LET'S FIRST RECAP!

PUZZLE HEIST!

PRESS START

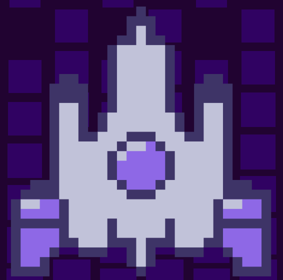
RULES

Each group gets a puzzle sheet with multiple rounds of questions

Solve the puzzle revealing a passcode

YOU HAVE 15 MINS !

ARE YOU READY?



indexing: a numbering of elements/characters begins with 0

5TH LETTER IN WORD = 4TH INDEX

	0	1	2	3	4	5	6	7
Word =	“E	N	V	E	L	O	P	E”

	0	1	2	3	4	5	6	7	8	9	10	11	12	13
Band =	“G	U	N	S		A	N	D		R	O	S	E	S”

Accessing letters and slicing

Band [5]= ‘A’	
Band [4]= ‘ ’	SPACES ARE ALSO CHARACTERS
Band [0]= ‘G’	FIRST CHAR IN THE STRING
Band [-1]= ‘S’	LAST CHAR IN THE STRING

Band [5 : 8]
Band [: 5]
Band [9 :]

?

IN BUILT STRING FUNCTIONS

(functions you can use without having created them yourself)

Where might we use these? What datatype values do you think they return?

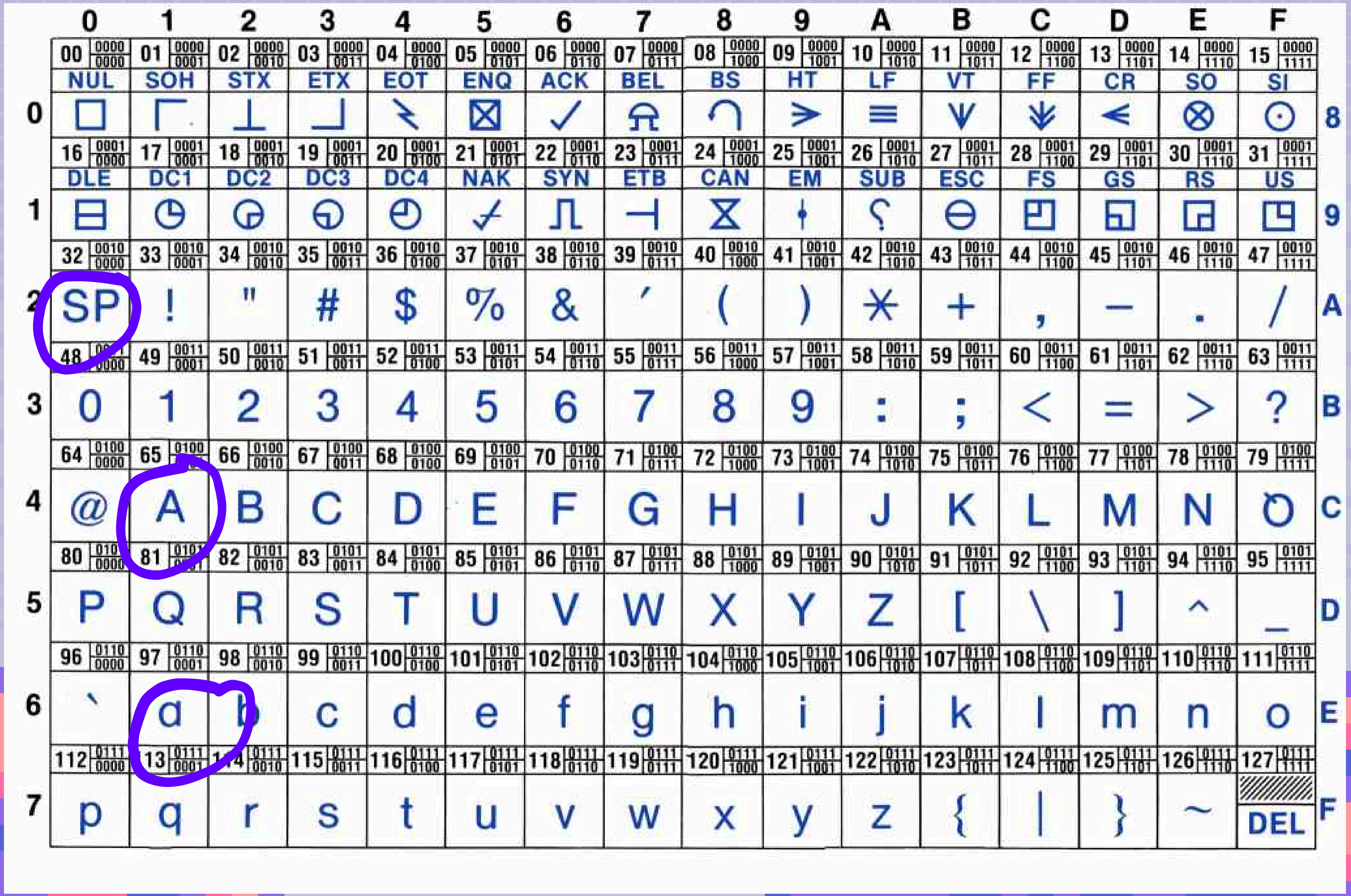
text.lower()	Converts all characters to lowercase
text.upper()	Converts all characters to uppercase
text.find('cloud')	Checks if the string 'cloud' is present in string text
text.count('or')	Counts number of times 'or' is present in string text
text.isalpha()	Checks if all of a string's characters are alphabets
text.isdigit()	Checks if all of a string's characters are digits

