

JAVA.UTIL.LOCALE.GETISOCOUNTRIES METHOD

http://www.tutorialspoint.com/java/util/locale_getisocountries.htm

Copyright © tutorialspoint.com

Description

The **java.util.Locale.getISOCountries** method returns a list of all 2-letter country codes defined in ISO 3166. This can be used to create Locales.

Declaration

Following is the declaration for **java.util.Locale.getISOCountries** method

```
public static String[] getISOCountries()
```

Parameters

- NA

Return Value

This method does not return a value.

Exception

- NA

Example

The following example shows the usage of java.util.Locale.getISOCountries method.

```
package com.tutorialspoint;

import java.util.*;

public class LocaleDemo {

    public static void main(String[] args) {

        // create a new locale
        Locale locale = new Locale("en", "US", "WIN");

        // print locale
        System.out.println("Locale:" + locale);

        // get ISO countries
        String[] countries = Locale.getISOCountries();

        // print countries
        System.out.println("Countries are:");
        for (int i = 0; i < countries.length; i++) {
            System.out.println(i + ":" + countries[i]);
        }
    }
}
```

Let us compile and run the above program, this will produce the following result:

```
Locale:en_US_WIN
Countries are:
0:AD
1:AE
2:AF
3:AG
```

4:AI
5:AL
6:AM
7:AN
8:AO
9:AQ
10:AR
11:AS
12:AT
13:AU
14:AW
15:AX
16:AZ
17:BA
18:BB
19:BD
20:BE
21:BF
22:BG
23:BH
24:BI
25:BJ
26:BL
27:BM
28:BN
29:BO
30:BR
31:BS
32:BT
33:BV
34:BW
35:BY
36:BZ
37:CA
38:CC
39:CD
40:CF
41:CG
42:CH
43:CI
44:CK
45:CL
46:CM
47:CN
48:CO
49:CR
50:CS
51:CU
52:CV
53:CX
54:CY
55:CZ
56:DE
57:DJ
58:DK
59:DM
60:DO
61:DZ
62:EC
63:EE
64:EG
65:EH
66:ER
67:ES
68:ET
69:FI
70:FJ
71:FK
72:FM

73:FO
74:FR
75:GA
76:GB
77:GD
78:GE
79:GF
80:GG
81:GH
82:GI
83:GL
84:GM
85:GN
86:GP
87:GQ
88:GR
89:GS
90:GT
91:GU
92:GW
93:GY
94:HK
95:HM
96:HN
97:HR
98:HT
99:HU
100:ID
101:IE
102:IL
103:IM
104:IN
105:IO
106:IQ
107:IR
108:IS
109:IT
110:JE
111:JM
112:JO
113:JP
114:KE
115:KG
116:KH
117:KI
118:KM
119:KN
120:KP
121:KR
122:KW
123:KY
124:KZ
125:LA
126:LB
127:LC
128:LI
129:LK
130:LR
131:LS
132:LT
133:LU
134:LV
135:LY
136:MA
137:MC
138:MD
139:ME
140:MF
141:MG

142:MH
143:MK
144:ML
145:MM
146:MN
147:MO
148:MP
149:MQ
150:MR
151:MS
152:MT
153:MU
154:MV
155:MW
156:MX
157:MY
158:MZ
159:NA
160:NC
161:NE
162:NF
163:NG
164:NI
165:NL
166:NO
167:NP
168:NR
169:NU
170:NZ
171:OM
172:PA
173:PE
174:PF
175:PG
176:PH
177:PK
178:PL
179:PM
180:PN
181:PR
182:PS
183:PT
184:PW
185:PY
186:QA
187:RE
188:RO
189:RS
190:RU
191:RW
192:SA
193:SB
194:SC
195:SD
196:SE
197:SG
198:SH
199:SI
200:SJ
201:SK
202:SL
203:SM
204:SN
205:SO
206:SR
207:ST
208:SV
209:SY
210:SZ

211:TC
212:TD
213:TF
214:TG
215:TH
216:TJ
217:TK
218:TL
219:TM
220:TN
221:TO
222:TR
223:TT
224:TV
225:TW
226:TZ
227:UA
228:UG
229:UM
230:US
231:UY
232:UZ
233:VA
234:VC
235:VE
236:VG
237:VI
238:VN
239:VU
240:WF
241:WS
242:YE
243:YT
244:ZA
245:ZM
246:ZW

Loading [MathJax]/jax/output/HTML-CSS/fonts/TeX/fontdata.js