* + - * 1. }
      1. }
    1. ｝
  1. 属性：设对象为xhr
     1. .status 响应状态码 .statusText 响应字符串 response 响应体
     2. .timeout 设置超时毫秒数
  2. 方法：设对象为xhr
     1. .open（‘请求类型’，‘url’） 初始化
     2. .setRequesetHeader（‘属性名’，‘值’） 设置请求头的属性
     3. .send（[‘参数’]） 发送请求，POST可以传参
     4. .abort（） 取消请求
     5. .getAllResponseHeaders() 返回整个响应头
  3. 使用fetch函数发送请求：返回Promise对象，ie不兼容
     1. 与jQuery.ajax（）相比：
        1. ①收到一个代表错误的HTTP状态码时，返回的Promise状态仍为fulfilled，但返回值的ok属性为false，仅网络故障或请求被阻止时，才会标记为rejected；②可以接受跨域cookies；③不发送cookies
     2. fetch（‘url’，配置对象）
        1. 配置对象的属性：
           1. method：‘POST’、headers：｛｝、
           2. body：｛｝/ ‘xx’ 请求体信息，可为对象可为字符串
     3. fetch（）.then（response=>{}） 响应处理函数
  4. 事件：
     1. timeout 超时 readystatechange XMLHttpRequest（readyState）变化
  5. IE缓存问题：IE会把ajax请求缓存，再次向服务器请求时，直接读取。会有时效性信息无法获取的情况
     1. 解决方法：传输一个属性，t=date.now（）
  6. jQuery中的AJAX：
     1. $.get（‘url’，对象，回调函数[，‘响应体类型（如json）’]） 发送get请求，post同理
        1. 对象：｛a：100，b：200｝
        2. 回调函数：可传入变量data，表示响应体
        3. 响应体类型，若为json，data会自动转换为对象
     2. $.ajax（对象） 通用的ajax方法
        1. 对象的属性：
           1. url：xx，data：｛xx｝，type：‘GET’，success：fn（data）｛成功的回调｝，headers：｛xx：xx｝，
           2. dataType：‘响应体类型’，timeout：2000，error：fn（xhr，textStatus，errorThrown）｛失败的回调｝
  7. axios中的AJAX（**推荐**）：npm i axios
     1. axios.default.baseURL = ‘域名’ 若设置了baseURL，则下面的url可以省去baseURL中的部分
     2. axios.get（‘url’，配置对象） 返回promise对象
        1. 配置对象的属性：
           1. param：｛xx｝ 通过url传递的参数
           2. headers：｛xx｝ 请求头信息
     3. axios.post（‘url’，｛请求体属性｝，配置对象同上） 返回promise对象
     4. axios（对象） 通用请求函数，返回promise对象
        1. 对象的属性：
           1. method：‘POST’、url：‘xx’、params：｛｝、headers：｛｝、data：｛｝
     5. axios.get（xxx）.then（response=>｛。。。｝） 响应处理函数，response变量为一个对象，包含响应信息