

```
1  #include "pch.h"
2  #include "CTextHandler.h"
3
4  // Public constructors
5  CTextHandler::CTextHandler()
6  {
7      this->shft = FALSE;
8  }
9  CTextHandler::~CTextHandler()
10 {
11 }
12
13 // Public commands
14 CString CTextHandler::RecieveText()
15 {
16     CString temp = this->text;
17     this->text = "";
18     return temp;
19 }
20 void CTextHandler::DestroyText()
21 {
22     this->text = "";
23 }
24
25 // Public message handlers
26 BOOL CTextHandler::OnKeyDown(UINT nChar, UINT nRepCnt, UINT nFlags)
27 {
28     BOOL caps = GetKeyState(VK_CAPITAL);
29
30     switch (nChar) {
31
32     case VK_SPACE:
33         this->text.Append(L" ");
34         break;
35
36     case VK_SHIFT:
37         this->shft = TRUE;
38         break;
39
40     case 0x30:
41         if (this->shft) {
42             this->text.Append(L"0");
43         }
44         else {
45             this->text.Append(L"0");
46         }
47         break;
48
49     case 0x31:
50         if (this->shft) {
51             this->text.Append(L"!");
52         }
53         else {
54             this->text.Append(L"1");
55         }
56         break;
```

```
57
58     case 0x32:
59         if (this->shft) {
60             this->text.Append(L"\");
61         }
62         else {
63             this->text.Append(L"2");
64         }
65         break;
66
67     case 0x33:
68         if (this->shft) {
69             this->text.Append(L"£");
70         }
71         else {
72             this->text.Append(L"3");
73         }
74         break;
75
76     case 0x34:
77         if (this->shft) {
78             this->text.Append(L"$");
79         }
80         else {
81             this->text.Append(L"4");
82         }
83         break;
84
85     case 0x35:
86         if (this->shft) {
87             this->text.Append(L"%");
88         }
89         else {
90             this->text.Append(L"5");
91         }
92         break;
93
94     case 0x36:
95         if (this->shft) {
96             this->text.Append(L"^");
97         }
98         else {
99             this->text.Append(L"6");
100        }
101        break;
102
103    case 0x37:
104        if (this->shft) {
105            this->text.Append(L"&");
106        }
107        else {
108            this->text.Append(L"7");
109        }
110        break;
111
112    case 0x38:
```

```
113         if (this->shft) {
114             this->text.Append(L"*");
115         }
116         else {
117             this->text.Append(L"8");
118         }
119         break;
120
121     case 0x39:
122         if (this->shft) {
123             this->text.Append(L "(");
124         }
125         else {
126             this->text.Append(L"9");
127         }
128         break;
129
130     case 0x41:
131         if (!(this->shft && caps) && (this->shft || caps)) {
132             this->text.Append(L"A");
133         }
134         else {
135             this->text.Append(L"a");
136         }
137         break;
138
139     case 0x42:
140         if (!(this->shft && caps) && (this->shft || caps)) {
141             this->text.Append(L"B");
142         }
143         else {
144             this->text.Append(L"b");
145         }
146         break;
147
148     case 0x43:
149         if (!(this->shft && caps) && (this->shft || caps)) {
150             this->text.Append(L"C");
151         }
152         else {
153             this->text.Append(L"c");
154         }
155         break;
156
157     case 0x44:
158         if (!(this->shft && caps) && (this->shft || caps)) {
159             this->text.Append(L"D");
160         }
161         else {
162             this->text.Append(L"d");
163         }
164         break;
165
166     case 0x45:
167         if (!(this->shft && caps) && (this->shft || caps)) {
168             this->text.Append(L"E");
```

```
169     }
170     else {
171         this->text.Append(L"e");
172     }
173     break;
174
175     case 0x46:
176         if (!(this->shft && caps) && (this->shft || caps)) {
177             this->text.Append(L"F");
178         }
179         else {
180             this->text.Append(L"f");
181         }
182         break;
183
184     case 0x47:
185         if (!(this->shft && caps) && (this->shft || caps)) {
186             this->text.Append(L"G");
187         }
188         else {
189             this->text.Append(L"g");
190         }
191         break;
192
193     case 0x48:
194         if (!(this->shft && caps) && (this->shft || caps)) {
195             this->text.Append(L"H");
196         }
197         else {
198             this->text.Append(L"h");
199         }
200         break;
201
202     case 0x49:
203         if (!(this->shft && caps) && (this->shft || caps)) {
204             this->text.Append(L"I");
205         }
206         else {
207             this->text.Append(L"i");
208         }
209         break;
210
211     case 0x4A:
212         if (!(this->shft && caps) && (this->shft || caps)) {
213             this->text.Append(L"J");
214         }
215         else {
216             this->text.Append(L"j");
217         }
218         break;
219
220     case 0x4B:
221         if (!(this->shft && caps) && (this->shft || caps)) {
222             this->text.Append(L"K");
223         }
224         else {
```

```
225         this->text.Append(L"k");
226     }
227     break;
228
229     case 0x4C:
230         if (!(this->shft && caps) && (this->shft || caps)) {
231             this->text.Append(L"L");
232         }
233         else {
234             this->text.Append(L"l");
235         }
236         break;
237
238     case 0x4D:
239         if (!(this->shft && caps) && (this->shft || caps)) {
240             this->text.Append(L"M");
241         }
242         else {
243             this->text.Append(L"m");
244         }
245         break;
246
