Cookie：用户数据存储在客户端

参考文章1
参考文章2

- 客户端第一次访问：服务端返回的响应头中加上Set-Cookie请求头

属性	说明
name=value	cookie键值对
domain	指定cookie的域名，默认是当前域名，一级域名和二级域名之间允许共享使用， 否则cookie不能跨域名
path	指定cookie在哪个路径下生效，默认是 `/` , 如果设置为 `/abc` , 则只有 `/abc` 下的路由可以访问到该cookie，例如： `/abc/read`
maxAge	cookie有效时长，单位是秒，默认为是-1（浏览器关闭即失效）
expires	设置cookie过期时间
secure	默认为false， 如果为true, 则cookie只在https中生效
httpOnly	如果给某个 cookie 设置了 httpOnly 属性，则无法通过 JS 脚本 读取到该 cookie 的信息，但还是能通过 Application 中手动修改 cookie， 所以只是在一定程度上可以防止 XSS 攻击，不是绝对的安全

- 前端操作cookie:

  ```
  document.cookie = 'key=value'
  ```

  直接设置document.cookie不会覆盖原有cookie，而是添加新cookie，除非设置的cookie已经存在
- 后续发起请求时，请求头的cookie头会附上所有未过期的cookie

session：用户数据存储在服务端，客户端只存储用户标识

- 客户端第一次访问时，服务端生成sessionId，将sessionId附在set-cookie头中返回给客户端
- 客户端后续每次发起请求，都将sessionId作为cookie带入请求头中
- 客户数据存在服务端，用sessionId来标识是哪个用户的数据

token：

- 客户端第一次访问时，服务端验证完用户身份后，将用户标识（用户id）进行加密签名，将用户标识和签名一起，作为会话的token（令牌）附在响应头中
- 客户端后续发起请求，都携带token，服务端从token中取出用户id，用生成签名的算法重新计算一次，看看与token中携带的签名是否一致，如果一致，说明已经登陆过了
  参考文章