codingbat.com — CodingBat Java String-2

CodingBat code practice

about | help | code help+videos | done | report | prefs | hakloi @mail.ru [log out]

Java | Python

String-2 ★★★☆☆☆☆ chance

Medium String problems -- 1 loop. See the Java String Help document for help with strings.

✓ doubleChar H	✓ countHi H	✓ catDog
✓ countCode	✓ endOther	✓ xyzThere
✓ bobThere	✓ xyBalance	✓ mixString
✓ repeatEnd	✓ repeatFront	✓ repeatSeparator
✓ prefixAgain	✓ xyzMiddle	✓ getSandwich
✓ sameStarChar	✓ oneTwo	✓ zipZap
✓ starOut	✓ plusOut	✓ wordEnds

Java Help

- Java Example Solution Code
- Java String Introduction (video)
- Java Substring v2 (video)
- Java String Equals and Loops
- Java String indexOf and Parsing
- Java If and Boolean Logic
- If Boolean Logic Example Solution Code 1 (video)
- If Boolean Logic Example Solution Code 2 (video)
- Java For and While Loops
- Java Arrays and Loops
- Java Map Introduction
- Java Map WordCount
- Java Functional Mapping
- Java Functional Filtering

Misc Code Practice

Поиск ENG 11:42 02.10.2023

codingbat.com — CodingBat Java String-2 doubleChar

String-2 > doubleChar

prev | next | chance

Given a string, return a string where for every char in the original, there are two chars.

doubleChar("The") → "TThhee"
doubleChar("AAbb") → "AAAAbbbb"
doubleChar("Hi-There") → "HHii--TThheerree"

Go ...Save, Compile, Run (ctrl-enter) Show Hint

```
public String doubleChar(String str) {
  String res="";
    for(int i=0; i < str.length(); i++){
      res = res + str.charAt(i) + str.charAt(i);
    }
  return res;
}
```

Go

Expected	Run		
doubleChar("The") → "TThhee"	"TThhee"	OK	
doubleChar("AAbb") → "AAAAbbbb"	"AAAAbbbb"	OK	
doubleChar("Hi-There") → "HHii--TThheerree"	"HHii--TThheerree"	OK	
doubleChar("Word!") → "WWoorrdd!!"	"WWoorrdd!!"	OK	
doubleChar("!!") → "!!!!"	"!!!!"	OK	
doubleChar("") → ""	""	OK	
doubleChar("a") → "aa"	"aa"	OK	
doubleChar(".") → ".."	".."	OK	
doubleChar("aa") → "aaaa"	"aaaa"	OK	
other tests		OK	

All Correct

Good job -- problem solved. You can see our solution as an alternative.

See Our Solution

next | chance

codingbat.com — CodingBat Java String-2 countHi

prev | next | chance

Return the number of times that the string "hi" appears anywhere in the given string.

countHi("abc hi ho") → 1
countHi("ABChi hi") → 2
countHi("hihi") → 2

Go ...Save, Compile, Run (ctrl-enter) Show Hint

```
public int countHi(String str) {
  int count = 0;
  if (str.length() == 1 && str.charAt(0) == 'h')
    count = 0;
  else {
    for(int i = 0; i < str.length(); i++) {
      if ((str.charAt(i) == 'h') && (str.charAt(i+1) == 'i'))
       count += 1;
      }
  }
  return count;
}
```

Go

Expected	Run		
countHi("abc hi ho") → 1	1	OK	
countHi("ABChi hi") → 2	2	OK	
countHi("hihi") → 2	2	OK	
countHi("hiHIhi") → 2	2	OK	
countHi("") → 0	0	OK	
countHi("h") → 0	0	OK	
countHi("hi") → 1	1	OK	
countHi("Hi is no HI on ahI") → 0	0	OK	
countHi("hiho not HOHIhi") → 2	2	OK	
other tests		OK	

All Correct

Good job -- problem solved. You can see our solution as an alternative.

