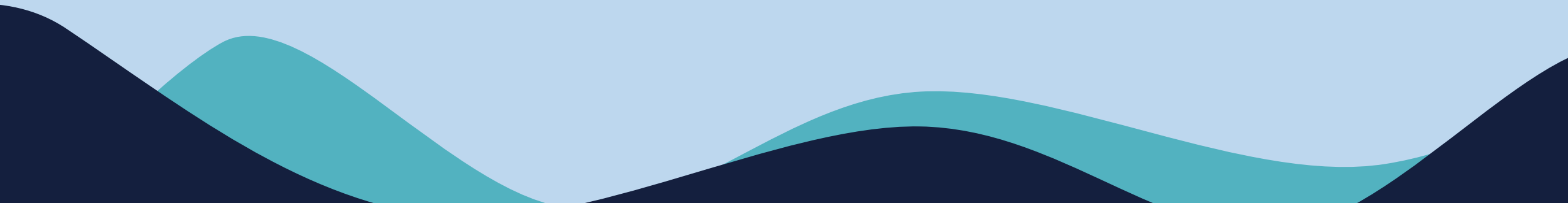


BASIC



BASIC



ภาษาคอมพิวเตอร์

BASIC



ภาษาคอมพิวเตอร์

BINARY 0,1

1

ภาษาเครื่อง

BASIC



ภาษาคอมพิวเตอร์

A B C D รหัสแทนการทำงาน

2

ภาษาระดับต่ำ

BASIC



ภาษาคอมพิวเตอร์

คล้ายภาษามนุษย์ อ่านแล้วเข้าใจ

3

ภาษาระดับสูง

BASIC



ภาษาคอมพิวเตอร์

ภาษาเครื่อง ภาษาระดับต่ำ ภาษาระดับสูง



ภาษาเดียวที่คอมพิวเตอร์เข้าใจ

BASIC



ภาษาคอมพิวเตอร์



BASIC



ภาษาคอมพิวเตอร์

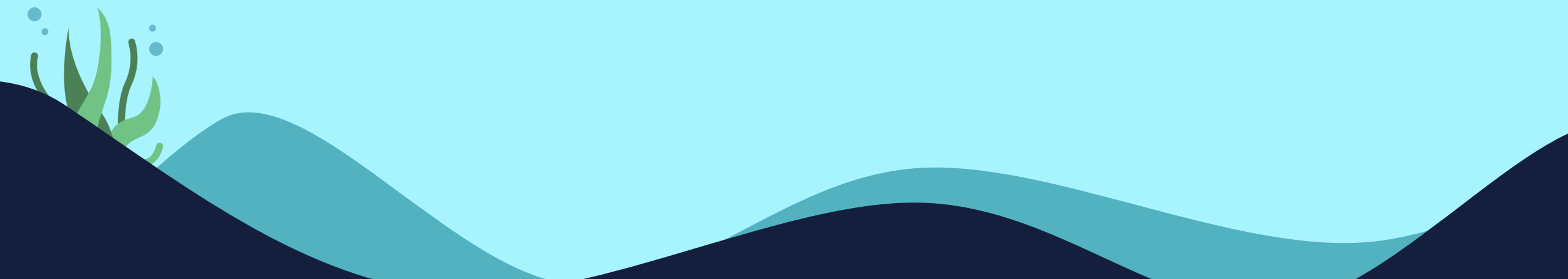
COMPILER



QUIZ !? ភាសា C ខ្ញុំរក្សាបានឬទេ ??



โครงสร้าง



โครงสร้าง

```
#include<stdio.h>

void function() {

}

int main() {
    return 0;
}
```

โครงสร้าง

```
#include<stdio.h>
```

```
void function() {  
}
```

```
int main() {  
    return 0;  
}
```

— **HEADER FILE**

— **FUNCTION**

— **BODY**

HEADER FILE

```
#include<stdio.h>
```

เป็นตัวจัดเก็บชุดคำสั่ง

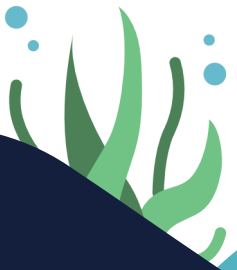
เริ่มต้นด้วย # ทำงานก่อนที่เครื่องจะ compile

HEADER FILE

```
#include<stdio.h>
```

▶ printf();

▶ scanf();



