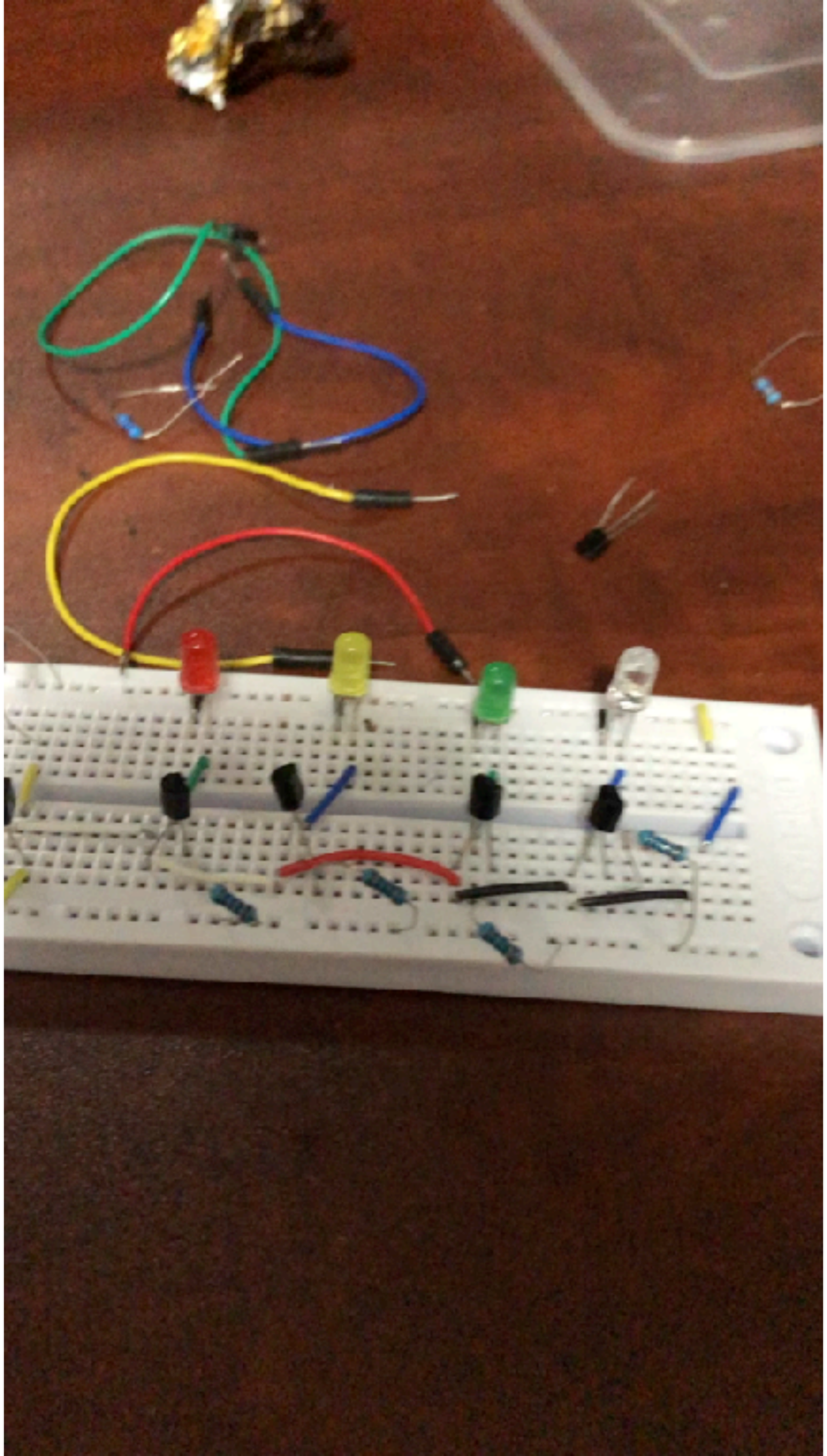
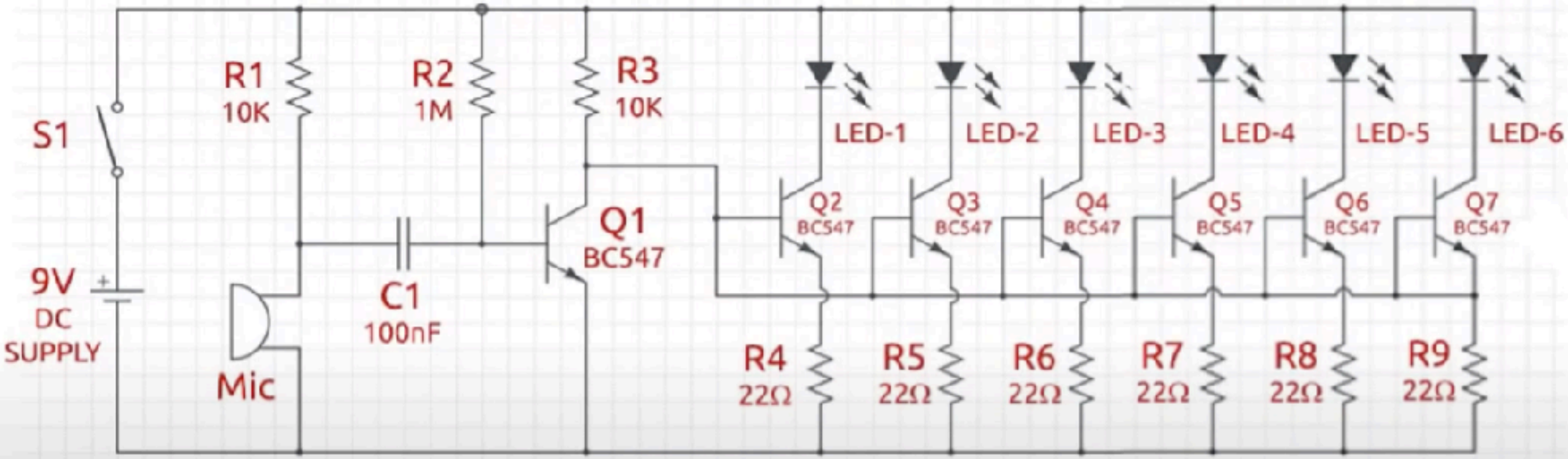
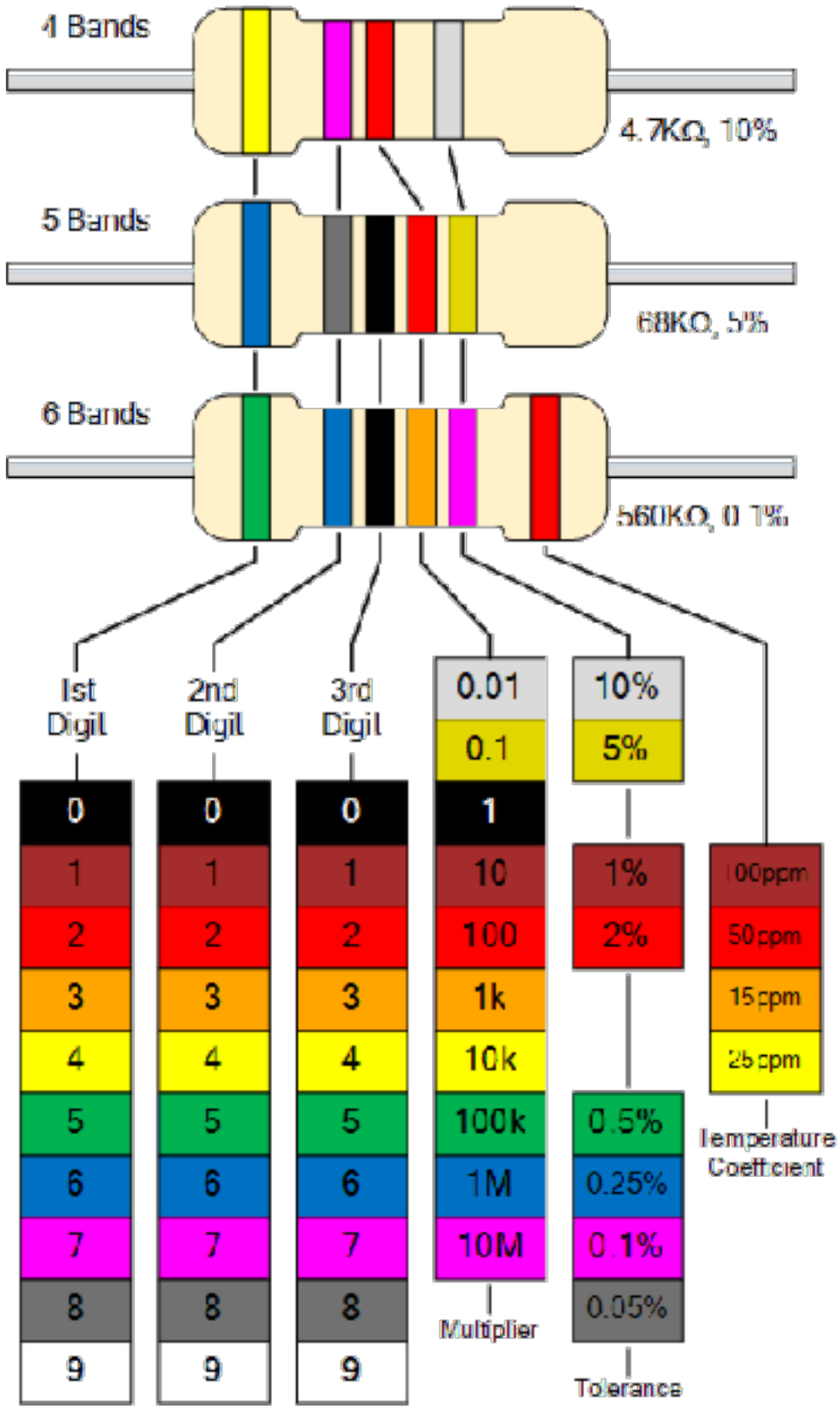
我喜欢的一些东西

