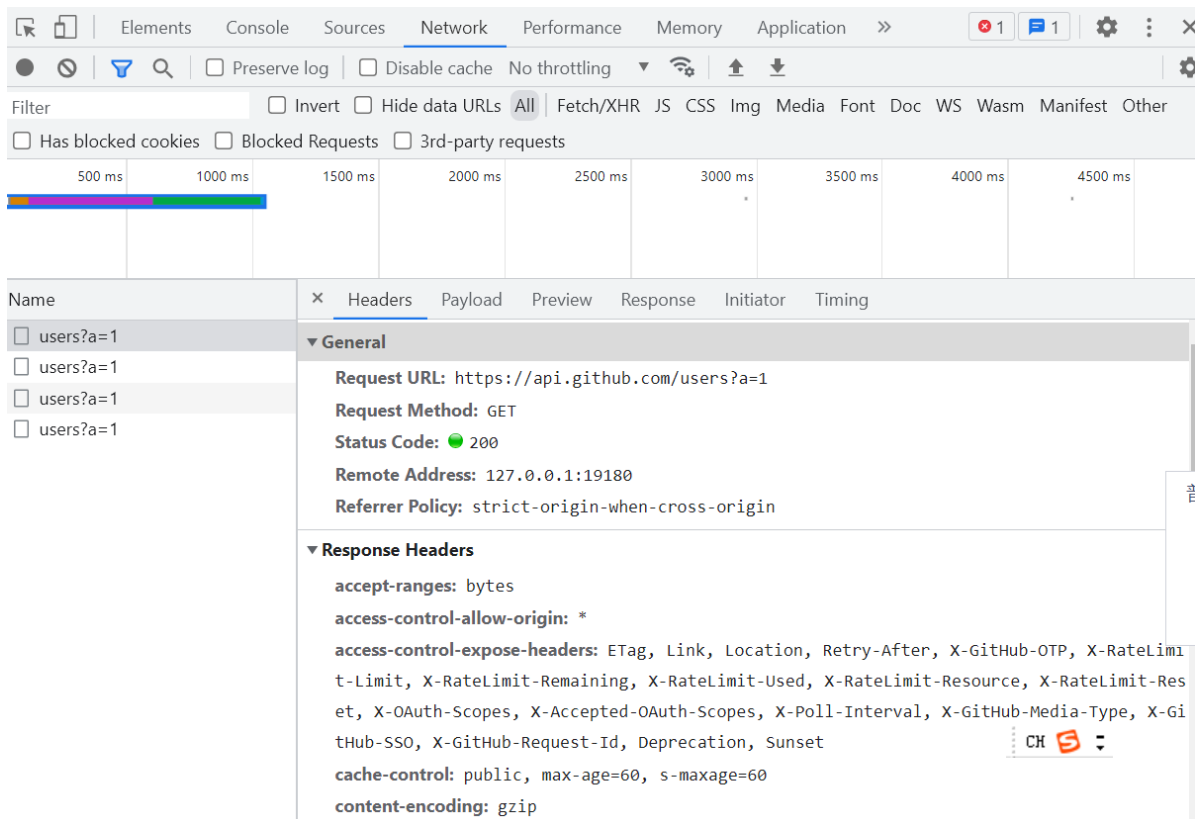
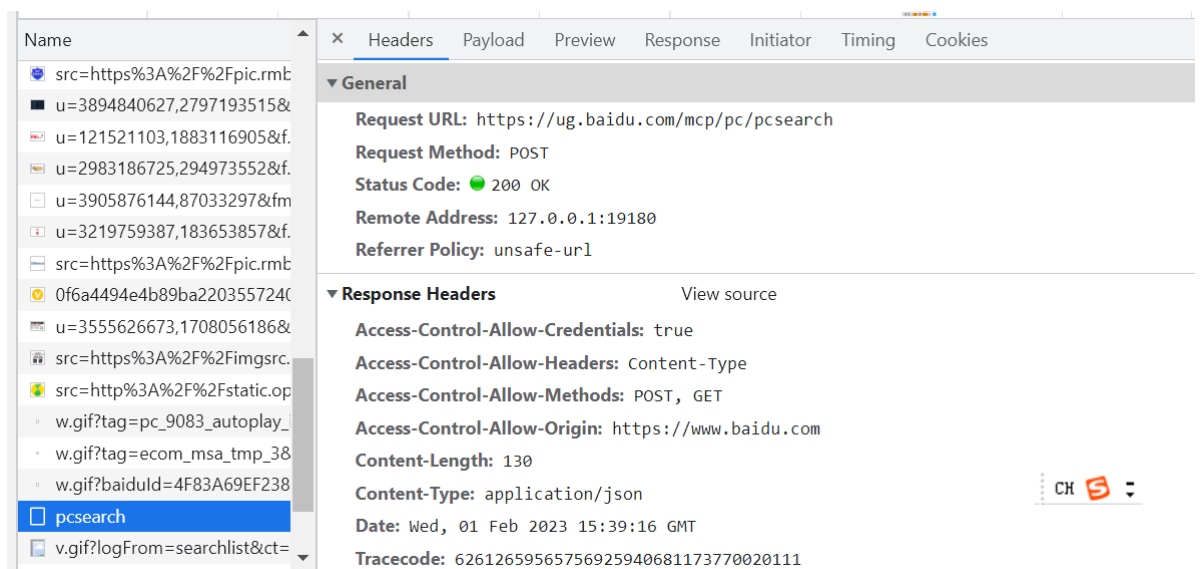


