
Subject: I can't to build exe-file from default package.
Posted by [AZelenUppFm](#) on Thu, 24 May 2018 23:16:39 GMT
[View Forum Message](#) <> [Reply to Message](#)

I am tried to build the default GUI package. But i am get the error. How to build the exe-file correctly ? I am uploaded the screenshot.

File Attachments

1) [upp-err.jpg](#), downloaded 374 times

```
FwiExam - GUI - TheIDE - C:\Users\Алекс\Программы\Programming\ELP\upp\MyApps\FwiExam\FwiExam.cpp windows-1251 CRLF { MyA
File Edit Project Build Debug Assist Setup Help
FwiExam.cpp resource.h FwiExam.rc
1 #include <windows.h>
2 #include "resource.h"
3
4 HINSTANCE hInst;
5
6 LRESULT CALLBACK About(HWND hDlg, UINT message, WPARAM wParam)
7 {
8     switch(message)
9     {
10        case WM_INITDIALOG:
11            return TRUE;
12
13        case WM_COMMAND:
14            if(LOWORD(wParam) == IDOK || LOWORD(wParam) == IDCANCEL)
15            {
16                EndDialog(hDlg, LOWORD(wParam));
17                return TRUE;
18            }
19            break;
20    }
21    return FALSE;
22 }
23
24 LRESULT CALLBACK WndProc(HWND hWnd, UINT message, WPARAM wParam, LPARAM lParam)
25 {
26     int wmId, wmEvent;
27     PAINTSTRUCT ps;
28     HDC hdc;
29
30     const TCHAR szHello[] = "Hello World!";
31
32     switch(message)
33     {
34        case WM_INITDIALOG:
35            DialogBox(hInst, (HRSRC)FindResource(hInst, RT_DIALOG, MAKEINTRESOURCE(IDD_DIALOG1)), hWnd, (DLGPROC)About);
36            break;
37        case WM_DESTROY:
38            PostQuitMessage(0);
39            break;
40        default:
41            DefDlgProc(hWnd, message, wParam, lParam);
42    }
43    return 0;
44 }
45
46 int WINAPI WinMain(HINSTANCE hInst, HINSTANCE hPrevInst, LPSTR lpszCmdLine, int iCmdShow)
47 {
48     if(!InitInstance(hInst, lpszCmdLine))
49         return 0;
50
51     HINSTANCE hAppInst = hInst;
52     HWND hWnd = CreateDialog(hAppInst, (HRSRC)FindResource(hAppInst, RT_DIALOG, MAKEINTRESOURCE(IDD_DIALOG1)), hPrevInst, (DLGPROC)About);
53     ShowDialog(hWnd);
54     DestroyDialog(hWnd);
55
56     return 0;
57 }
58
59 int WINAPI InitInstance(HINSTANCE hInst, LPSTR lpszCmdLine)
60 {
61     if(!hInst)
62         return 0;
63
64     if(!hInst)
65         return 0;
66
67     if(!hInst)
68         return 0;
69
70     if(!hInst)
71         return 0;
72
73     if(!hInst)
74         return 0;
75
76     if(!hInst)
77         return 0;
78
79     if(!hInst)
80         return 0;
81
82     if(!hInst)
83         return 0;
84
85     if(!hInst)
86         return 0;
87
88     if(!hInst)
89         return 0;
90
91     if(!hInst)
92         return 0;
93
94     if(!hInst)
95         return 0;
96
97     if(!hInst)
98         return 0;
99
100    if(!hInst)
101        return 0;
102
103    if(!hInst)
104        return 0;
105
106    if(!hInst)
107        return 0;
108
109    if(!hInst)
110        return 0;
111
112    if(!hInst)
113        return 0;
114
115    if(!hInst)
116        return 0;
117
118    if(!hInst)
119        return 0;
120
121    if(!hInst)
122        return 0;
123
124    if(!hInst)
125        return 0;
126
127    if(!hInst)
128        return 0;
129
130    if(!hInst)
131        return 0;
132
133    if(!hInst)
134        return 0;
135
136    if(!hInst)
137        return 0;
138
139    if(!hInst)
140        return 0;
141
142    if(!hInst)
143        return 0;
144
145    if(!hInst)
146        return 0;
147
148    if(!hInst)
149        return 0;
150
151    if(!hInst)
152        return 0;
153
154    if(!hInst)
155        return 0;
156
157    if(!hInst)
158        return 0;
159
160    if(!hInst)
161        return 0;
162
163    if(!hInst)
164        return 0;
165
166    if(!hInst)
167        return 0;
168
169    if(!hInst)
170        return 0;
171
172    if(!hInst)
173        return 0;
174
175    if(!hInst)
176        return 0;
177
178    if(!hInst)
179        return 0;
180
181    if(!hInst)
182        return 0;
183
184    if(!hInst)
185        return 0;
