Text Functions

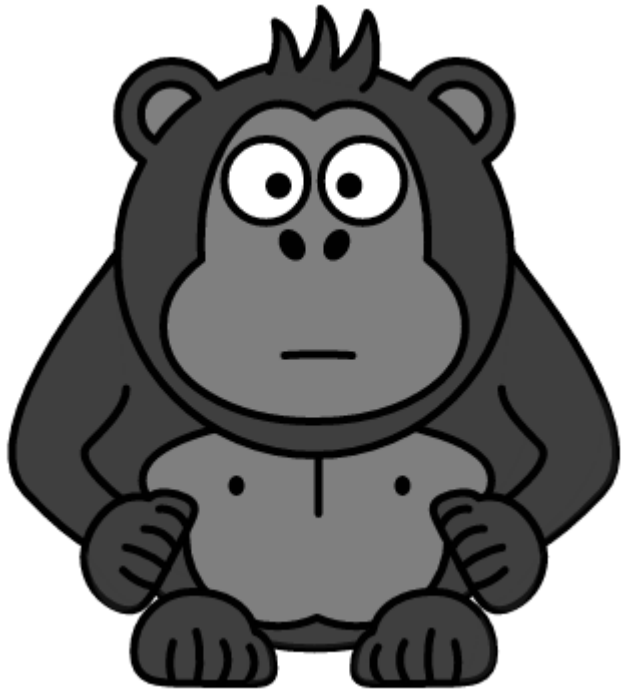
Comparison, Concatenation, Indexing

```
var s = "Gorilla";
```

0	1	2	3	4	5	6
G	o	r	i	l	l	a

Index

Element



Index = an address, plural = indices

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]
a	a	r	d	v	a	r	k	s

Index = an address, plural = indices

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]
a	a	r	d	v	a	r	k	s

Element = a place in the array

Index = an address, plural = indices

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]
a	a	r	d	v	a	r	k	s

Element = a place in the array

What is in element 0?

Index = an address, plural = indices

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]
a	a	r	d	v	a	r	k	s

Element = a place in the array

What is in element 0?

a

Index = an address, plural = indices

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]
a	a	r	d	v	a	r	k	s

Element = a place in the array

What is in element 0?

a

What is the index of v?

Index = an address, plural = indices

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]
a	a	r	d	v	a	r	k	s

Element = a place in the array

What is in element 0?

a

What is the index of v?

4

Index

- The address of an element.
- The indices go from 0 to the array length - 1.

Element

- A value stored at a specific index.
- Only one things is stored in each spot.

1. Use this code to fill in the memory diagram.

```
var a = "atari asteroids";
```

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

- In RAM, text variables are stored with each _____ in a box.
- The _____ are the numbers of the top of the table. For this text variable, they go from _____ to _____.
- The plural of index is _____.
- The length of this text variable is _____.
- Because the index numbers start at _____, the length is _____ bigger than the last index.
- The _____ is the part in the bottom of the table.

1. Use this code to fill in the memory diagram.

```
var a = "atari asteroids";
```

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

- In RAM, text variables are stored with each letter (char) in a box.
- The _____ are the numbers of the top of the table. For this text variable, they go from _____ to _____.
- The plural of index is _____.
- The length of this text variable is _____.
- Because the index numbers start at _____, the length is _____ bigger than the last index.
- The _____ is the part in the bottom of the table.

1. Use this code to fill in the memory diagram.

```
var a = "atari asteroids";
```

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

- In RAM, text variables are stored with each letter (char) in a box.
- The index numbers are the numbers of the top of the table. For this text variable, they go from 0 to 14.
- The plural of index is _____.
- The length of this text variable is _____.
- Because the index numbers start at _____, the length is _____ bigger than the last index.
- The _____ is the part in the bottom of the table.