ASCII Codes

Numeric representation of characters such as digits, letters, & symbols

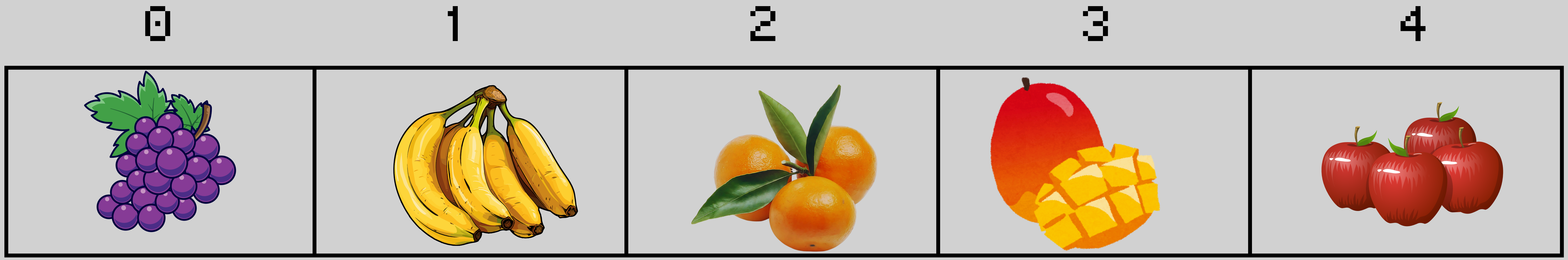
ascii_code= ord(‘A’)
chr(65)= ‘A’

why might this be useful?