186
187    if(!hInst)
188        return 0;
189
190    if(!hInst)
191        return 0;
192
193    if(!hInst)
194        return 0;
195
196    if(!hInst)
197        return 0;
198
199    if(!hInst)
200        return 0;
201
202    if(!hInst)
203        return 0;
204
205    if(!hInst)
206        return 0;
207
208    if(!hInst)
209        return 0;
210
211    if(!hInst)
212        return 0;
213
214    if(!hInst)
215        return 0;
216
217    if(!hInst)
218        return 0;
219
220    if(!hInst)
221        return 0;
222
223    if(!hInst)
224        return 0;
225
226    if(!hInst)
227        return 0;
228
229    if(!hInst)
230        return 0;
231
232    if(!hInst)
233        return 0;
234
235    if(!hInst)
236        return 0;
237
238    if(!hInst)
239        return 0;
240
241    if(!hInst)
242        return 0;
243
244    if(!hInst)
245        return 0;
246
247    if(!hInst)
248        return 0;
249
250    if(!hInst)
251        return 0;
252
253    if(!hInst)
254        return 0;
255
256    if(!hInst)
257        return 0;
258
259    if(!hInst)
260        return 0;
261
262    if(!hInst)
263        return 0;
264
265    if(!hInst)
266        return 0;
267
268    if(!hInst)
269        return 0;
270
271    if(!hInst)
272        return 0;
273
274    if(!hInst)
275        return 0;
276
277    if(!hInst)
278        return 0;
279
280    if(!hInst)
281        return 0;
282
283    if(!hInst)
284        return 0;
285
286    if(!hInst)
287        return 0;
288
289    if(!hInst)
290        return 0;
291
292    if(!hInst)
293        return 0;
294
295    if(!hInst)
296        return 0;
297
298    if(!hInst)
299        return 0;
300
301    if(!hInst)
302        return 0;
303
304    if(!hInst)
305        return 0;
306
307    if(!hInst)
308        return 0;
309
310    if(!hInst)
311        return 0;
312
313    if(!hInst)
314        return 0;
315
316    if(!hInst)
317        return 0;
318
319    if(!hInst)
320        return 0;
321
322    if(!hInst)
323        return 0;
324
325    if(!hInst)
326        return 0;
327
328    if(!hInst)
329        return 0;
330
331    if(!hInst)
332        return 0;
333
334    if(!hInst)
335        return 0;
336
337    if(!hInst)
338        return 0;
339
340    if(!hInst)
341        return 0;
342
343    if(!hInst)
344        return 0;
345
346    if(!hInst)
347        return 0;
348
349    if(!hInst)
350        return 0;
351
352    if(!hInst)
353        return 0;
354
355    if(!hInst)
356        return 0;
357
358    if(!hInst)
359        return 0;
360
361    if(!hInst)
362        return 0;
363
364    if(!hInst)
365        return 0;
366
367    if(!hInst)
368        return 0;
369
370    if(!hInst)
371        return 0;
372
373    if(!hInst)
374        return 0;
375
376    if(!hInst)
377        return 0;
378
379    if(!hInst)
380        return 0;
381
382    if(!hInst)
383        return 0;
384
385    if(!hInst)
386        return 0;
387
388    if(!hInst)
389        return 0;
390
391    if(!hInst)
392        return 0;
393
394    if(!hInst)
395        return 0;
396
397    if(!hInst)
398        return 0;
399
400    if(!hInst)
401        return 0;
402
403    if(!hInst)
404        return 0;
405
406    if(!hInst)
407        return 0;
408
409    if(!hInst)
410        return 0;
411
412    if(!hInst)
413        return 0;
414
415    if(!hInst)
416        return 0;
417
418    if(!hInst)
419        return 0;
420
421    if(!hInst)
422        return 0;
423
424    if(!hInst)
425        return 0;
426
427    if(!hInst)
428        return 0;
429
430    if(!hInst)
431        return 0;
432
433    if(!hInst)
434        return 0;
435
436    if(!hInst)
437        return 0;
438
439    if(!hInst)
440        return 0;
441
442    if(!hInst)
443        return 0;
444
445    if(!hInst)
446        return 0;
447
448    if(!hInst)
449        return 0;
450
451    if(!hInst)
452        return 0;
453
454    if(!hInst)
455        return 0;
456
457    if(!hInst)
458        return 0;
459
460    if(!hInst)
461        return 0;