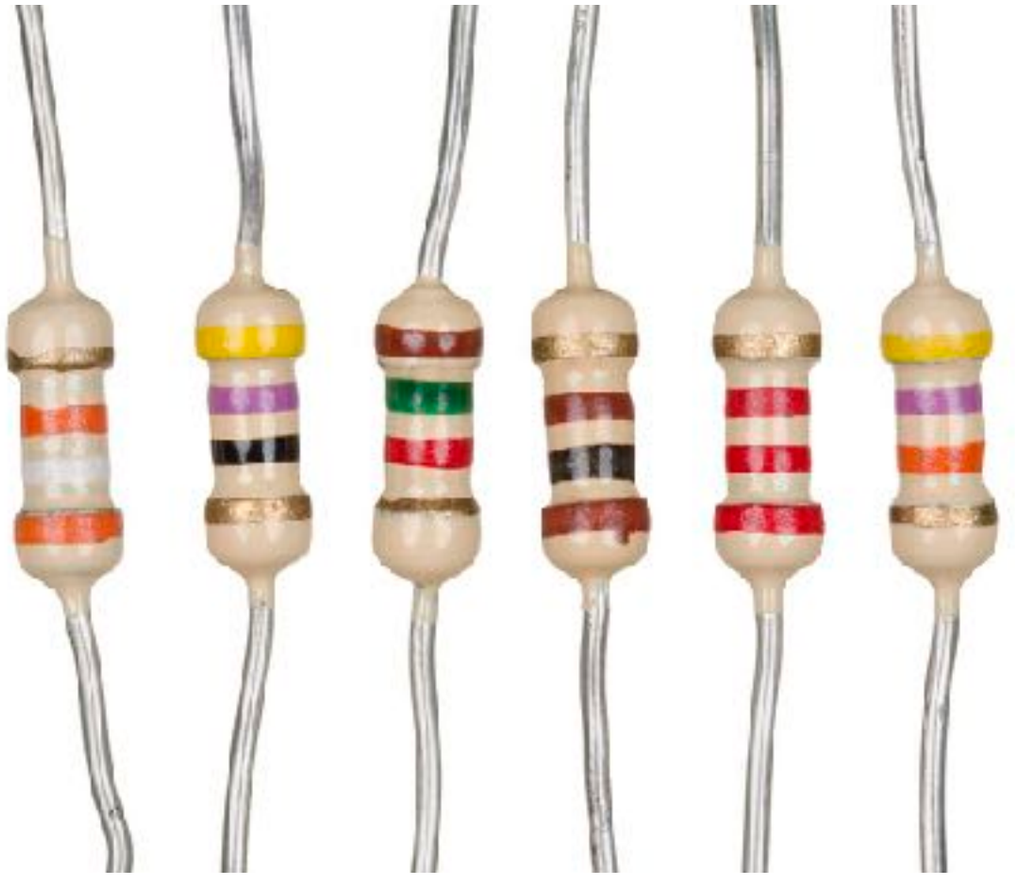
李智维 2021.8.5



MUSIC RHTYHM LED FLASHLIGHT CIRCUI




4 Bands
4.7KΩ, 10%
5 Bands
68KΩ, 5%
6 Bands
560KΩ, 0.1%
1st Digit
2nd Digit
3rd Digit
Multiplier
Tolerance
Temperature Coefficient
0.01
0.1
1
10
100
1k
10k
100k
1M
10M
10%
5%
1%
2%
0.5%
0.25%
0.1%
0.05%
100ppm
50ppm
15ppm
25ppm