See Our Solution

next | chance

Java > String-2

Поиск ENG 10:59 02.10.2023

codingbat.com

CodingBat Java String-2 catDog

prev | next | chance

Return true if the string "cat" and "dog" appear the same number of times in the given string.

catDog("catdog") → true
catDog("catcat") → false
catDog("1cat1cadodog") → true

Go ...Save, Compile, Run (ctrl-enter)

```
public boolean catDog(String str) {
  int len = str.length();
  int cat = 0;
  int dog = 0;

  for (int i = 0; i < len - 2; i++) {
    String temp = str.substring(i, i+3);
    if (temp.compareTo("cat") == 0)
      cat += 1;
    if (temp.compareTo("dog") == 0)
      dog += 1;
  }
  return (cat == dog) ? true : false;
}
```

Go

	Expected	Run		
catDog("catdog") → true	true	OK	
catDog("catcat") → false	false	OK	
catDog("1cat1cadodog") → true	true	OK	
catDog("catxxdogxxxdog") → false	false	OK	
catDog("catxdogxdogxcat") → true	true	OK	
catDog("catxdogxdogxca") → false	false	OK	
catDog("dogdogcat") → false	false	OK	
catDog("dogogcat") → true	true	OK	
catDog("dog") → false	false	OK	
catDog("cat") → false	false	OK	
catDog("ca") → true	true	OK	
catDog("c") → true	true	OK	
catDog("") → true	true	OK	
other tests		OK	

All Correct

next | chance

Поиск

ENG 11:03 02.10.2023

codingbat.com

CodingBat Java String-2 countCode

Return the number of times that the string "code" appears anywhere in the given string, except we'll accept any letter for the 'd', so "cope" and "cooe" count.

countCode("aaacodebbb") → 1
countCode("codexxcode") → 2
countCode("cozexxcope") → 2

Go ...Save, Compile, Run (ctrl-enter)

```
public int countCode(String str) {
  int len = str.length();
  int count = 0;
  String co = "co";
  String e = "e";

  if (len < 4)
    return 0;
  for (int i = 0; i < len - 3; i++) {
    if (co.compareTo(str.substring(i,i+2)) == 0
    && e.compareTo(str.substring(i+3, i+4)) == 0)
      count++;
  }
  return count;
}
```

Go

Editor font size %: 125

Shorter output

	Expected	Run	
countCode("aaacodebbb") → 1	1	OK	
countCode("codexxcode") → 2	2	OK	
countCode("cozexxcope") → 2	2	OK	
countCode("cozfxxcope") → 1	1	OK	
countCode("xxcozeyycop") → 1	1	OK	
countCode("cozcop") → 0	0	OK	
countCode("abcxyz") → 0	0	OK	
countCode("code") → 1	1	OK	
countCode("ode") → 0	0	OK	
countCode("c") → 0	0	OK	
countCode("") → 0	0	OK	
countCode("AAcodeBBcoleCCccoreDD") → 3	3	OK	
countCode("AAcodeBBcoleCCccorfDD") → 2	2	OK	
countCode("coAcodeBcoleccoreDD") → 3	3	OK	
other tests		OK	

All Correct

next | chance

Поиск

ENG 11:05 02.10.2023

codingbat.com

CodingBat Java String-2 endOther

string, ignoring upper/lower case differences (in other words, the computation should not be "case sensitive"). Note: str.toLowerCase() returns the lowercase version of a string.

endOther("Hiabc", "abc") → true
endOther("AbC", "HiaBc") → true
endOther("abc", "abXabc") → true

Go ...Save, Compile, Run (ctrl-enter)

```
public boolean endOther(String a, String b) {
   a = a.toLowerCase();
  int alen = a.length();
  b = b.toLowerCase();
  int blen = b.length();

  if (alen < blen) {
    String temp = b.substring(blen - alen, blen);
    return (temp.compareTo(a) == 0)? true : false;

  } else {
    String temp = a.substring(alen - blen, alen);
    return (temp.compareTo(b) == 0) ? true : false;
  }
}
```

Go

Editor font size %: 125

Shorter output

Expected	Run		
endOther("Hiabc", "abc") → true	true	OK	
endOther("AbC", "HiaBc") → true	true	OK	
endOther("abc", "abXabc") → true	true	OK	
endOther("Hiabc", "abcd") → false	false	OK	
endOther("Hiabc", "bc") → true	true	OK	
endOther("Hiabcx", "bc") → false	false	OK	
endOther("abc", "abc") → true	true	OK	
endOther("xyz", "12xyz") → true	true	OK	
endOther("yz", "12xz") → false	false	OK	
endOther("Z", "12xz") → true	true	OK	
endOther("12", "12") → true	true	OK	
endOther("abcXYZ", "abcDEF") → false	false	OK	
endOther("ab", "ab12") → false	false	OK	
endOther("ab", "12ab") → true	true	OK	
other tests		OK	