462
463    if(!hInst)
464        return 0;
465
466    if(!hInst)
467        return 0;
468
469    if(!hInst)
470        return 0;
471
472    if(!hInst)
473        return 0;
474
475    if(!hInst)
476        return 0;
477
478    if(!hInst)
479        return 0;
480
481    if(!hInst)
482        return 0;
483
484    if(!hInst)
485        return 0;
486
487    if(!hInst)
488        return 0;
489
490    if(!hInst)
491        return 0;
492
493    if(!hInst)
494        return 0;
495
496    if(!hInst)
497        return 0;
498
499    if(!hInst)
500        return 0;
501
502    if(!hInst)
503        return 0;
504
505    if(!hInst)
506        return 0;
507
508    if(!hInst)
509        return 0;
510
511    if(!hInst)
512        return 0;
513
514    if(!hInst)
515        return 0;
516
517    if(!hInst)
518        return 0;
519
520    if(!hInst)
521        return 0;
522
523    if(!hInst)
524        return 0;
525
526    if(!hInst)
527        return 0;
528
529    if(!hInst)
530        return 0;
531
532    if(!hInst)
533        return 0;
534
535    if(!hInst)
536        return 0;
537
538    if(!hInst)
539        return 0;
540
541    if(!hInst)
542        return 0;
543
544    if(!hInst)
545        return 0;
546
547    if(!hInst)
548        return 0;
549
550    if(!hInst)
551        return 0;
552
553    if(!hInst)
554        return 0;
555
556    if(!hInst)
557        return 0;
558
559    if(!hInst)
560        return 0;
561
562    if(!hInst)
563        return 0;
564
565    if(!hInst)
566        return 0;
567
568    if(!hInst)
569        return 0;
570
571    if(!hInst)
572        return 0;
573
574    if(!hInst)
575        return 0;
576
577    if(!hInst)
578        return 0;
579
580    if(!hInst)
581        return 0;
582
583    if(!hInst)
584        return 0;
585
586    if(!hInst)
587        return 0;
588
589    if(!hInst)
590        return 0;
591
592    if(!hInst)
593        return 0;
594
595    if(!hInst)
596        return 0;
597
598    if(!hInst)
599        return 0;
600
601    if(!hInst)
602        return 0;
603
604    if(!hInst)
605        return 0;
606
607    if(!hInst)
608        return 0;
609
610    if(!hInst)
611        return 0;
612
613    if(!hInst)
614        return 0;
615
616    if(!hInst)
617        return 0;
618
619    if(!hInst)
620        return 0;
621
622    if(!hInst)
623        return 0;
624
625    if(!hInst)
626        return 0;
627
628    if(!hInst)
629        return 0;
630
631    if(!hInst)
632        return 0;
633
634    if(!hInst)
635        return 0;
636
637    if(!hInst)
638        return 0;
639
640    if(!hInst)
641        return 0;
642
643    if(!hInst)
644        return 0;
645
646    if(!hInst)
647        return 0;
648
649    if(!hInst)
650        return 0;
651
652    if(!hInst)
653        return 0;
654
655    if(!hInst)
656        return 0;
657
658    if(!hInst)
659        return 0;
660
661    if(!hInst)
662        return 0;
663
664    if(!hInst)
665        return 0;
666
667    if(!hInst)
668        return 0;
669
670    if(!hInst)
671        return 0;
672
673    if(!hInst)
674        return 0;
675
676    if(!hInst)
677        return 0;
678
679    if(!hInst)
680        return 0;
681
682    if(!hInst)
683        return 0;
684
685    if(!hInst)
686        return 0;
687
688    if(!hInst)
689        return 0;
690
691    if(!hInst)
692        return 0;
693
694    if(!hInst)
695        return 0;
696
697    if(!hInst)
698        return 0;
699
700    if(!hInst)
701        return 0;
702
703    if(!hInst)
704        return 0;
705
706    if(!hInst)
707        return 0;
708
709    if(!hInst)
710        return 0;
711
712    if(!hInst)
713        return 0;
714
715    if(!hInst)
716        return 0;
717
718    if(!hInst)
719        return 0;
720
721    if(!hInst)
722        return 0;
723
724    if(!hInst)
725        return 0;
726
727    if(!hInst)
728        return 0;
729
730    if(!hInst)
731        return 0;
732
733    if(!hInst)
734        return 0;
735
736    if(!hInst)
