

```
1 #!/usr/bin/python -tt
2 # Exercícios by Nick Parlante (CodingBat)
3
4 # A. multstring
5 # seja uma string s e um inteiro positivo n
6 # retorna uma string com n cópias da string original
7 # multstring('Hi', 2) -> 'HiHi'
8 def multstring(s, n):
9     return
10
11 # B. string_splosion
12 # string_splosion('Code') -> 'CCoCodCode'
13 # string_splosion('abc') -> 'aababc'
14 # string_splosion('ab') -> 'aab'
15 def string_splosion(s):
16     return
17
18 # C. array_count9
19 # conta quantas vezes aparece o 9 numa lista nums
20 def array_count9(nums):
21     return
22
23 # D. array_front9
24 # verifica se pelo menos um dos quatro primeiros é nove
25 # array_front9([1, 2, 9, 3, 4]) -> True
26 # array_front9([1, 2, 3, 4, 9]) -> False
27 # array_front9([1, 2, 3, 4, 5]) -> False
28 def array_front9(nums):
29     return
30
31 # E. hello_name
32 # seja uma string name
33 # hello_name('Bob') -> 'Hello Bob!'
34 # hello_name('Alice') -> 'Hello Alice!'
35 # hello_name('X') -> 'Hello X!'
36 def hello_name(name):
37     return
38
39 # F. make_tags
40 # make_tags('i', 'Yay'), '<i>Yay</i>'
41 # make_tags('i', 'Hello'), '<i>Hello</i>'
42 # make_tags('cite', 'Yay'), '<cite>Yay</cite>'
43 def make_tags(tab, word):
44     return
45
46 # G. extra_end
47 # seja um string s com no mínimo duas letras
48 # retorna três vezes as duas últimas letras
49 # extra_end('Hello'), 'lololo'
50 # extra_end('ab'), 'ababab'
51 # extra_end('Hi'), 'HiHiHi'
52 def extra_end(s):
53     return
54
55 # H. first_half
56 # seja uma string s
57 # retorna a primeira metade da string
58 # first_half('WooHoo') -> 'Woo'
59 # first_half('HelloThere') -> 'Hello'
60 # first_half('abcdef') -> 'abc'
61 def first_half(s):
62     return
63
64 # I. sem_pontas
65 # seja uma string s de pelo menos dois caracteres
66 # retorna uma string sem o primeiro e último caracter
67 # without_end('Hello') -> 'ell'
68 # without_end('python') -> 'ytho'
69 # without_end('coding') -> 'odin'
```

```

70 def sem_pontas(s):
71     return
72
73 # J. roda2
74 # rodar uma string s duas posições
75 # a string possui pelo menos 2 caracteres
76 # left2('Hello') -> 'lloHe'
77 # left2('Hi') -> 'Hi'
78 def roda2(s):
79     return
80
81
82 # Provided simple test() function used in main() to print
83 # what each function returns vs. what it's supposed to return.
84 def test(obtido, esperado):
85     if obtido == esperado:
86         prefixo = ' Parabéns!'
87     else:
88         prefixo = ' Ainda não'
89     print ('%s obtido: %s esperado: %s'
90           % (prefixo, repr(obtido), repr(esperado)))
91
92 def main():
93     print ('Multstring')
94     test(multstring('Hi', 2), 'HiHi')
95     test(multstring('Hi', 3), 'HiHiHi')
96     test(multstring('Hi', 1), 'Hi')
97     test(multstring('Hi', 0), '')
98     test(multstring('Hi', 5), 'HiHiHiHiHi')
99     test(multstring('Oh Boy!', 2), 'Oh Boy!Oh Boy!')