All Correct

next | chance

Java > String-2

Поиск

ENG

11:08
02.10.2023

codingbat.com — CodingBat Java String-2 bobThere

String-2 > bobThere

prev | next | chance

Return true if the given string contains a "bob" string, but where the middle 'o' char can be any char.

bobThere("abcbob") → true
bobThere("b9b") → true
bobThere("bac") → false

Go ...Save, Compile, Run (ctrl-enter)

```
public boolean bobThere(String str) {
  int len = str.length();

  for (int i = 0; i < len - 2; i++) {
    if (str.charAt(i) == 'b' && str.charAt(i+2) == 'b')
    return true;
  }
  return false;
}
```

Go

Expected	Run		
bobThere("abcbob") → true	true	OK	
bobThere("b9b") → true	true	OK	
bobThere("bac") → false	false	OK	
bobThere("bbb") → true	true	OK	
bobThere("abcdefb") → false	false	OK	
bobThere("123abcbcdbabxyz") → true	true	OK	
bobThere("b12") → false	false	OK	
bobThere("b1b") → true	true	OK	
bobThere("b12b1b") → true	true	OK	
bobThere("bbc") → false	false	OK	
bobThere("bbb") → true	true	OK	
bobThere("bb") → false	false	OK	
bobThere("b") → false	false	OK	
other tests		OK	

All Correct

Поиск ENG 11:17 02.10.2023

codingbat.com — CodingBat Java String-2 xyBalance

balance multiple 'x's. Return true if the given string is xy-balanced.

xyBalance("aaxbby") → true
xyBalance("aaxbb") → false
xyBalance("yaaxbb") → false

Go ...Save, Compile, Run (ctrl-enter)

```
public boolean xyBalance(String str) {
  Boolean x = false;
  Boolean y = false;
  int len = str.length();

  for (int i = 0; i < len; i++) {
    if (str.charAt(i) == 'x' && y == true){
      x = true;
      y = false;
    } else if (str.charAt(i) == 'x') {
      x = true;
    }

    if (str.charAt(i) == 'y' && x == true)
      y = true;
  }
  if (x == false)
    y = true;
  return y;
}
```

Go

Editor font size %: 125

Shorter output

	Expected	Run	
xyBalance("aaxbby") → true	true	OK	
xyBalance("aaxbb") → false	false	OK	
xyBalance("yaaxbb") → false	false	OK	
xyBalance("yaaxbby") → true	true	OK	
xyBalance("xaxxbby") → true	true	OK	
xyBalance("xaxxbbyx") → false	false	OK	
xyBalance("xxbxy") → true	true	OK	
xyBalance("xxbx") → false	false	OK	
xyBalance("bbb") → true	true	OK	
xyBalance("bxbb") → false	false	OK	
xyBalance("bxyb") → true	true	OK	
xyBalance("xy") → true	true	OK	
xyBalance("y") → true	true	OK	
xyBalance("x") → false	false	OK	
xyBalance("") → true	true	OK	
xyBalance("yxyxyxyx") → false	false	OK	
xyBalance("yxyxyxyxy") → true	true	OK	
xyBalance("12xabxxydxyxyzz") → true	true	OK	
other tests		OK	

Поиск ENG 11:28 02.10.2023

codingbat.com — CodingBat Java String-2 repeatEnd

String-2 > repeatEnd

prev | next | chance

Given a string and an int n, return a string made of n repetitions of the last n characters of the string. You may assume that n is between 0 and the length of the string, inclusive.

repeatEnd("Hello", 3) → "llollollo"
repeatEnd("Hello", 2) → "lolo"
repeatEnd("Hello", 1) → "o"

Go ...Save, Compile, Run (ctrl-enter)

```
public String repeatEnd(String str, int n) {
  int len = str.length();
  String word = "";

  for (int i = 0; i < n; i++) {
    word += str.substring(len - n, len);
  }
  return word;
}
```