737        return 0;
738
739    if(!hInst)
740        return 0;
741
742    if(!hInst)
743        return 0;
744
745    if(!hInst)
746        return 0;
747
748    if(!hInst)
749        return 0;
750
751    if(!hInst)
752        return 0;
753
754    if(!hInst)
755        return 0;
756
757    if(!hInst)
758        return 0;
759
760    if(!hInst)
761        return 0;
762
763    if(!hInst)
764        return 0;
765
766    if(!hInst)
767        return 0;
768
769    if(!hInst)
770        return 0;
771
772    if(!hInst)
773        return 0;
774
775    if(!hInst)
776        return 0;
777
778    if(!hInst)
779        return 0;
780
781    if(!hInst)
782        return 0;
783
784    if(!hInst)
785        return 0;
786
787    if(!hInst)
788        return 0;
789
790    if(!hInst)
791        return 0;
792
793    if(!hInst)
794        return 0;
795
796    if(!hInst)
797        return 0;
798
799    if(!hInst)
800        return 0;
801
802    if(!hInst)
803        return 0;
804
805    if(!hInst)
806        return 0;
807
808    if(!hInst)
809        return 0;
810
811    if(!hInst)
812        return 0;
813
814    if(!hInst)
815        return 0;
816
817    if(!hInst)
818        return 0;
819
820    if(!hInst)
821        return 0;
822
823    if(!hInst)
824        return 0;
825
826    if(!hInst)
827        return 0;
828
829    if(!hInst)
830        return 0;
831
832    if(!hInst)
833        return 0;
834
835    if(!hInst)
836        return 0;
837
838    if(!hInst)
839        return 0;
840
841    if(!hInst)
842        return 0;
843
844    if(!hInst)
845        return 0;
846
847    if(!hInst)
848        return 0;
849
850    if(!hInst)
851        return 0;
852
853    if(!hInst)
854        return 0;
855
856    if(!hInst)
857        return 0;
858
859    if(!hInst)
860        return 0;
861
862    if(!hInst)
863        return 0;
864
865    if(!hInst)
866        return 0;
867
868    if(!hInst)
869        return 0;
870
871    if(!hInst)
872        return 0;
873
874    if(!hInst)
875        return 0;
876
877    if(!hInst)
878        return 0;
879
880    if(!hInst)
881        return 0;
882
883    if(!hInst)
884        return 0;
885
886    if(!hInst)
887        return 0;
888
889    if(!hInst)
890        return 0;
891
892    if(!hInst)
893        return 0;
894
895    if(!hInst)
896        return 0;
897
898    if(!hInst)
899        return 0;
900
901    if(!hInst)
902        return 0;
903
904    if(!hInst)
905        return 0;
906
907    if(!hInst)
908        return 0;
909
910    if(!hInst)
911        return 0;
912
913    if(!hInst)
914        return 0;
915
916    if(!hInst)
917        return 0;
918
919    if(!hInst)
920        return 0;
921
922    if(!hInst)
923        return 0;
924
925    if(!hInst)
926        return 0;
927
928    if(!hInst)
929        return 0;
930
931    if(!hInst)
932        return 0;
933
934    if(!hInst)
935        return 0;
936
937    if(!hInst)
938        return 0;
939
940    if(!hInst)
941        return 0;
942
943    if(!hInst)
944        return 0;
945
946    if(!hInst)
947        return 0;
948
949    if(!hInst)
950        return 0;
951
952    if(!hInst)
953        return 0;
954
955    if(!hInst)
956        return 0;
957
958    if(!hInst)
959        return 0;
960
961    if(!hInst)
962        return 0;
963
964    if(!hInst)
965        return 0;
966
967    if(!hInst)
968        return 0;
969
970    if(!hInst)
971        return 0;
972
973    if(!hInst)
974        return 0;
975
976    if(!hInst)
977        return 0;
978
979    if(!hInst)
980        return 0;
981
982    if(!hInst)
983        return 0;
984
985    if(!hInst)
986        return 0;
987
988    if(!hInst)
989        return 0;
990
991    if(!hInst)
992        return 0;
993
994    if(!hInst)
995        return 0;
996
997    if(!hInst)
998        return 0;
999
1000    if(!hInst)
1001        return 0;
1002
1003    if(!hInst)
1004        return 0;
1005
1006    if(!hInst)
1007        return 0;
1008
1009    if(!hInst)
1010        return 0;
1011
1012    if(!hInst)