1. Use this code to fill in the memory diagram.

```
var a = "atari asteroids";
```

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

- In RAM, text variables are stored with each letter (char) in a box.
- The index numbers are the numbers of the top of the table. For this text variable, they go from 0 to 14.
- The plural of index is indices.
- The length of this text variable is _____.
- Because the index numbers start at _____, the length is _____ bigger than the last index.
- The _____ is the part in the bottom of the table.

1. Use this code to fill in the memory diagram.

```
var a = "atari asteroids";
```

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

- In RAM, text variables are stored with each letter (char) in a box.
- The index numbers are the numbers of the top of the table. For this text variable, they go from 0 to 14.
- The plural of index is indices.
- The length of this text variable is 15.
- Because the index numbers start at _____, the length is _____ bigger than the last index.
- The _____ is the part in the bottom of the table.

1. Use this code to fill in the memory diagram.

```
var a = "atari asteroids";
```

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

- In RAM, text variables are stored with each letter (char) in a box.
- The index numbers are the numbers of the top of the table. For this text variable, they go from 0 to 14.
- The plural of index is indices.
- The length of this text variable is 15.
- Because the index numbers start at 0, the length is one bigger than the last index.
- The _____ is the part in the bottom of the table.

1. Use this code to fill in the memory diagram.

```
var a = "atari asteroids";
```

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

- In RAM, text variables are stored with each letter (char) in a box.
- The index numbers are the numbers of the top of the table. For this text variable, they go from 0 to 14.
- The plural of index is indices.
- The length of this text variable is 15.
- Because the index numbers start at 0, the length is one bigger than the last index.
- The element is the part in the bottom of the table.

```
var s = "Gorilla";
```

0	1	2	3	4	5	6
G	o	r	i	l	l	a

```
setText ("answer", s.length ());
```

Prints: 7

```
var s = "Gorilla";
```

0	1	2	3	4	5	6
G	o	r	i	l	l	a

```
setText ("answer", s.toUpperCase ());
```

Prints: GORILLA

```
var s = "Gorilla";
```

0	1	2	3	4	5	6
G	o	r	i	l	l	a

```
setText ("answer", s.toLowerCase ());
```

Prints: gorilla

```
var s = "Gorilla";
```

0	1	2	3	4	5	6
G	o	r	i	l	l	a

```
setText ("answer", s.indexOf("a"));
```

Prints: 6

```
var s = "Gorilla";
```

0	1	2	3	4	5	6
G	o	r	i	l	l	a

```
setText ("answer", s.indexOf("l"));
```

Prints: 4

If two or more, prints the first one.

```
var s = "Gorilla";
```

0	1	2	3	4	5	6
G	o	r	i	l	l	a

```
setText ("answer", s.charAt (1));
```

Prints: o

```
var s = "Gorilla";
```

0	1	2	3	4	5	6
G	o	r	i	l	l	a

Start at the first number

```
setText ("answer", s.substring(2,6));
```

Stops BEFORE the second number

Prints: rill

```
var s = "Gorilla";
```

0	1	2	3	4	5	6
G	o	r	i	l	l	a

```
setText ("answer", s.substring(3,5));
```

Prints: il

```
var s = "Gorilla";
```

0	1	2	3	4	5	6
G	o	r	i	l	l	a

```
setText ("answer", s.substring(1,s.length()));
```

Prints: orilla

You can nest functions

```
var s = "Gorilla";
```

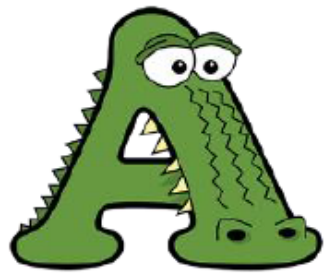
0	1	2	3	4	5	6
G	o	r	i	l	l	a

```
setText ("answer", s.substring(s.indexOf("r"),5);
```

Prints: ril

Boolean Operators:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z



4. Fill in the blanks:

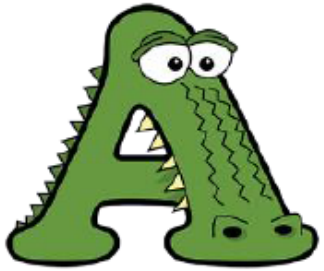
- Text functions have interesting _____ expressions.
- They compare using the _____.
- `Apple` is _____ than `Ball` because _____ is earlier in the alphabet than _____.
- This is used to _____ a list of names and put it in alphabetical order.

