

Kata Training

4 kyuVigenère Cipher Helper

☆ 785🏆 130👍 90% of 974🕒 3,029 of 7,197👤 jacobb🚩 7 Issues Reported

InstructionsOutput

Time: 869msPassed: 4Failed: 0

Test Results:

Testing with lowercase latin alphabet

- Password of 'password'
- Password of 'pizza'
- Password of 'attackatdawn', random tokens

Completed in 2ms

Testing with katakana alphabet

- Password of 'カタカナ'

You have passed all of the tests! :)

JavaScriptNode v18.x

VIMEMACS

Solution

```
22 for (let i = 0; i < str.length; i++) {
23     // якщо символу в алфавіті немає - додати символ з даного рядка
24     if(abc.indexOf(str[i]) === -1){
25         cipher += str[i];
26     }
27     else{ // додати символ, який знаходиться в алфавіті на відстані відповідного символу з key від символу із заданого рядка
28         cipher += abc.slice((abc.indexOf(str[i]) - abc.indexOf(newKey[i])) % abc.length, ((abc.indexOf(str[i]) - abc.indexC
29     )
30     }
31     return cipher;
32 };
33
34 // створення ключа, який буде довжиною як задана строка
35 this.createKey = function (str){
36     const concatenationNum = Math.floor(str.length / key.length);
37     const sliceNum = str.length % key.length;
38
39     let newKey = key.repeat(concatenationNum);
40     newKey += key.slice(0, sliceNum);
41     return newKey;
42 }
```

Good Job! You may take your time to refactor/comment your solution. Submit when ready.

Sample Tests

```
1 const Test = require('@codewars/test-compat');
2
3 describe("Tests", () => {
4     it("test", () => {
5         var abc, key;
6         abc = "abcdefghijklmnopqrstuvwxyz";
7         key = "password"
8         c = new VigenèreCipher(key, abc);
9
10        Test.assertEquals(c.encode('codewars'), 'rovwsoiv');
11        Test.assertEquals(c.decode('rovwsoiv'), 'codewars');
12
13        Test.assertEquals(c.encode('waffles'), 'laxxhsj');
14        Test.assertEquals(c.decode('laxxhsj'), 'waffles');
```

SKIPDISCUSS (156)RESETTESTSUBMIT

**5 kyu** Vector class

☆ 562 📁 122 📈 88% of 819 🕒 2,957 of 7,223 👤 eugene-bulkin ⚠️ 8 Issues Reported

Instructions Output

Time: 885ms Passed: 6 Failed: 0

## Test Results:

## Testing arithmetic

- > Addition
- > Subtraction
- > Dot Product
- > Norms

Completed in 3ms

## Auxiliary function

- > Equality
- > Strings

You have passed all of the tests! :)

JavaScript

Node v18.x

VIM

EMACS

## Solution

```
42 this.toString = function () {
43   let str = '()';
44   str = str.slice(0, 1) + components.join(',') + str.slice(-1);
45   return str;
46 };
47
48 this.equals = function (secondVector) {
49   if (components.length !== secondVector.components.length) {
50     return false;
51   }
52   for (let i = 0; i < components.length; i++) {
53     if (components[i] !== secondVector.components[i]) {
54       return false;
55     }
56   }
57   return true;
58 };
59
60 this.isEqualLength = function (secondVector) {
61   if (components.length !== secondVector.components.length) {
62     throw new Error('error');
63   }
64 }
```

👏 Impressive! You may take your time to refactor/comment your solution. Submit when ready.

## Sample Tests

```
1 const Test = require('@codewars/test-compat');
2
3 describe("Tests", () => {
4   it("test", () => {
5     var a = new Vector([1,2]);
6     var b = new Vector([3,4]);
7
8     Test.expect(a.add(b).equals(new Vector([4,6])));
9   });
10 });
11
```

SKIP

UNLOCK SOLUTIONS

DISCUSS (98)

RESET

TEST

SUBMIT

Kata Training

4 kyu

The Greatest Warrior

☆ 645

📁 133

📈 93% of 564

👤 810 of 2,987

👤 BurstNova

🚨 6 Issues Reported

Instructions

Output

Time: 739ms

Passed: 2

Failed: 0

Test Results:

▼ Kata Tests

> Standard Tests

> Random Tests

Completed in 4ms

You have passed all of the tests! :)