1. 针对任意网页，调研其不同方式请求，  
至少包括get、post请求，写出或截图其请求及相应数据包
2. 针对任意网页，使用Jquery，能够出发某一事件，  
写出至少三条语句，截图响应前后不同的状态
3. 完成一个浏览器插件，功能不限，文档中写明功能及代码

1, (1)get请求



(2)post请求（下图是在百度提交搜索时产生的post请求）



关于get和post请求的对比：

Get 是用来从服务器上获得数据,而 Post 是用来向服务器上传递数据。

1. GET在浏览器回退时是无害的，而POST会再次提交请求。
2. GET产生的URL地址可以被Bookmark，而POST不可以。
3. GET请求会被浏览器主动cache，而POST不会，除非手动设置。
4. GET请求只能进行url编码，而POST支持多种编码方式。
5. GET请求参数会被完整保留在浏览器历史记录里，而POST中的参数不会被保留。
6. GET请求在URL中传送的参数是有长度限制的，而POST么有。
7. 对参数的数据类型，GET只接受ASCII字符，而POST没有限制。
8. GET比POST更不安全，因为参数直接暴露在URL上，所以不能用来传递敏感信息。
9. GET参数通过URL传递，POST放在Request body中。

(本标准答案参考自w3schools)

2, 在jQuery中，触发事件有3种方式，第1种是调用事件方法，第2种是通过trigger()方法触发事件，第3种是通过triggerHandler()方法触发事件。