1013        return 0;
1014
1015    if(!hInst)
1016        return 0;
1017
1018    if(!hInst)
1019        return 0;
1020
1021    if(!hInst)
1022        return 0;
1023
1024    if(!hInst)
1025        return 0;
1026
1027    if(!hInst)
1028        return 0;
1029
1030    if(!hInst)
1031        return 0;
1032
1033    if(!hInst)
1034        return 0;
1035
1036    if(!hInst)
1037        return 0;
1038
1039    if(!hInst)
1040        return 0;
1041
1042    if(!hInst)
1043        return 0;
1044
1045    if(!hInst)
1046        return 0;
1047
1048    if(!hInst)
1049        return 0;
1050
1051    if(!hInst)
1052        return 0;
1053
1054    if(!hInst)
1055        return 0;
1056
1057    if(!hInst)
1058        return 0;
1059
1060    if(!hInst)
1061        return 0;
1062
1063    if(!hInst)
1064        return 0;
1065
1066    if(!hInst)
1067        return 0;
1068
1069    if(!hInst)
1070        return 0;
1071
1072    if(!hInst)
1073        return 0;
1074
1075    if(!hInst)
1076        return 0;
1077
1078    if(!hInst)
1079        return 0;
1080
1081    if(!hInst)
1082        return 0;
1083
1084    if(!hInst)
1085        return 0;
1086
1087    if(!hInst)
1088        return 0;
1089
1090    if(!hInst)
1091        return 0;
1092
1093    if(!hInst)
1094        return 0;
1095
1096    if(!hInst)
1097        return 0;
1098
1099    if(!hInst)
1100        return 0;
1101
1102    if(!hInst)
1103        return 0;
1104
1105    if(!hInst)
1106        return 0;
1107
1108    if(!hInst)
1109        return 0;
1110
1111    if(!hInst)
1112        return 0;
1113
1114    if(!hInst)
1115        return 0;
1116
1117    if(!hInst)
1118        return 0;
1119
1120    if(!hInst)
1121        return 0;
1122
1123    if(!hInst)
1124        return 0;
1125
1126    if(!hInst)
1127        return 0;
1128
1129    if(!hInst)
1130        return 0;
1131
1132    if(!hInst)
1133        return 0;
1134
1135    if(!hInst)
1136        return 0;
1137
1138    if(!hInst)
1139        return 0;
1140
1141    if(!hInst)
1142        return 0;
1143
1144    if(!hInst)
1145        return 0;
1146
1147    if(!hInst)
1148        return 0;
1149
1150    if(!hInst)
1151        return 0;
1152
1153    if(!hInst)
1154        return 0;
1155
1156    if(!hInst)
1157        return 0;
1158
1159    if(!hInst)
1160        return 0;
1161
1162    if(!hInst)
1163        return 0;
1164
1165    if(!hInst)
1166        return 0;
1167
1168    if(!hInst)
1169        return 0;
1170
1171    if(!hInst)
1172        return 0;
1173
1174    if(!hInst)
1175        return 0;
1176
1177    if(!hInst)
1178        return 0;
1179
1180    if(!hInst)
1181        return 0;
1182
1183    if(!hInst)
1184        return 0;
1185
1186    if(!hInst)
1187        return 0;
1188
1189    if(!hInst)
1190        return 0;
1191
1192    if(!hInst)
1193        return 0;
1194
1195    if(!hInst)
1196        return 0;
1197
1198    if(!hInst)
1199        return 0;
1200
1201    if(!hInst)
1202        return 0;
1203
1204    if(!hInst)
1205        return 0;
1206
1207    if(!hInst)
1208        return 0;
1209
1210    if(!hInst)
1211        return 0;
1212
1213    if(!hInst)
1214        return 0;
1215
1216    if(!hInst)
1217        return 0;
1218
1219    if(!hInst)
1220        return 0;
1221
1222    if(!hInst)
1223        return 0;
1224
1225    if(!hInst)
1226        return 0;
1227
1228    if(!hInst)
1229        return 0;
1230
1231    if(!hInst)
1232        return 0;
1233
1234    if(!hInst)
1235        return 0;
1236
1237    if(!hInst)
1238        return 0;
1239
1240    if(!hInst)
1241        return 0;
1242
1243    if(!hInst)
1244        return 0;
1245
1246    if(!hInst)
1247        return 0;
1248
1249    if(!hInst)
1250        return 0;
1251
1252    if(!hInst)
1253        return 0;
1254
1255    if(!hInst)
1256        return 0;
1257
1258    if(!hInst)
1259        return 0;
1260
1261    if(!hInst)
1262        return 0;
1263
1264    if(!hInst)
1265        return 0;
1266
1267    if(!hInst)
1268        return 0;
1269