色环数

4 环 5 环 6 环

电阻器参数

第 1 色环

棕色 1

第 2 色环

黑色 0

第 3 色环

红色 2

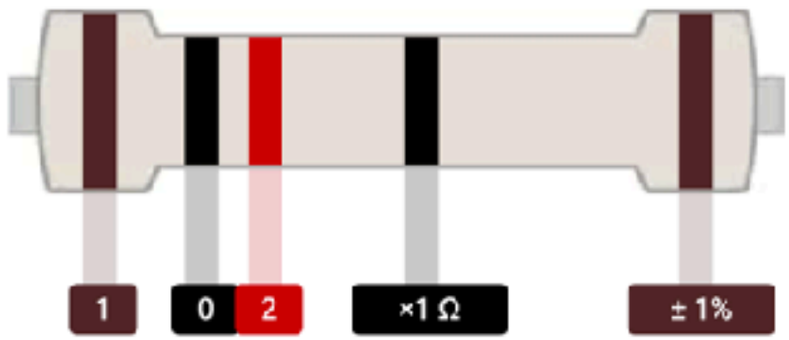
乘数

黑色 ×1 Ω

公差

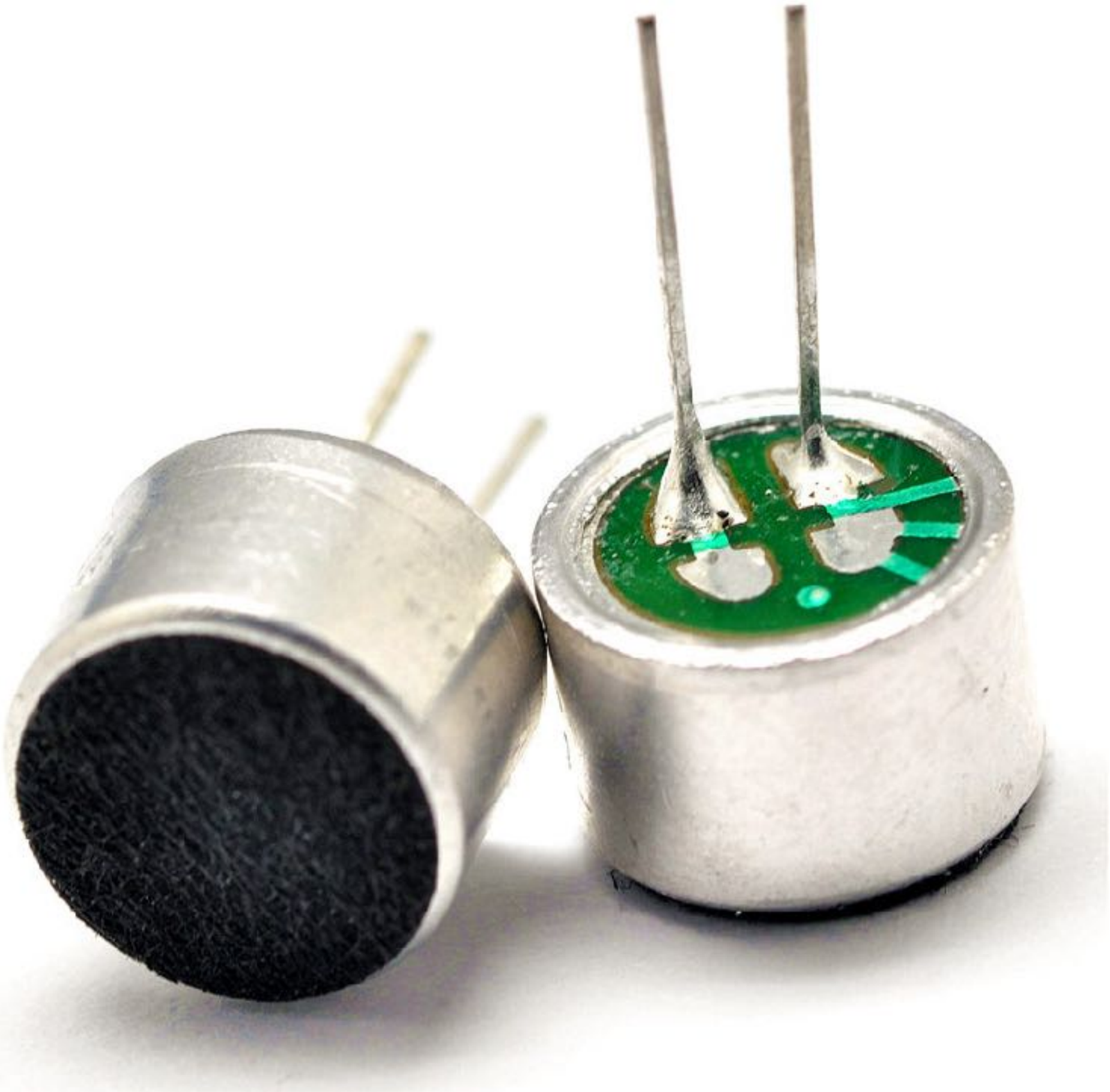
棕色 ± 1%

输出



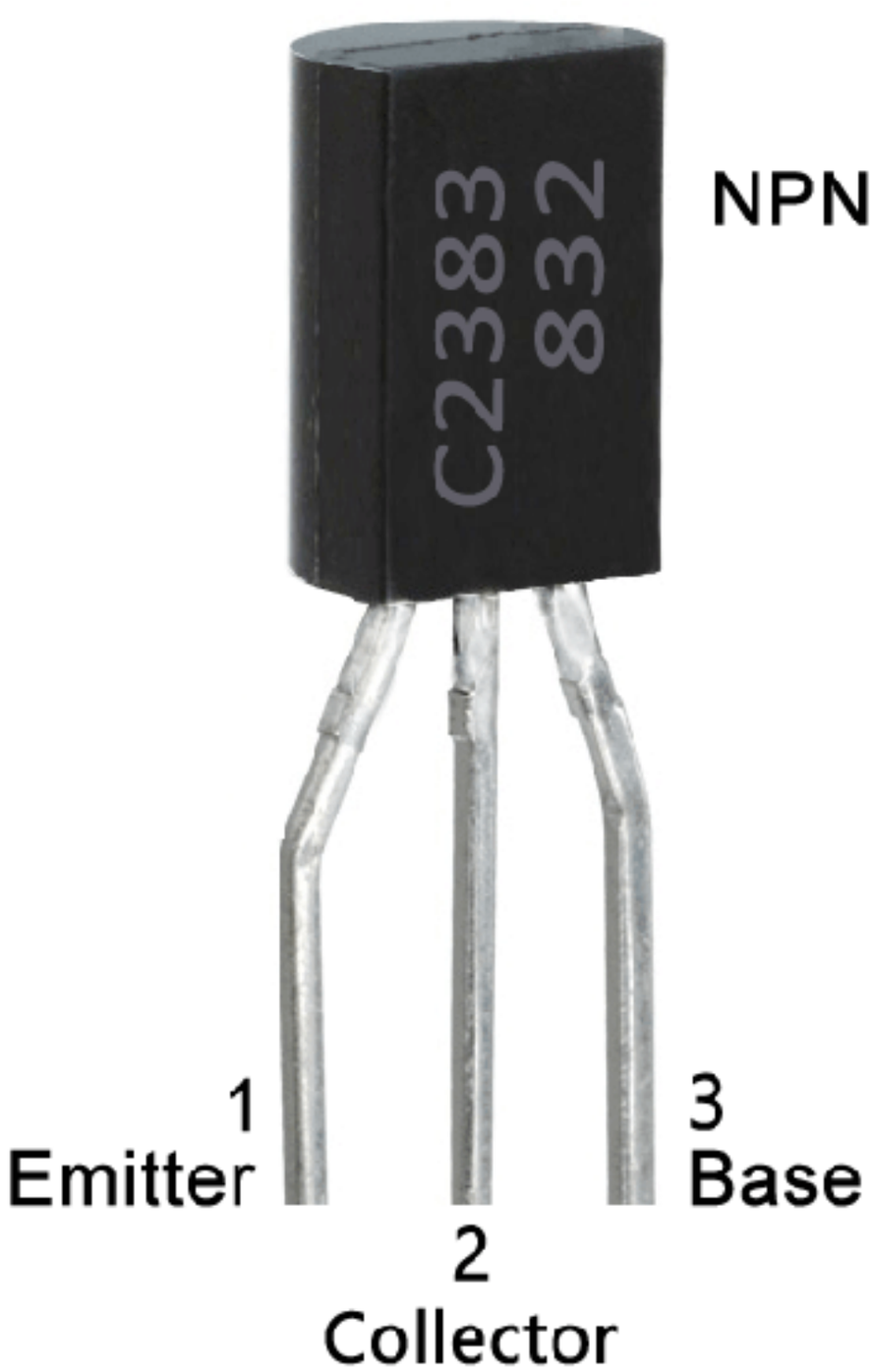
电阻值:
102 Ohms 1%


VICTOR VC890C+
DIGITAL MULTIMETER
AUTO POWER OFF
NPN
PNP
EBCE
EBCE
OFF
hFE
200mV
2V
20V
200V
1000V
750V
200V
20V
2V
V~
20mA
200mA
20A
20A
A⎓
A~
20A
TEMP
VΩ
mA
COM
MAX
200mA
FUSED
MAX
1000V DC
750V AC
CAT III
600V
MAX
20A
FUSED
20MΩ
2MΩ
200kΩ
20kΩ
2kΩ
200Ω
200µ
2µ
F
20m
°C
200µA
20mA
200mA
Baidu经验
jingyan.baidu.com