5. Use these variables to figure out the Boolean expressions.

```
var word1 = "alligator";     var word3 = "zebra";
var word2 = "bear";          var word4 = "zebra";
```

Boolean Operators:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z



4. Fill in the blanks:

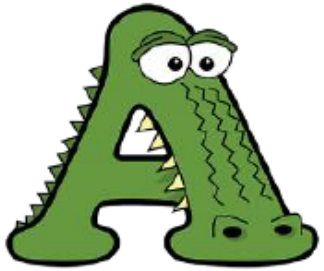
- Text functions have interesting _____Boolean_____ expressions.
- They compare using the _____.
- `Apple` is _____ than `Ball` because _____ is earlier in the alphabet than _____.
- This is used to _____ a list of names and put it in alphabetical order.

5. Use these variables to figure out the Boolean expressions.

```
var word1 = "alligator";     var word3 = "zebra";
var word2 = "bear";          var word4 = "zebra";
```

Boolean Operators:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z



4. Fill in the blanks:

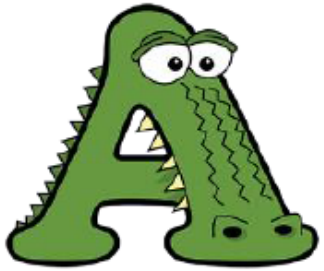
- Text functions have interesting Boolean expressions.
- They compare using the alphabet position of the char.
- `Apple` is _____ than `Ball` because _____ is earlier in the alphabet than _____.
- This is used to _____ a list of names and put it in alphabetical order.

5. Use these variables to figure out the Boolean expressions.

```
var word1 = "alligator";     var word3 = "zebra";
var word2 = "bear";          var word4 = "zebra";
```

Boolean Operators:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z



4. Fill in the blanks:

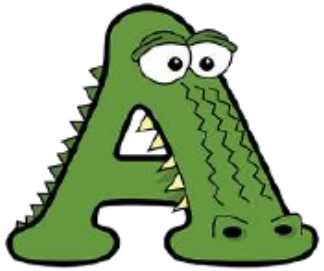
- Text functions have interesting ___Boolean___ expressions.
- They compare using the ___alphabet position of the char___.
- `Apple` is ___smaller___ than `Ball` because ___A___ is earlier in the alphabet than ___B___.
- This is used to _____ a list of names and put it in alphabetical order.

5. Use these variables to figure out the Boolean expressions.

```
var word1 = "alligator";     var word3 = "zebra";
var word2 = "bear";          var word4 = "zebra";
```

Boolean Operators:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z



4. Fill in the blanks:

- Text functions have interesting ___Boolean___ expressions.
- They compare using the ___alphabet position of the char___.
- `Apple` is ___smaller___ than `Ball` because ___A___ is earlier in the alphabet than ___B___.
- This is used to ___sort___ a list of names and put it in alphabetical order.