代码如下：

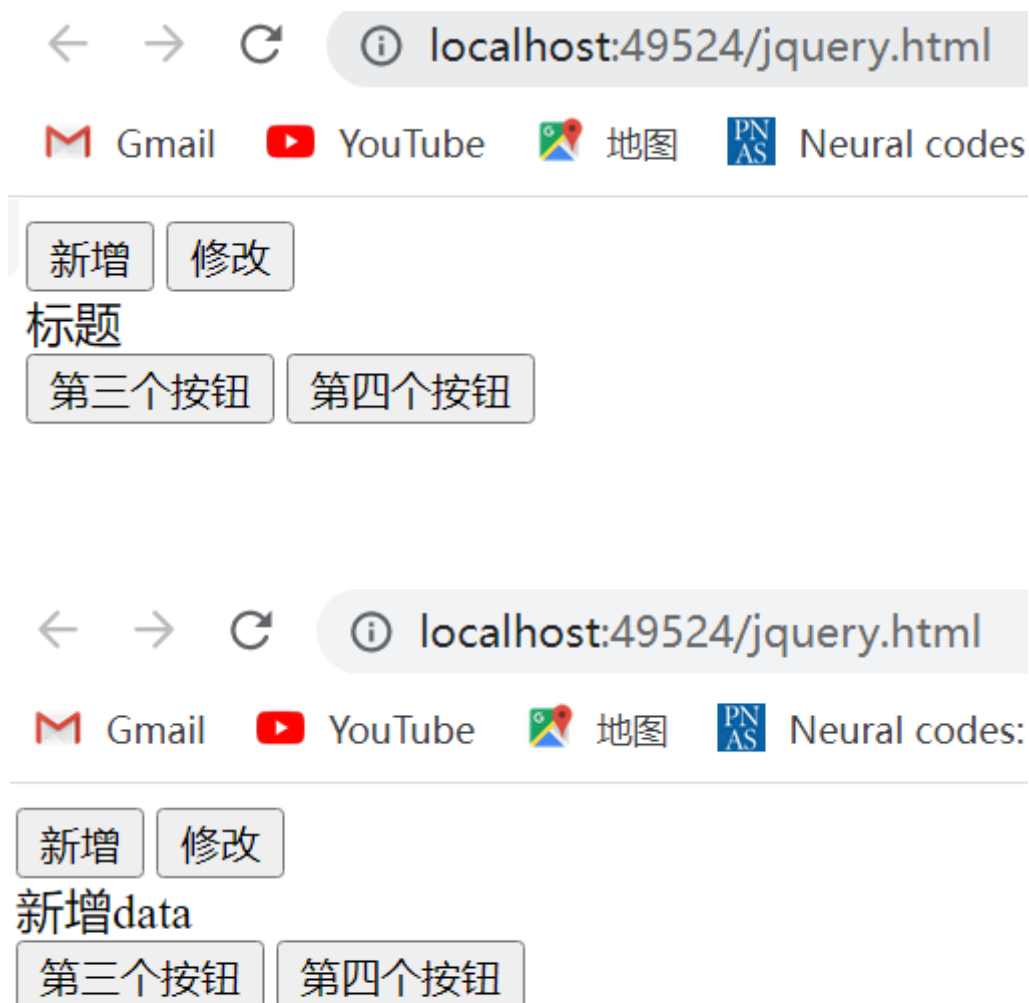
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>01-事件的基本使用.html</title>
</head>
<body>
  <button>新增</button>
  <button>修改</button>
  <div>标题</div>
  <script src="https://cdn.bootcdn.net/ajax/libs/jquery/1.12.4/jquery.js">
</script>
  <script>
    $(function () {
      // 第一个参数 事件类型 第二个参数(可选的) 要传递的参数 第三个参数 事件处理函数
      /*
      $('button').bind('click', '123', function (event) {
        console.log('button点击');
        console.log(event);
        console.log(event.data); //123
      })
      */
      // 点击新增或者修改按钮，改变div内部的内容
      $('button').bind('click', function (event) {
        // 通过event.target(DOM节点，可以拿到内部的内容) 来判断当前是哪个按钮点击
        // console.log(event.target);
        // console.log(event.target.innerHTML);
        if ($(event.target).html() === '新增') {
          $('div').html('新增data')
        } else {
          $('div').html('修改data')
        }
      })
      $("#btn3").click(function () {
        $(this).css("backgroundColor", "red");
      });
    });
  </script>
</body>
</html>
```

```

        $("#btn4").click(function () {
            $("#btn3").triggerHandler("click");
        });
        // 解绑 unbind() 无参时 解除绑定的所有事件
        // $('button').unbind()
        // 解绑 unbind('click') 传递参数->事件类型
        // $('button').unbind('click')
        // 解绑 unbind() 解绑click对应的事件
        function clickEvent() {
            console.log('我是click事件');
        }
        $('button').click(clickEvent)
        // 第一个参数 事件类型 第二个参数 事件名称
        $('button').unbind('click', clickEvent)
    })
</script>
<input type="button" value="第三个按钮" id="btn3">
<input type="button" value="第四个按钮" id="btn4">
</body>
</html>

```

效果:




← → ↻ ⓘ localhost:49524/jquery.html

 Gmail  YouTube  地图  Neural code

新增 修改

修改data

第三个按钮 第四个按钮

← → ↻ ⓘ localhost:49524/jquery.html 

 Gmail  YouTube  地图  Neural codes: Firi..

新增 修改

修改data

第三个按钮 第四个按钮