1、使用flex实现
```
<style>
        .container{ display: flex; }
        .left{ width: 100px; height: 500px; background: blue; }
        .right{ width: 100px; height: 500px; background: yellow; }
        .center{ height: 500px; flex: 1; background: red; }
</style>
<div class="container">
        <div class="left">欢迎您</div>
        <div class="center">中</div>
        <div class="right">退出登录</div>
</div>
```
使用grid实现
```
<style>
        .container {
                display: grid; /* 将其划分为两行，其中一列有本身宽度决定，一列占剩余宽度*/
                grid-template-columns: auto 1fr auto;
        }
        .item{ height: 80px; }
        .left,.right{ width: 200px; background-color: green; }
        .center{ background-color: red; }
</style>
```
2、使用reduce
```
let obj = list.reduce((obj, value) =>{ obj[value.lang] ? obj[value.lang]++ : obj[value.lang] = 1 return obj }, {});
const result = list.reduce((acc, curr) => {
        if(!acc[curr.lang]) { acc[curr.lang] = 1;
} else {
        acc[curr.lang]++;
} return acc; }, {});
```
3、使用Vue-cli 3.x代理方式
```
module.exports = {
        devServer: {
                port: 8000,
                proxy: {
                        "/api": {
                                target: "http://localhost:8080"
                        }
                }
        }
};
```
Node中CORS的解决代码
```
app.use(async (ctx, next) => {
  ctx.set("Access-Control-Allow-Origin", ctx.headers.origin);
  ctx.set("Access-Control-Allow-Credentials", true);
  ctx.set("Access-Control-Request-Method", "PUT,POST,GET,DELETE,OPTIONS");
  ctx.set(
    "Access-Control-Allow-Headers",
    "Origin, X-Requested-With, Content-Type, Accept, cc"
  );
```

```
  if (ctx.method === "OPTIONS") {
    ctx.status = 204;
    return;
  }
  await next();
});
```
禁用chrome安全策略。.\Google\Chrome\Application\chrome.exe --disable-web-security --user-data-dir=xxxx

使用自己的代理工具（如：charles，fiddler）。