1270    if(!hInst)
1271        return 0;
1272
1273    if(!hInst)
1274        return 0;
1275
1276    if(!hInst)
1277        return 0;
1278
1279    if(!hInst)
1280        return 0;
1281
1282    if(!hInst)
1283        return 0;
1284
1285    if(!hInst)
1286        return 0;
1287
1288    if(!hInst)
1289        return 0;
1290
1291    if(!hInst)
1292        return 0;
1293
1294    if(!hInst)
1295        return 0;
1296
1297    if(!hInst)
1298        return 0;
1299
1300    if(!hInst)
1301        return 0;
1302
1303    if(!hInst)
1304        return 0;
1305
1306    if(!hInst)
1307        return 0;
1308
1309    if(!hInst)
1310        return 0;
1311
1312    if(!hInst)
1313        return 0;
1314
1315    if(!hInst)
1316        return 0;
1317
1318    if(!hInst)
1319        return 0;
1320
1321    if(!hInst)
1322        return 0;
1323
1324    if(!hInst)
1325        return 0;
1326
1327    if(!hInst)
1328        return 0;
1329
1330    if(!hInst)
1331        return 0;
1332
1333    if(!hInst)
1334        return 0;
1335
1336    if(!hInst)
1337        return 0;
1338
1339    if(!hInst)
1340        return 0;
1341
1342    if(!hInst)
1343        return 0;
1344
1345    if(!hInst)
1346        return 0;
1347
1348    if(!hInst)
1349        return 0;
1350
1351    if(!hInst)
1352        return 0;
1353
1354    if(!hInst)
1355        return 0;
1356
1357    if(!hInst)
1358        return 0;
1359
1360    if(!hInst)
1361        return 0;
1362
1363    if(!hInst)
1364        return 0;
1365
1366    if(!hInst)
1367        return 0;
1368
1369    if(!hInst)
1370        return 0;
1371
1372    if(!hInst)
1373        return 0;
1374
1375    if(!hInst)
1376        return 0;
1377
1378    if(!hInst)
1379        return 0;
1380
1381    if(!hInst)
1382        return 0;
1383
1384    if(!hInst)
1385        return 0;
1386
1387    if(!hInst)
1388        return 0;
1389
1390    if(!hInst)
1391        return 0;
1392
1393    if(!hInst)
1394        return 0;
1395
1396    if(!hInst)
1397        return 0;
1398
1399    if(!hInst)
1400        return 0;
1401
1402    if(!hInst)
1403        return 0;
1404
1405    if(!hInst)
1406        return 0;
1407
1408    if(!hInst)
1409        return 0;
1410
1411    if(!hInst)
1412        return 0;
1413
1414    if(!hInst)
1415        return 0;
1416
1417    if(!hInst)
1418        return 0;
1419
1420    if(!hInst)
1421        return 0;
1422
1423    if(!hInst)
1424        return 0;
1425
1426    if(!hInst)
1427        return 0;
1428
1429    if(!hInst)
1430        return 0;
1431
1432    if(!hInst)
1433        return 0;
1434
1435    if(!hInst)
1436        return 0;
1437
1438    if(!hInst)
1439        return 0;
1440
1441    if(!hInst)
1442        return 0;
1443
1444    if(!hInst)
1445        return 0;
1446
1447    if(!hInst)
1448        return 0;
1449
1450    if(!hInst)
1451        return 0;
1452
1453    if(!hInst)
1454        return 0;
1455
1456    if(!hInst)
1457        return 0;
1458
1459    if(!hInst)
1460        return 0;
1461
1462    if(!hInst)
1463        return 0;
1464
1465    if(!hInst)
1466        return 0;
1467
1468    if(!hInst)
1469        return 0;
1470
1471    if(!hInst)
1472        return 0;
1473
1474    if(!hInst)
1475        return 0;
1476
1477    if(!hInst)
1478        return 0;
1479
1480    if(!hInst)
1481        return 0;
1482
1483    if(!hInst)
1484        return 0;
1485
1486    if(!hInst)
1487        return 0;
1488
1489    if(!hInst)
1490        return 0;
1491
1492    if(!hInst)
1493        return 0;
1494
1495    if(!hInst)
1496        return 0;
1497
1498    if(!hInst)
1499        return 0;
1500
1501    if(!hInst)
1502        return 0;
1503
1504    if(!hInst)
1505        return 0;
1506
1507    if(!hInst)
1508        return 0;
1509
1510    if(!hInst)
1511        return 0;
1512
1513    if(!hInst)
1514        return 0;
1515
1516    if(!hInst)
1517        return 0;
1518
1519    if(!hInst)
1520        return 0;
1521
1522    if(!hInst)
1523        return 0;
1524
1525    if(!hInst)
1526        return 0;
1527
```