5. Use these variables to figure out the Boolean expressions.

```
var word1 = "alligator";     var word3 = "zebra";
var word2 = "bear";          var word4 = "zebra";
```

```
var s = "Gorilla";
```

0	1	2	3	4	5	6
G	o	r	i	l	l	a

```
setText ("answer", (s=="Water"));
```

Gorilla == Water

Prints: false

```
var s = "Gorilla";
```

0	1	2	3	4	5	6
G	o	r	i	l	l	a

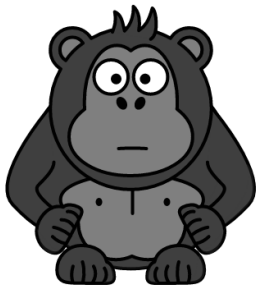
```
setText ("answer",(s=="Gorilla"));
```

Gorilla == Gorilla

Prints: true

```
var s = "Gorilla";
```

0	1	2	3	4	5	6
G	o	r	i	l	l	a



```
setText ("answer",(s=="gorilla"));
```

Gorilla == gorilla

Prints: false

```
var s = "Gorilla";
```

0	1	2	3	4	5	6
G	o	r	i	l	l	a

```
setText ("answer",(s>"Zebra"));
```

Gorilla > Zebra

Prints: false

```
var s = "Gorilla";
```

0	1	2	3	4	5	6
G	o	r	i	l	l	a

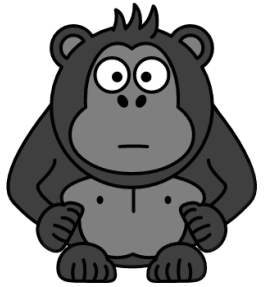
```
setText ("answer",(s<"Zebra")));
```

Gorilla < Zebra

Prints: True

```
var s = "Gorilla";
```

0	1	2	3	4	5	6
G	o	r	i	l	l	a

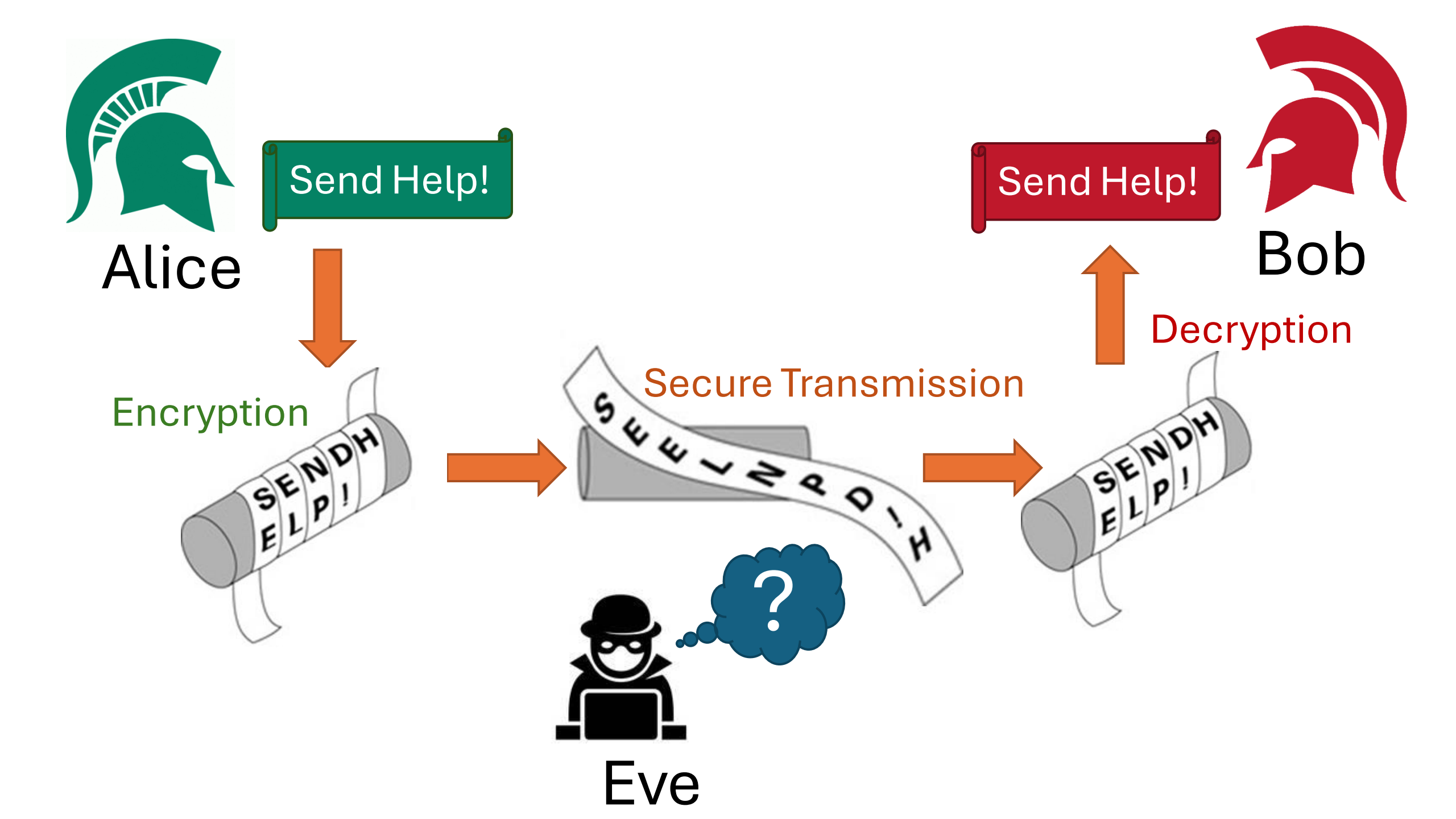


```
setText ("answer",(s<"Apple") < 0));
```