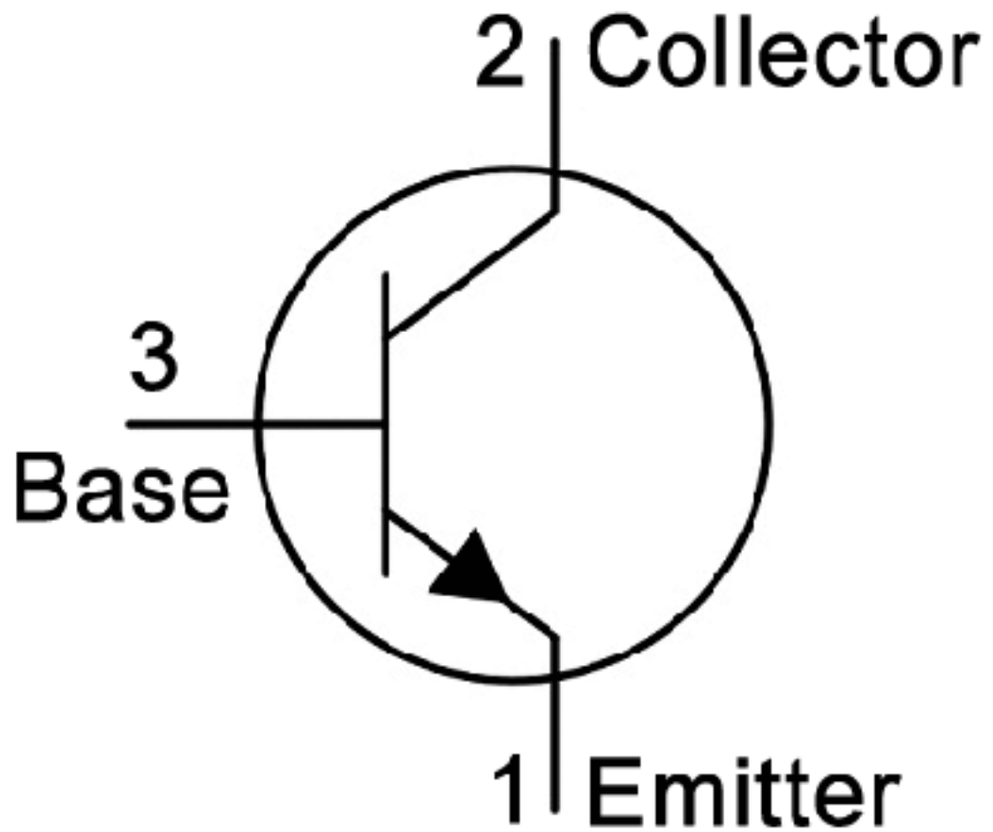


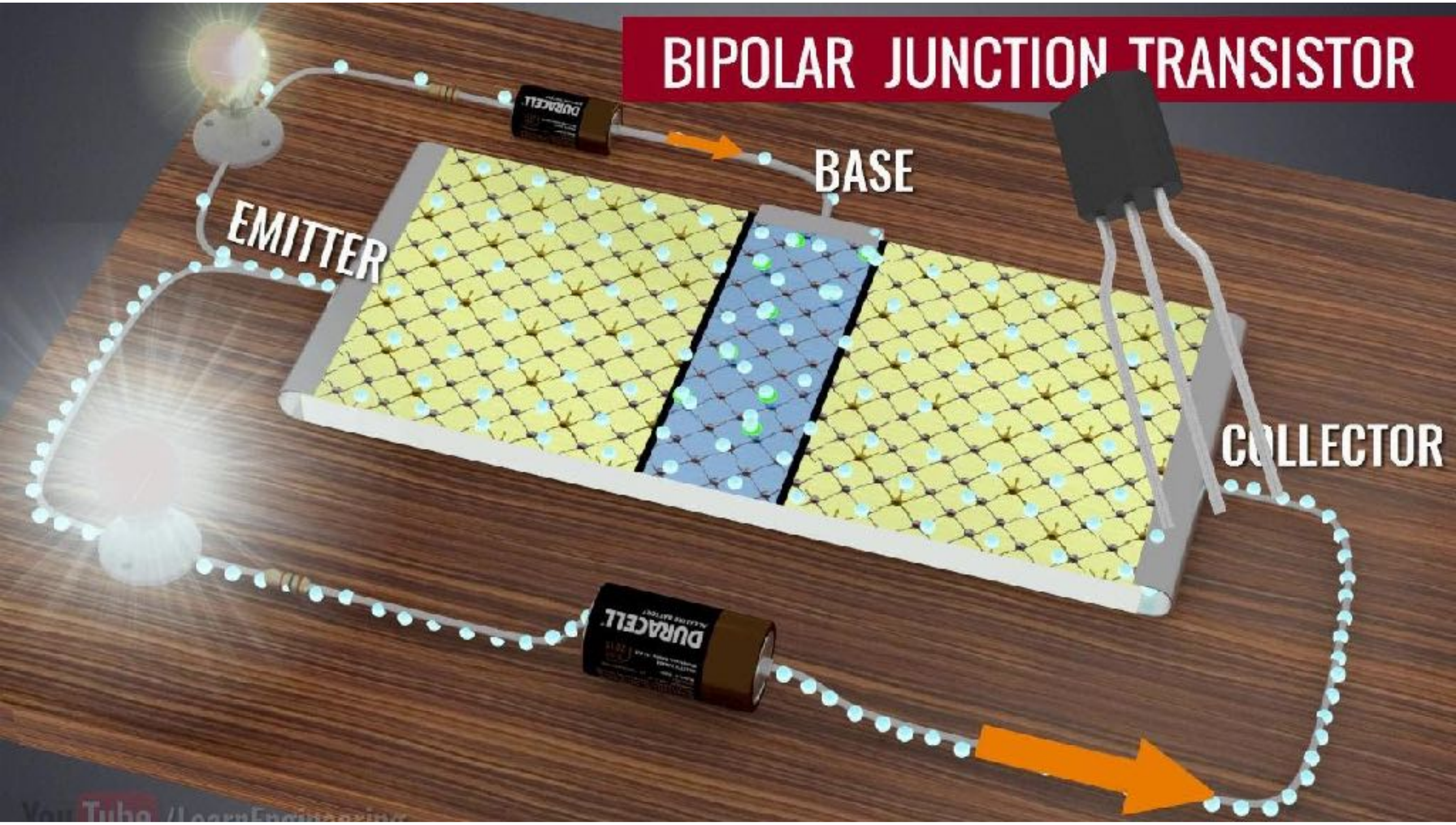
2SC2383 Transistor Pinout

TO-92L Package



NPN



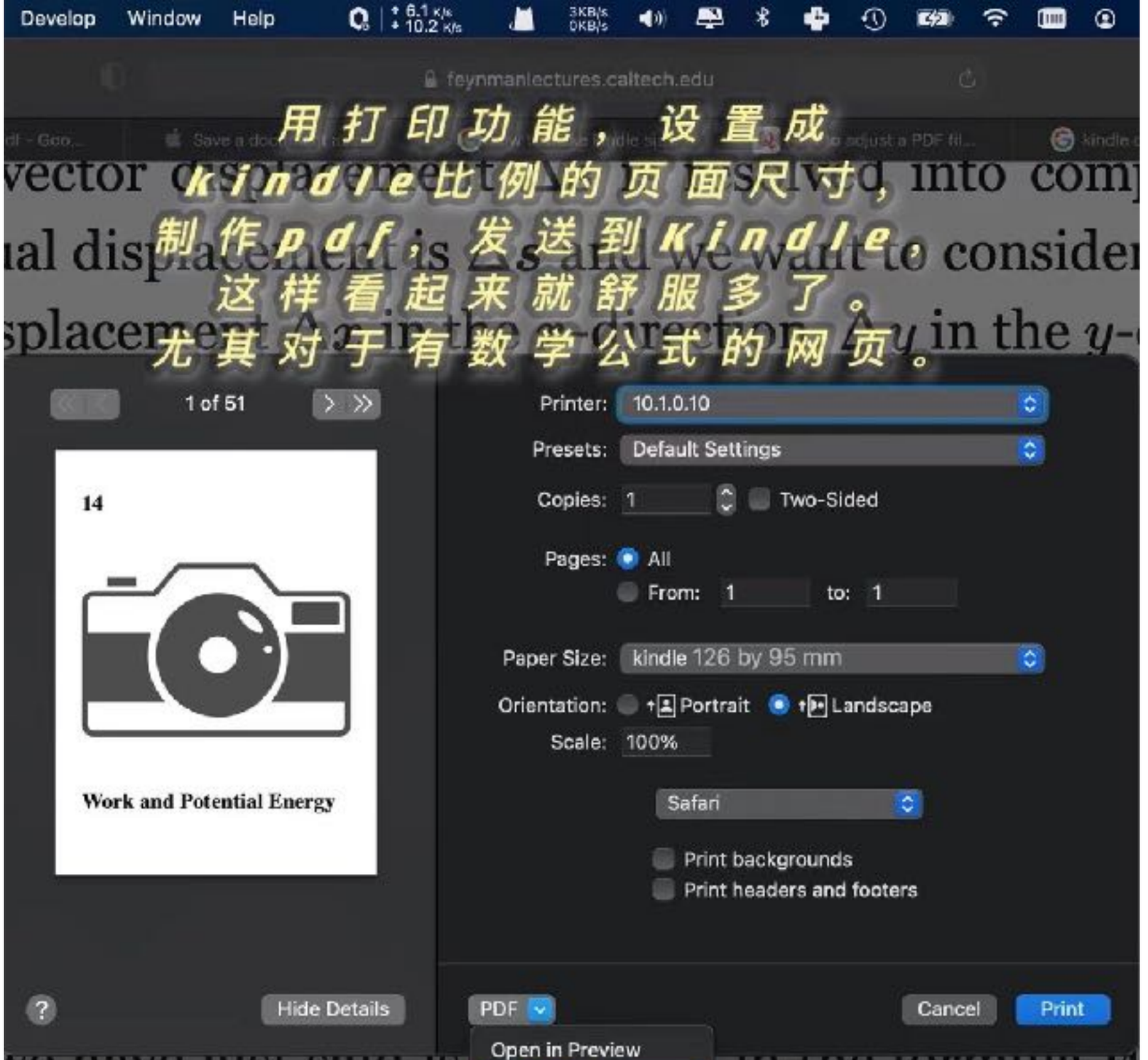

BIPOLAR JUNCTION TRANSISTOR
EMITTER
BASE
COLLECTOR
DURACELL
DURACELL
YouTube /LearnEngineering


7L30
7L30
7L30
105°C 105°C 105°C
M47
M47

10 x 1	10 x 10	10 x 100	10 x 1,000	10 x 10,000
=10pF	=100pF	=1,000pF	=10,000pF	=100,000pF

Kindle阅读技巧

用打印功能，设置成
kindle比例的页面尺寸，
制作pdf，发送到Kindle。
这样看起来就舒服多了。
尤其对于有数学公式的网页。



Data Miner

Data Miner

Free Office Hours | Contact Us | My Account

Data Miner 5
User Manual

SCRAPE
Page Scrape
Crawl Scrape
New Recipe

MY DATA
Saved Results (14)
Uploaded Files (0)

Version 5.2.74
2021 © Data Miner

(?) (1) Select a Recipe (2) Scrape (3) Download

Select a recipe to scrape the page.

Next Step

Public recipes are made and shared by our community of users. **Generic** recipes are made by Data Miner Team and extract common content you may see online. **My Recipes** are the ones you made or copied from other users. Usually they are private and visible only by you.