Hello,

If you are new in Upp world. I highly suggest to you, to firstly build any example attached to Upp standard distribution. Then if everything works as expected, you should try creating your own package with creator.

I also see that you start using WinAPI. Upp provides portable GUI library, so there is not need to use native API.

Please select following package - if you can not find it for some reasons please let me know:

Sincerely,
Klugier

File Attachments

1) [ExemplaryGUIPackage.PNG](#), downloaded 550 times

- Assembly
- examples
- examples-bazaar
- MyApps
- MyApps-bazaar
- reference
- tutorial
- uppsrc

Package (60)	Nest	Description
CoreTutorial	tutorial	
Draw01	tutorial	Basic drawing operations
Draw02	tutorial	Offsets and clipping
Draw03	tutorial	Fonts and font metrics
Draw04	tutorial	DrawingDraw
Draw05	tutorial	ImageDraw
Draw06	tutorial	Printing
Grid01	tutorial	Basics
Gui01	tutorial	GUI application main function
Gui02	tutorial	Application window
Gui03	tutorial	Modifying TopWindow properties
Gui04	tutorial	Painting the view area
Gui05	tutorial	Reacting to input events
Gui06	tutorial	Multiple main windows
Gui07	tutorial	Menu
Gui07a	tutorial	Menu - as single lambda
Gui08	tutorial	Context menu
Gui09	tutorial	Context menu alternative and using CallbackArgTarget
Gui10	tutorial	Adding images
Gui11	tutorial	ToolBar
Gui12	tutorial	StatusBar and decorative Frames
Gui13	tutorial	Adding child Ctrls - widgets, reacting to widget events
Gui14	tutorial	More about logical coordinates
Gui15	tutorial	Font-zoomed logical coordinates
Gui16	tutorial	Layouts

Application window

Main packages of first nest ▾

 Brief list