247     case 0x4E:
248         if (!(this->shft && caps) && (this->shft || caps)) {
249             this->text.Append(L"N");
250         }
251         else {
252             this->text.Append(L"n");
253         }
254         break;
255
256     case 0x4F:
257         if (!(this->shft && caps) && (this->shft || caps)) {
258             this->text.Append(L"O");
259         }
260         else {
261             this->text.Append(L"o");
262         }
263         break;
264
265     case 0x50:
266         if (!(this->shft && caps) && (this->shft || caps)) {
267             this->text.Append(L"P");
268         }
269         else {
270             this->text.Append(L"p");
271         }
272         break;
273
274     case 0x51:
275         if (!(this->shft && caps) && (this->shft || caps)) {
276             this->text.Append(L"Q");
277         }
278         else {
279             this->text.Append(L"q");
280         }
```

```
281         break;
282
283     case 0x52:
284         if (!(this->shft && caps) && (this->shft || caps)) {
285             this->text.Append(L"R");
286         }
287         else {
288             this->text.Append(L"r");
289         }
290         break;
291
292     case 0x53:
293         if (!(this->shft && caps) && (this->shft || caps)) {
294             this->text.Append(L"S");
295         }
296         else {
297             this->text.Append(L"s");
298         }
299         break;
300
301     case 0x54:
302         if (!(this->shft && caps) && (this->shft || caps)) {
303             this->text.Append(L"T");
304         }
305         else {
306             this->text.Append(L"t");
307         }
308         break;
309
310     case 0x55:
311         if (!(this->shft && caps) && (this->shft || caps)) {
312             this->text.Append(L"U");
313         }
314         else {
315             this->text.Append(L"u");
316         }
317         break;
318
319     case 0x56:
320         if (!(this->shft && caps) && (this->shft || caps)) {
321             this->text.Append(L"V");
322         }
323         else {
324             this->text.Append(L"v");
325         }
326         break;
327
328     case 0x57:
329         if (!(this->shft && caps) && (this->shft || caps)) {
330             this->text.Append(L"W");
331         }
332         else {
333             this->text.Append(L"w");
334         }
335         break;
336
```

```
337     case 0x58:
338         if (!(this->shft && caps) && (this->shft || caps)) {
339             this->text.Append(L"X");
340         }
341         else {
342             this->text.Append(L"x");
343         }
344         break;
345
346     case 0x59:
347         if (!(this->shft && caps) && (this->shft || caps)) {
348             this->text.Append(L"Y");
349         }
350         else {
351             this->text.Append(L"y");
352         }
353         break;
354
355     case 0x5A:
356         if (!(this->shft && caps) && (this->shft || caps)) {
357             this->text.Append(L"Z");
358         }
359         else {
360             this->text.Append(L"z");
361         }
362         break;
363
364     case VK_MULTIPLY:
365         this->text.Append(L"*");
366         break;
367
368     case VK_ADD:
369         this->text.Append(L"+");
370         break;
371
372     case VK_SUBTRACT:
373         this->text.Append(L"-");
374         break;
375
376     case VK_DIVIDE:
377         this->text.Append(L"/");
378         break;
379
380     case VK_OEM_1:
381         if (this->shft) {
382             this->text.Append(L":");
383         }
384         else {
385             this->text.Append(L";");
386         }
387         break;
388
389     case VK_OEM_PLUS:
390         if (this->shft) {
391             this->text.Append(L="+");
392         }
```

```
393         else {
394             this->text.Append(L"=");
395         }
396         break;
397
398     case VK_OEM_COMMA:
399         if (this->shft) {
400             this->text.Append(L"<");
401         }
402         else {
403             this->text.Append(L",");
404         }
405         break;
406
407     case VK_OEM_MINUS:
408         if (this->shft) {
409             this->text.Append(L"_");
410         }
411         else {
412             this->text.Append(L"-");
413         }
414         break;
415
416     case VK_OEM_PERIOD:
417         if (this->shft) {
418             this->text.Append(L">");
419         }
420         else {
421             this->text.Append(L".");
422         }
423         break;
424
425     case VK_OEM_2:
426         if (this->shft) {
427             this->text.Append(L"?");
428         }
429         else {
430             this->text.Append(L"/");
431         }
432         break;
433
434     case VK_OEM_3:
435         if (this->shft) {
436             this->text.Append(L"@");
437         }
438         else {
439             this->text.Append(L"'");
440         }
441         break;
442
443     case VK_OEM_4:
444         if (this->shft) {
445             this->text.Append(L"{");
446         }
447         else {
448             this->text.Append(L"[");
```



```
449     }
450     break;
451
452     case VK_OEM_5:
453         if (this->shft) {
454             this->text.Append(L"|");
455         }
456         else {
457             this->text.Append(L"\\");
458         }
459         break;
460
461     case VK_OEM_6:
462         if (this->shft) {
463             this->text.Append(L"}");
464         }
465         else {
466             this->text.Append(L"]");
467         }
468         break;
469
470     default:
471         return FALSE;
472
473     }
474     return TRUE;
475 }
476 BOOL CTextHandler::OnKeyUp(UINT nChar, UINT nRepCnt, UINT nFlags)
477 {
478     if (nChar == VK_SHIFT) {
479         this->shft = FALSE;
480     }
481     return 0;
482 }
483
484
```