Gorilla < Apple

Prints: False

Spartan Scytale


Alice
Send Help!
Encryption
SENDH
ELP!
Secure Transmission
SELNRD!H
?
Eve
Send Help!
Bob
Decryption
SENDH
ELP!


Pig
Latin

HELP

Plaintext

Alice

Algorithm: First letter to end AND add ay.

HELP … ELP+H … ELPH+AY

Ciphertext

ELPHAY

Bob


Caesar
Shift

Alice

HELP

Plaintext

Key: shift of 1 letter

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
ZABCDEFGHIJKLMNOPQRSTUVWXY
```

Ciphertext

GDKO

Bob

Alice

HELP

Encryption

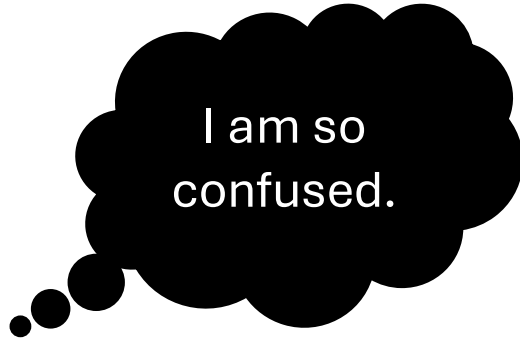
Key: shift of 1 letter

ABCDEFGHIJKLMNOPQRSTUVWXYZ
ZABCDEFGHIJKLMNOPQRSTUVWXY

GDKO

Bob

Eve



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
2	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y
3	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x
4	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w
5	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v
6	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u
7	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t
8	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s
9	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r
10	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q
11	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p
12	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
13	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n
14	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m
15	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l
16	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k
17	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j
18	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i
19	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h
20	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g
21	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f
22	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e
23	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d
24	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c
25	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b
26	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a
27	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
28																										

There are only 26 possible keys… and one of them is very poor.

Why is that key so poor?

Brute Force Attack

Alice

HELP

Encryption

Key: shift of 1 letter

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
ZABCDEFGHIJKLMNOPQRSTUVWXY
```

GDKO

Bob

I know the key. I can decrypt the message.

Eve

I only need to try 25 keys. That's seconds. I can decrypt the message.

Key Distribution Problem

Alice

HELP

I need to pick a key.

Key: shift of 1 letter

Send the key out.

Bob

I know the key. I can decrypt the message.

Eve

Perfect. I will save the key for later.