100    test(multstring('x', 4), 'xxxx')
101    test(multstring('', 4), '')
102    test(multstring('code', 2), 'codecode')
103    test(multstring('code', 3), 'codecodecode')
104
105    print ()
106    print ('String Explosion')
107    test(string_splosion('Code'), 'CCoCodCode')
108    test(string_splosion('abc'), 'aababc')
109    test(string_splosion('ab'), 'aab')
110    test(string_splosion('x'), 'x')
111    test(string_splosion('fade'), 'ffafadfade')
112    test(string_splosion('There'), 'TThTheTherThere')
113    test(string_splosion('Kitten'), 'KKiKitKittKitteKitten')
114    test(string_splosion('Bye'), 'BByBye')
115    test(string_splosion('Good'), 'GGoGooGood')
116    test(string_splosion('Bad'), 'BBaBad')
117
118    print ()
119    print ('Array count 9')
120    test(array_count9([1, 99, 9]), 1)
121    test(array_count9([1, 9, 9]), 2)
122    test(array_count9([1, 9, 9, 3, 9]), 3)
123    test(array_count9([1, 2, 3]), 0)
124    test(array_count9([]), 0)
125    test(array_count9([4, 2, 4, 3, 1]), 0)
126    test(array_count9([9, 2, 99, 3, 1]), 1)
127
128    print ()
129    print ('Array front 9')
130    test(array_front9([1, 2, 9, 3, 4]), True)
131    test(array_front9([1, 2, 3, 4, 9]), False)
132    test(array_front9([1, 2, 3, 4, 5]), False)
133    test(array_front9([9, 2, 3]), True)
134    test(array_front9([1, 9, 9]), True)
135    test(array_front9([1, 2, 3]), False)
136    test(array_front9([1, 9]), True)
137    test(array_front9([5, 5]), False)
138    test(array_front9([2]), False)

```

```
139 test(array_front9([9]), True)
140 test(array_front9([]), False)
141 test(array_front9([3, 9, 2, 3, 3]), True)
142
143 print ()
144 print ('Hello Name')
145 test(hello_name('Bob'), 'Hello Bob!')
146 test(hello_name('Alice'), 'Hello Alice!')
147 test(hello_name('X'), 'Hello X!')
148 test(hello_name('Hello'), 'Hello Hello!')
149
150 print ()
151 print ('Make Tags')
152 test(make_tags('i', 'Yay'), '<i>Yay</i>')
153 test(make_tags('i', 'Hello'), '<i>Hello</i>')
154 test(make_tags('cite', 'Yay'), '<cite>Yay</cite>')
155 test(make_tags('address', 'here'), '<address>here</address>')
156 test(make_tags('body', 'Heart'), '<body>Heart</body>')
157 test(make_tags('i', 'i'), '<i>i</i>')
158 test(make_tags('i', ''), '<i></i>')
159
160 print ()
161 print ('Extra End')
162 test(extra_end('Hello'), 'lololo')
163 test(extra_end('ab'), 'ababab')
164 test(extra_end('Hi'), 'HiHiHi')
165 test(extra_end('Candy'), 'dydydy')
166 test(extra_end('Code'), 'dedede')
167
168 print ()
169 print ('First Half')
170 test(first_half('WooHoo'), 'Woo')
171 test(first_half('HelloThere'), 'Hello')
172 test(first_half('abcdef'), 'abc')
173 test(first_half('ab'), 'a')
174 test(first_half(''), '')
175 test(first_half('0123456789'), '01234')
176 test(first_half('kitten'), 'kit')
177
178 print ()
179 print ('Sem Pontas')
180 test(sem_pontas('Hello'), 'ell')
181 test(sem_pontas('Python'), 'ytho')
182 test(sem_pontas('coding'), 'odin')
183 test(sem_pontas('code'), 'od')
184 test(sem_pontas('ab'), '')
185 test(sem_pontas('Chocolate!'), 'hocolate')
186 test(sem_pontas('kitten'), 'itte')
187 test(sem_pontas('woohoo'), 'ooho')
188
189 print ()
190 print ('Roda 2')
191 test(roda2('Hello'), 'lloHe')
192 test(roda2('python'), 'thonpy')
193 test(roda2('Hi'), 'Hi')
194 test(roda2('code'), 'deco')
195 test(roda2('cat'), 'tca')
196 test(roda2('12345'), '34512')
197 test(roda2('Chocolate'), 'ocolateCh')
198 test(roda2('bricks'), 'icksbr')
199
200 if __name__ == '__main__':
201     main()
```