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0	00 0000 0000 NUL	01 0000 0001 SOH	02 0000 0010 STX	03 0000 0011 ETX	04 0000 0100 EOT	05 0000 0101 ENQ	06 0000 0110 ACK	07 0000 0111 BEL	08 0000 1000 BS	09 0000 1001 HT	10 0000 1010 LF	11 0000 1011 VT	12 0000 1100 FF	13 0000 1101 CR	14 0000 1110 SO	15 0000 1111 SI	8
1	16 0001 0000 DLE	17 0001 0001 DC1	18 0001 0010 DC2	19 0001 0011 DC3	20 0001 0100 DC4	21 0001 0101 NAK	22 0001 0110 SYN	23 0001 0111 ETB	24 0001 1000 CAN	25 0001 1001 EM	26 0001 1010 SUB	27 0001 1011 ESC	28 0001 1100 FS	29 0001 1101 GS	30 0001 1110 RS	31 0001 1111 US	9
2	32 0010 0000 SP	33 0010 0001 !	34 0010 0010 "	35 0010 0011 #	36 0010 0100 $	37 0010 0101 %	38 0010 0110 &	39 0010 0111 '	40 0010 1000 (	41 0010 1001)	42 0010 1010 *	43 0010 1011 +	44 0010 1100 ,	45 0010 1101 -	46 0010 1110 .	47 0010 1111 /	A
3	48 0011 0000 0	49 0011 0001 1	50 0011 0010 2	51 0011 0011 3	52 0011 0100 4	53 0011 0101 5	54 0011 0110 6	55 0011 0111 7	56 0011 1000 8	57 0011 1001 9	58 0011 1010 :	59 0011 1011 ;	60 0011 1100 <	61 0011 1101 =	62 0011 1110 >	63 0011 1111 ?	B
4	64 0100 0000 @	65 0100 0001 A	66 0100 0010 B	67 0100 0011 C	68 0100 0100 D	69 0100 0101 E	70 0100 0110 F	71 0100 0111 G	72 0100 1000 H	73 0100 1001 I	74 0100 1010 J	75 0100 1011 K	76 0100 1100 L	77 0100 1101 M	78 0100 1110 N	79 0100 1111 O	C
5	80 0101 0000 P	81 0101 0001 Q	82 0101 0010 R	83 0101 0011 S	84 0101 0100 T	85 0101 0101 U	86 0101 0110 V	87 0101 0111 W	88 0101 1000 X	89 0101 1001 Y	90 0101 1010 Z	91 0101 1011 [	92 0101 1100 \	93 0101 1101]	94 0101 1110 ^	95 0101 1111 _	D
6	96 0110 0000 `	97 0110 0001 a	98 0110 0010 b	99 0110 0011 c	100 0110 0100 d	101 0110 0101 e	102 0110 0110 f	103 0110 0111 g	104 0110 1000 h	105 0110 1001 i	106 0110 1010 j	107 0110 1011 k	108 0110 1100 l	109 0110 1101 m	110 0110 1110 n	111 0110 1111 o	E
7	112 0111 0000 p	113 0111 0001 q	114 0111 0010 r	115 0111 0011 s	116 0111 0100 t	117 0111 0101 u	118 0111 0110 v	119 0111 0111 w	120 0111 1000 x	121 0111 1001 y	122 0111 1010 z	123 0111 1011 {	124 0111 1100 |	125 0111 1101 }	126 0111 1110 ~	127 0111 1111 DEL	F

Arrays



fruits = [“apples”, “bananas”, “oranges”, “mangoes”, “grapes”]

accessing elements

fruits [0], fruits [1], etc.

follows the similar indexing starting from 0 as in strings

How might we perform a swap of elements? say we want to swap the first and last element.

```
temp = fruits[0]
fruits[0]=fruits[-1]
fruits[-1]=temp
```

IN BUILT ARRAY FUNCTIONS

(functions you can use without having created them yourself)

array.append(x)	Adds element x to the array
array.remove(x)	removes the element from the array
array.sort()	Sorts the array in place
array. count(x)	counts the number of times x is present in array
array.index(x)	gives us the index location of first instance of x

YOUR TURN!

SOLVE

Behavioural profiling: Not just what you click, but what you might do (predictive models).

Trying to categorize you based on behavior.

Activity: Call up a three people from class and make them read out a list of ten things they recently searched up that they are ok sharing

What ads would you recommend them, how might they be tagged?

What are some design tricks used to nudge people into consent to tracking?
(aka dark patterns)

Political angle: How can data tracking control political systems or undermine democracy?
cambridge analytica: https://www.youtube.com/watch?v=mrnXv-g4yKU
filter bubble, eli parsier: https://www.youtube.com/watch?v=prx9bxzns3g&t=15s

Economic angle: Data as the “new oil” - how has data been commodified? who are the data brokers??

data brokers: https://www.youtube.com/watch?v=wqn3gR1WTcA