Public 0 | Generic 4 | My Recipes 7 | Pin 0

Recipes Type: Single Page Auto Pagination

Search by recipe name

Karpathy
Columns (1): Column 1
Site: karpathy.github.io

Sam Altman Blog
Columns (1): Column 1
Site: blog.samaltman.com

WeiboText
Columns (1): C
Site: weibo.com

YinWang
Columns (1): Column 1

Wechat Reward Name

Status: Stopped! **158** rows and 1 column scraped

1	Column 1
2	https://www.yinwang.org/blog-cn/2021/02/16/cs-reading-c
3	https://www.yinwang.org/blog-cn/2021/02/15/cs3
4	https://www.yinwang.org/blog-cn/2021/01/19/bach-mozart
5	https://www.yinwang.org/blog-cn/2020/12/30/guitar-tuning
6	https://www.yinwang.org/blog-cn/2020/12/19/music
7	https://www.yinwang.org/blog-cn/2020/11/12/china-esthet
8	https://www.yinwang.org/blog-cn/2020/09/24/cs-class2
9	https://www.yinwang.org/blog-cn/2020/08/28/life-imitating
10	https://www.yinwang.org/blog-cn/2020/08/24/ganhuo
	log-cn/2020/08/23/xueba
	log-cn/2020/06/10/new-ugly-ch
	g/blog-cn/2020/05/23/american-en
	log-cn/2020/04/30/individualize
	3/26/entry-level-c
	cn/2020/03/23/wisdom-of-ir
17	https://www.yinwang.org/blog-cn/2020/03/06/english-learn

DataMiner，很方便的Chrome爬虫插件，简单几步轻松爬取链接，配合Getbook来把博客网站制作成电子书

You are scraping this tab: https://www.yinwang.org/#

Installation

Install with pip:

```
$ pip install getbook
```

Note: this program only works on Python3.5+.

You may need to install kindlegen to create mobi format books.

Guide

There are serval ways to generate books.

1. You can generate a book from a feed:

   ```
   $ getbook -u http://example.com/feed
   ```

2. It is also possible to generate books from a config JSON file:

   ```
   $ getbook -f ./book.json
   ```

Use `--mobi` to generate mobi file for kindle:

```
$ getbook -f ./book.json --mobi
```

Use `--epub` to generate epub file:

Search or jump to... / Pull requests Issues Marketplace Explore

lepture / getbook

Code Issues 2 Pull requests Actions Projects Wiki Security Insights

master 1 branch 1 tag Go to file Add file Code

lepture Merge pull request #7 from lzwjava/scope ... 10c7aa3 on Mar 6 33 commits

demo	Add demos	3 years ago
getbook	fix beautifulsoup selector problem	5 months ago
.gitignore	ignore quora truncated text	2 years ago
LICENSE	Update readme and license	3 years ago
MANIFEST.in	Version bump 0.2	3 years ago
README.rst	Update readme	16 months ago
run.py	copy ebook code from doocer	3 years ago
setup.py	Version bump 0.3	2 years ago

README.rst

pdfjam --keepinfo --scale 1.2 --clip true —
suffix "cropped" Java.pdf

```
# Ruby 1.9 only
Object.superclass           # BasicObject: Object has a superclass in 1.9
BasicObject.superclass      # nil: BasicObject has no superclass
```

See §7.3 for more on `BasicObject`.

So a particularly straightforward way to check the class of an object is by direct comparison:

```
o.class == String       # true if o is a String
```

The `instance_of?` method does the same thing and is a little more elegant:

```
o.instance_of? String   # true if o is a String
```

Usually when we test the class of an object, we would also like to know if the object is an instance of any subclass of that class. To test this, use the `is_a?` method, or its synonym `kind_of?`:

```
x = 1                   # This is the value we're working with
x.instance_of? Fixnum   # true: is an instance of Fixnum
x.instance_of? Numeric  # false: instance_of? doesn't check inheritance
x.is_a? Fixnum          # true: x is a Fixnum
x.is_a? Integer         # true: x is an Integer
x.is_a? Numeric         # true: x is a Numeric
x.is_a? Comparable      # true: works with mixin modules, too
x.is_a? Object          # true for any value of x
```

The `Class` class defines the `===` operator in such a way that it can be used in place of `is_a?`:

```
Numeric === x           # true: x is a Numeric
```

This idiom is unique to Ruby and is probably less readable than using the more traditional `is_a?` method.

Every object has a well-defined class in Ruby, and that class never changes during the lifetime of the object. An object's *type*, on the other hand, is more fluid. The type of an object is related to its class, but the class is only part of an object's type. When we talk about the type of an object, we mean the set of behaviors that characterize the object. Another way to put it is that the type of an object is the set of methods it can respond to. (This definition becomes recursive because it is not just the name of the methods that matter, but also the types of arguments that those methods can accept.)

In Ruby programming, we often don't care about the class of an object, we just want to know whether we can invoke some method on it. Consider, for example, the `<<` operator. Arrays, strings, files, and other I/O-related classes define this as an append operator. If we are writing a method that produces textual output, we might write it generically to use this operator. Then our method can be invoked with any argument that implements `<<`. We don't care about the class of the argument, just that we can append to it. We can test for this with the `respond_to?` method:

```
o.respond_to? :"<<"  # true if o has an << operator
```

this surprising.) Each time Ruby encounters a string literal, it creates a new object. If you include a literal within the body of a loop, Ruby will create a new object for each iteration. You can demonstrate this for yourself as follows:

```
10.times { puts "test".object_id }
```

For efficiency, you should avoid using literals within loops.

3.2.1.8 The String.new method

In addition to all the string literal options described earlier, you can also create new strings with the `String.new` method. With no arguments, this method returns a newly created string with no characters. With a single string argument, it creates and returns a new `String` object that represents the same text as the argument object.

3.2.2 Character Literals

Single characters can be included literally in a Ruby program by preceding the character with a question mark. No quotation marks of any kind are used:

```
?A   # Character literal for the ASCII character A
?"   # Character literal for the double-quote character
??   # Character literal for the question mark character
```

Although Ruby has a character literal syntax, it does not have a special class to represent single characters. Also, the interpretation of character literals has changed between Ruby 1.8 and Ruby 1.9. In Ruby 1.8, character literals evaluate to the integer encoding of the specified character. `?A`, for example, is the same as 65 because the ASCII encoding for the capital letter A is the integer 65. In Ruby 1.8, the character literal syntax only works with ASCII and single-byte characters.

In Ruby 1.9 and later, characters are simply strings of length 1. That is, the literal `?A` is the same as the literal `'A'`, and there is really no need for this character literal syntax in new code. In Ruby 1.9, the character literal syntax works with multibyte characters and can also be used with the `\u` Unicode escape (though not with the multicodepoint form `\u{a b c}`):

```
?\u20AC == ?€    # => true: Ruby 1.9 only
?€ == "\u20AC"   # => true
```

The character literal syntax can actually be used with any of the character escapes listed earlier in Table 3-1:

```
?\t      # Character literal for the TAB character
?\C-x    # Character literal for Ctrl-X
?\111    # Literal for character whose encoding is 0111 (octal)
```

王垠

yinwang.org

yinwang.org


Not Secure — yinwang.org
当然我在扯淡
2019年07月21日
再谈"P vs NP"问题
2019年07月12日
学习的智慧
2018年11月23日
解谜英语语法
2018年04月13日
解谜计算机科学
2018年03月27日
扶门的礼仪
2018年03月08日
从苹果产品的设计错误学教训
2018年02月22日
智能合约和形式验证
2017年11月09日
iPhone X
2017年11月05日