Go

Expected	Run		
repeatEnd("Hello", 3) → "llollollo"	"llollollo"	OK	
repeatEnd("Hello", 2) → "lolo"	"lolo"	OK	
repeatEnd("Hello", 1) → "o"	"o"	OK	
repeatEnd("Hello", 0) → ""	""	OK	
repeatEnd("abc", 3) → "abcabcabc"	"abcabcabc"	OK	
repeatEnd("1234", 2) → "3434"	"3434"	OK	
repeatEnd("1234", 3) → "234234234"	"234234234"	OK	
repeatEnd("", 0) → ""	""	OK	
other tests		OK	

All Correct

next | chance

Java > String-2

hakloi@mail.ru done page

Your progress graph for this problem

codingbat.com

CodingBat Java String-2 repeatFront

prev | next | chance

Given a string and an int n, return a string made of the first n characters of the string, followed by the first n-1 characters of the string, and so on. You may assume that n is between 0 and the length of the string, inclusive (i.e. n >= 0 and n <= str.length()).

repeatFront("Chocolate", 4) → "ChocChoChC"
repeatFront("Chocolate", 3) → "ChoChC"
repeatFront("Ice Cream", 2) → "IcI"

Go ...Save, Compile, Run (ctrl-enter)

```
public String repeatFront(String str, int n) {
  int len = str.length();
  String word = "";

  for (int i = n; n > 0; n--) {
    word += str.substring(0, n);
  }
  return word;

}
```

Go

Expected	Run		
repeatFront("Chocolate", 4) → "ChocChoChC"	"ChocChoChC"	OK	
repeatFront("Chocolate", 3) → "ChoChC"	"ChoChC"	OK	
repeatFront("Ice Cream", 2) → "IcI"	"IcI"	OK	
repeatFront("Ice Cream", 1) → "I"	"I"	OK	
repeatFront("Ice Cream", 0) → ""	""	OK	
repeatFront("xyz", 3) → "xyzxyx"	"xyzxyx"	OK	
repeatFront("", 0) → ""	""	OK	
repeatFront("Java", 4) → "JavaJavJaJ"	"JavaJavJaJ"	OK	
repeatFront("Java", 1) → "J"	"J"	OK	
other tests		OK	

All Correct

next | chance

Java > String-2

hakloi@mail.ru done page

Your progress graph for this problem

Поиск

ENG 11:39 02.10.2023

codingbat.com

CodingBat Java String-2 repeatSeparator

prev | next | chance

Given two strings, **word** and a separator **sep**, return a big string made of **count** occurrences of the word, separated by the separator string.

repeatSeparator("Word", "X", 3) → "WordXWordXWord"
repeatSeparator("This", "And", 2) → "ThisAndThis"
repeatSeparator("This", "And", 1) → "This"

Go ...Save, Compile, Run (ctrl-enter)

```
public String repeatSeparator(String word, String sep, int count) {
  String word2 = "";

  for (int i = 0; i < count; i++) {
    if (i < count-1)
      word2 += word + sep;
    else
      word2 += word;
  }
  return word2;
}
```

Go

Expected	Run		
repeatSeparator("Word", "X", 3) → "WordXWordXWord"	"WordXWordXWord"	OK	
repeatSeparator("This", "And", 2) → "ThisAndThis"	"ThisAndThis"	OK	
repeatSeparator("This", "And", 1) → "This"	"This"	OK	
repeatSeparator("Hi", "-n-", 2) → "Hi-n-Hi"	"Hi-n-Hi"	OK	
repeatSeparator("AAA", "", 1) → "AAA"	"AAA"	OK	
repeatSeparator("AAA", "", 0) → ""	""	OK	
repeatSeparator("A", "B", 5) → "ABABABABA"	"ABABABABA"	OK	
repeatSeparator("abc", "XX", 3) → "abcXXabcXXabc"	"abcXXabcXXabc"	OK	
repeatSeparator("abc", "XX", 2) → "abcXXabc"	"abcXXabc"	OK	
repeatSeparator("abc", "XX", 1) → "abc"	"abc"	OK	
repeatSeparator("XYZ", "a", 2) → "XYZaXYZ"	"XYZaXYZ"	OK	
other tests		OK	

Поиск ENG 11:41 02.10.2023