code/src/csapp.c

```
1  int open_listenfd(char *port)
2  {
3      struct addrinfo hints, *listp, *p;
4      int listenfd, optval=1;
5
6      /* Get a list of potential server addresses */
7      memset(&hints, 0, sizeof(struct addrinfo));
8      hints.ai_socktype = SOCK_STREAM;             /* Accept connections */
9      hints.ai_flags = AI_PASSIVE | AI_ADDRCONFIG; /* ... on any IP address */
10     hints.ai_flags |= AI_NUMERICSERV;            /* ... using port number */
11     Getaddrinfo(NULL, port, &hints, &listp);
12
13     /* Walk the list for one that we can bind to */
14     for (p = listp; p; p = p->ai_next) {
15         /* Create a socket descriptor */
16         if ((listenfd = socket(p->ai_family, p->ai_socktype, p->ai_protocol))
17             < 0) continue;  /* Socket failed, try the next */
18
19         /* Eliminates "Address already in use" error from bind */
20         Setsockopt(listenfd, SOL_SOCKET, SO_REUSEADDR,
21                    (const void *)&optval , sizeof(int));
22
23         /* Bind the descriptor to the address */
24         if (bind(listenfd, p->ai_addr, p->ai_addrlen) == 0)
25             break; /* Success */
26         Close(listenfd); /* Bind failed, try the next */
27     }
28
29     /* Clean up */
30     Freeaddrinfo(listp);
31     if (!p) /* No address worked */
32         return -1;
33
34     /* Make it a listening socket ready to accept connection requests */
35     if (listen(listenfd, LISTENQ) < 0) {
36         Close(listenfd);
37         return -1;
38     }
39     return listenfd;
40 }
```

code/src/csapp.c

图 11-19　open_listenfd：打开并返回监听描述符的辅助函数。它是可重入和与协议无关的

最后，我们调用 listen 函数，将 listenfd 转换为一个监听描述符，并返回给调用者。如果 listen 失败，我们要小心地避免内存泄漏，在返回前关闭描述符。

11.4.9　echo 客户端和服务器的示例

学习套接字接口的最好方法是研究示例代码。图 11-20 展示了一个 echo 客户端的代