大部分人从学校，从书籍，从文献学知识，结果学到一堆“死知识”。要检验知识是不是死的，很简单。如果你遇到前所未见的问题，却不能把这些知识运用出来解决问题，那么这些知识就很可能是死的。

《学习的智慧》

追根溯源之后，你会发现这知识最初的创造者经过了成百上千的错误。这就像爱迪生发明灯泡，经过了几千次失败的实验。知识的创造者把最后的成功记录在文献里发表，然后你去读它。你以为得到了最宝贵的财富，然而最宝贵的财富却是看不见的。作者从那成百上千的失败中得到的经验教训，才是最宝贵的。而从来没有人把失败写下来发表。

《学习的智慧》

没有这些失败的经验，你就少了所谓“思路”，那你是不大可能从一个知识发展出新的知识的。就像你读了别人的重要 paper，你是不大可能由此发展出重大想法的。你的 paper 会比别人低一个档次，往往只能修修补补，弄出一个小点的想法。而原来的作者以及他的学生们，却可以很容易的变出新的花样，因为他们知道这些路是怎么走过来的，知道许许多多没有写下来的东西。“失败是成功之母”，在我脑子里就是这个意思。

《学习的智慧》

垠神从很早的时候就知道了这个道理，所以他很多时候不看书，不看 paper。

或者只看个开头，知道问题是什么。他看到一个问题，喜欢自己想出解决方案。他不是每次都成功，实际上他为此经历了许许多多的失败。运气好的时候，他得到跟已有成果一样的结果。运气再好一点的时候，他得到更好的结果。但他关心的不只是成功，中间的许多失败对他也是价值重大的。

《学习的智慧》

具体一点，“top-5”是什么意思呢？也就是说对于一张图片，你可以给出 5 个可能的分类，只要其中一个对了就算分类正确。比如图片上本来是汽车，我看到图片，说：

1.“那是苹果？”
2.“哦不对，是杯子？”
3.“还是不对，那是马？”
4.“还是不对，所以是手机？”
5.“居然还是不对，那我最后猜它是汽车！”

《机器与人类视觉能力的差距》

现在某个神经网络（ResNet-152）的 top-5 错误率是 4.49%，它的 top-1 错误率是 19.38%。你却只根据 top-5 得出结论，说神经网络超越了人类。是不是很荒谬?

《机器与人类视觉能力的差距》

这就是“AI 图像识别超越人类”这种说法来的来源。AI 业界所谓“超人类的识别率”，“90+% 的准确率”，全都是用“top-5 准确率”为标准的，而且用来比较的人类识别率的数字没有可靠的来源。等你用“top-1 准确率”或者更加公平的标准，使用客观公正抽选的人类实验者的时候，恐怕就会发现机器的准确率远远不如人类。

《机器与人类视觉能力的差距》

要实现真正的语言理解和视觉理解是非常困难的，可以说是毫无头绪。

一代又一代的神经学家，认知科学家，哲学家，为了弄明白人类“认知”和“理解”到底是怎么回事，已经付出了许多的努力。可是直到现在，对于人类认知和理解的认识都不足以让机器具有真正的理解能力。

《机器与人类视觉能力的差距》

但不要忘记，识别技术不是真的智能，它没有理解能力，不能用在自动驾驶，自动客服，送外卖，保洁阿姨，厨师，发型师，运动员等需要真正“视觉理解”或者“语言理解”能力的领域，更不能期望它们取代教师，程序员，科学家等需要高级知识的工作。机器也没有感情和创造力，不能取代艺术家，作家，电影导演。所有跟你说机器也能有“感情”或者“创造力”的都是忽悠，就像现在的对话系统一样，只是让人以为它们有那些功能，而其实根本就没有。

《机器与人类视觉能力的差距》

你也许会发现，机器学习很适合用来做那些不直观，人看不透，或者看起来很累的领域，比如各种数据分析。实际上那些就是统计学一直以来想解决的问题。可是视觉这种人类和高等动物的日常功能，机器的确非常难以超越。如果机器学习领域放弃对“人类级别智能”的盲目追求，停止拿“超人类视觉”一类的幌子来愚弄大众，各种夸大，那么他们应该能在很多方向做出积极的贡献。

《机器与人类视觉能力的差距》

我的话有什么不同呢？因为我不但告诉你“是什么”，而且我告诉你“为什么”。所以一个人要相信我说的话，他不需要知道我是谁，我也不需要什么地位。这句话有它自己的力量，这就是理性的力量。就像一个数学定理和它的证明，我不需要告诉你这个定理是什么大数学家或者天才提出来的。我只需要告诉你这定理说了什么，然后按部就班把它证明出来。

《理性的力量》

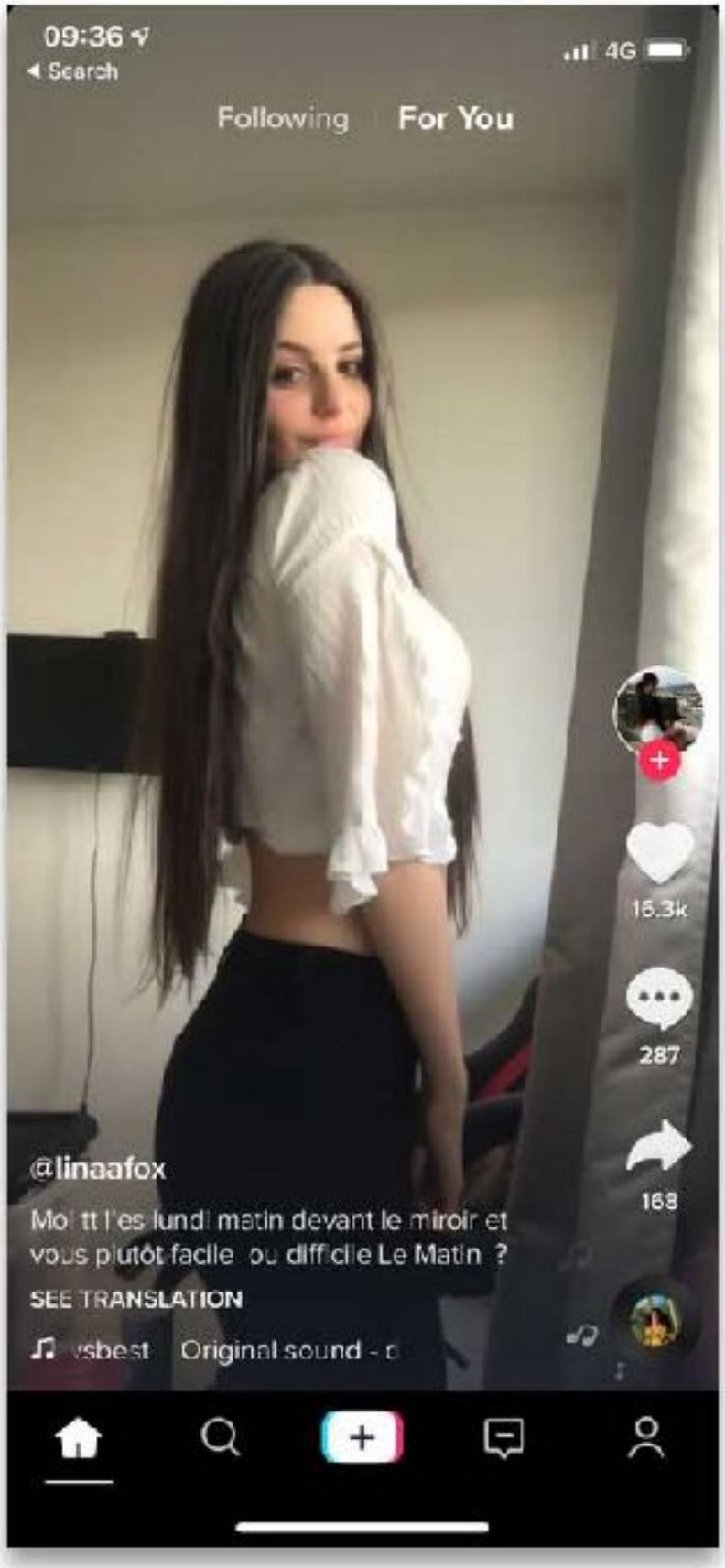
我平时用着Java偷着乐，本来懒得评论其它语言的。可是实在不忍心看着有些人被Scala和Clojure忽悠，所以在这里说几句。如果没有超级高的性能和资源需求（可能要用C这样的低级语言），目前我建议就老老实实用Java吧。虽然不如一些新的语言炫酷，然而实际的系统，还真没有什么是Java写不出来的。少数地方可能需要绕过一些限制，或者放宽一些要求，然而这样的情况不是很多。