JavaScript

Node v18.x

VIM

EMACS

Solution

```
24 battle = (lvl) => {
25   if(lvl < 1 || lvl > 100){
26     return "Invalid level";
27   }
28   if(((lvl - this.level()) >= 5) && (Math.floor(this.level()/10) < Math.floor(lvl/10))){
29     return "You've been defeated";
30   }
31   if(lvl === this.level()){
32     this.expAdd(10);
33     return "A good fight";
34   }
35   if((this.level() - lvl) === 1){
36     this.expAdd(5);
37     return "A good fight";
38   }
39   if((this.level() - lvl) > 1){
40     return "Easy fight";
41   }
42   if((lvl - this.level()) > 0){
43     this.expAdd(20 * Math.abs(lvl - this.level()) * Math.abs(lvl - this.level()));
44     return "An intense fight";
45   }
46 }
```

Correct! You may take your time to refactor/comment your solution. Submit when ready.

Sample Tests

```
6 Test.assertDeepEquals(Goku.level(), 1, "All warriors must start as LV 1")
7 Test.assertDeepEquals(Goku.rank(), "Pushover", "All warriors must start as a Pushover")
8 Test.assertDeepEquals(Goku.achievements(), [], "All warriors must start off with no achievements to their name")
9 Test.assertDeepEquals(Goku.training(["Do ten push-ups", 95, 1]), "Do ten push-ups", "After obtaining a mastery, the achievement
10 Test.assertDeepEquals(Goku.level(), 1, "After your warriors's first training, he/she must still be LV 1")
11 Test.assertDeepEquals(Goku.battle(0), "Invalid level", "LV 0 enemies do not exist. This must be a mistake!")
12 Test.assertDeepEquals(Goku.battle(1), "A good fight", "After fighting against a warrior of equal level, your warrior should pro
13 Test.assertDeepEquals(Goku.level(), 2, "With that fight under your belt, your warrior should now be ascended to LV 2")
14 Test.assertDeepEquals(Goku.achievements(), ["Do ten push-ups"], "Don't forget what your warrior has achieved so far")
15 Test.assertDeepEquals(Goku.rank(), "Pushover", "A pushover, however, your warrior should remain")
16 Test.assertDeepEquals(Goku.battle(3), "An intense fight", "Another battle, this time with a slightly more skilled enemy, should
17 Test.assertDeepEquals(Goku.level(), 2, "Your should still remain LV 2")
18 Test.assertDeepEquals(Goku.training(["Survive one night at the Forest of Death", 170, 2]), "Survive one night at the Forest of
19 Test.assertDeepEquals(Goku.battle(1580), "Not strong enough", "Your warrior isn't skilled
20 Test.assertDeepEquals(Goku.battle(1580), "Not strong enough", "Your warrior isn't skilled")
```

SKIP

DISCUSS (74)

RESET

TEST

SUBMIT



## 5 kyu PaginationHelper

☆ 1652 314 81% of 3,148 12,376 of 34,208 jhoffner 8 Issues Reported

Instructions

Output

Time: 860ms Passed: 102 Failed: 0

## Test Results:

## Tests suite

&gt; sample test : 24 items with 10 per page

&gt; empty collection

&gt; random tests

Completed in 19ms

You have passed all of the tests! :)

JavaScript



Node v18.x



VIM

EMACS



## Solution

```
12
13
14
15
16   msPerPage);
17
18
19   . page_index is zero based.
20   es that are out of range
21   < 0){
22
23
24   this.itemCount() % this.itemsPerPage === 0) ? this.itemsPerPage : this.collection.length % this.itemsPerPage) : this.itemsPerPage;
25
26
27   indexes
28   es that are out of range
29   < 0){
30
31
32   ge) - 1;
33
```

👏 Outstanding! You may take your time to refactor/comment your solution. Submit when ready.

## Sample Tests

```
1 describe("Tests suite", () => {
2   const { strictEqual } = require('chai').assert;
3
4   function doTest(instance, methodName, expected, ...args) {
5     const actual = instance[methodName](...args);
6     strictEqual(actual, expected, `for ${methodName}(${args.join(', ')} )`);
7   }
8
9   it("sample test : 24 items with 10 per page", () => {
10    const collection = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24]
11    const helper = new PaginationHelper(collection, 10)
12
13    doTest(helper, 'pageCount', 3);
14    doTest(helper, 'itemCount', 24);
```

SKIP

UNLOCK SOLUTIONS

DISCUSS (265)

RESET

TEST

SUBMIT