《给Java说句公道话》

编程使用什么工具是重要的，然而工具终究不如自己的技术重要。很多人花了太多时间，折腾各种新的语言，希望它们会奇迹一般的改善代码质量，结果最后什么都没做出来。选择语言最重要的条件，应该是“够好用”就可以，因为项目的成功最终是靠人，而不是靠语言。既然Java没有特别大的问题，不会让你没法做好项目，为什么要去试一些不靠谱的新语言呢?

《给Java说句公道话》

英语学习


09:36
Search
4G
Following
For You
16.3k
287
168
@linaafox
Mo tt l'es lundi matin devant le miroir et
vous plutôt facile ou difficile Le Matin ?
SEE TRANSLATION
sbest Original sound - d

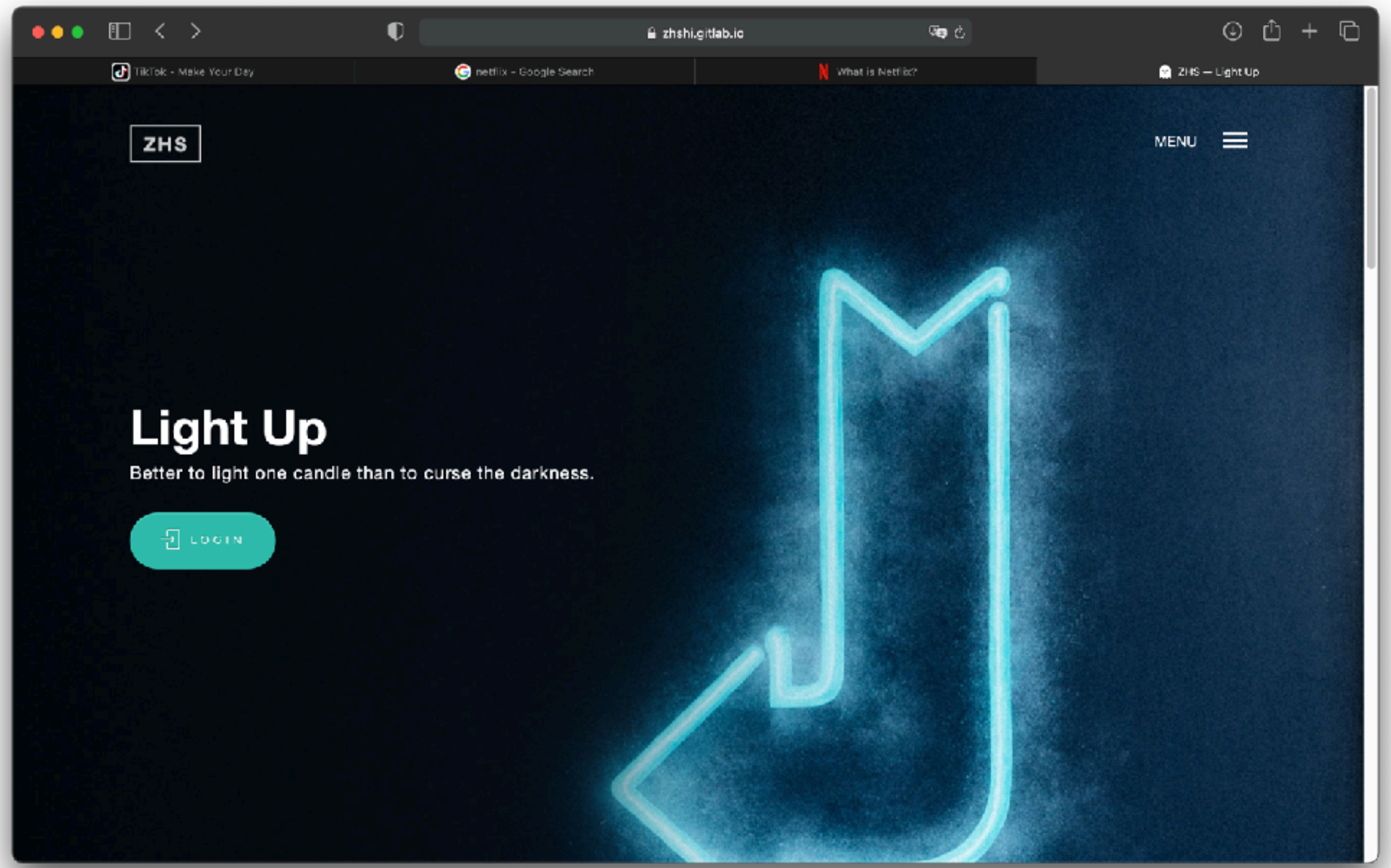

09:37
Search
4G
Following
For You
134.7k
962
1621
@fabianfx
#bratz #makeup #bratzchallenge
SEE TRANSLATION
und - fabianfx Original s


09:46
Search
4G
Following
For You
532.9k
1774
4913
@ilonaaln
MERCI POUR LE MILLION 💜 jpp nos tête
on a tellement eu peur🤣 #fail
#slowmotion @lea_spk ❤️
SEE TRANSLATION
Original sound - itsofficiel

tiktok.com

iOS 添加到桌面


WITCHER
THE CROWN
N SERIES
STRANGER THINGS
Available Now
Strange sightings. Government secrets. Fearless kids.
And a dark force that turns a small town upside down.
Play
More info
Trending Now
NETFLIX
NARCOS



https://zhshi.gitlab.io