FUNCTION

```
void function() {
```

```
}
```

ประกาศ function เพื่อใช้งาน

BODY

```
int main() {  
    return 0;  
}
```

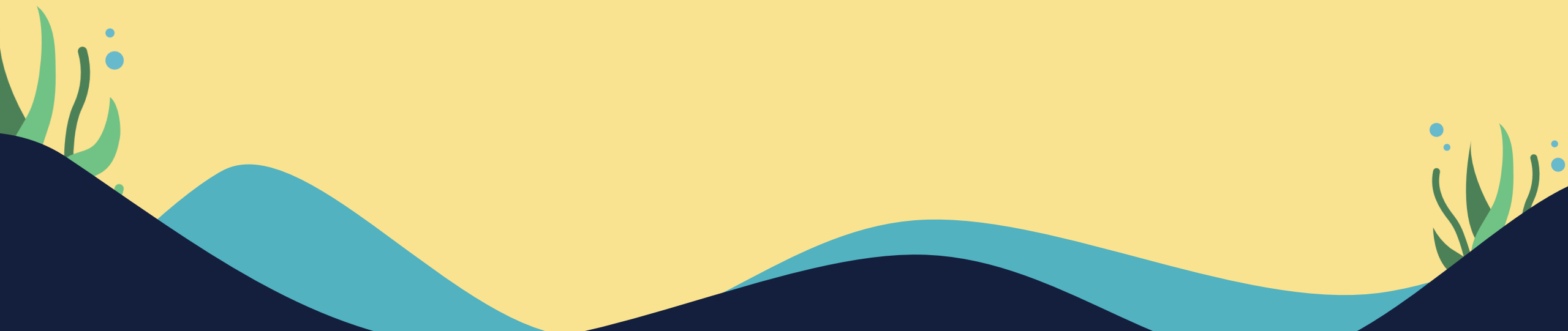
ส่วนหลักของการทำงาน

BODY

```
int main() {  
    return 0;  
}
```

ถ้าไม่มีจะไม่ทำงาน

VARIABLES



VARIABLES

X

VARIABLES

X Y Z A B

VARIABLES

$X = ?$

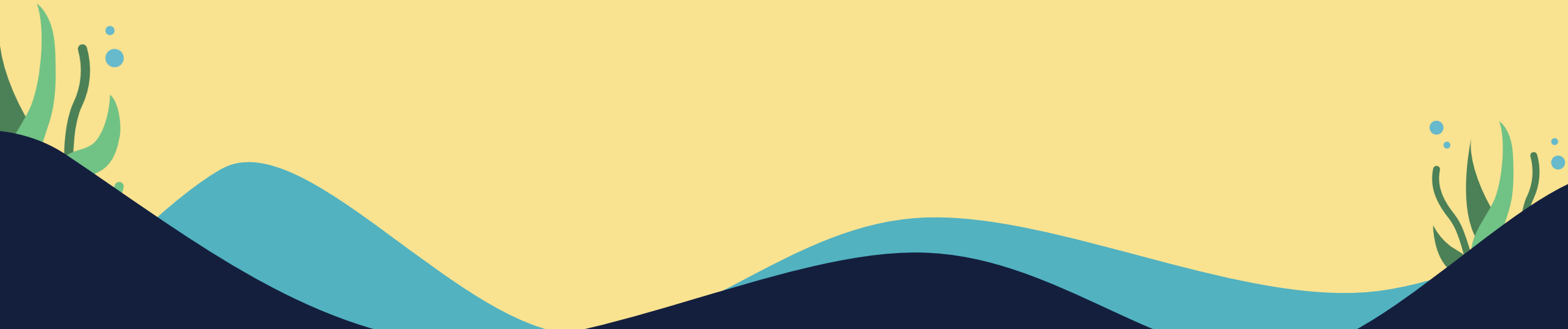
$$5X - 10 = 20$$



VARIABLES

$X = ?$

$$3X + 2(X+2) = 20 - (2X - 5)$$



VARIABLES

$$v = \frac{s}{t}$$

VARIABLES

รถยนต์คันหนึ่งวิ่งด้วยอัตราเร็วเฉลี่ย 80 กิโลเมตรต่อชั่วโมง จากเมือง A ไปยังเมือง B ที่อยู่ห่างกัน 200 กิโลเมตร ถ้าออกเดินทางเวลา 06.00 น. จะถึงปลายทางเวลาเท่าใด

$$v = \frac{S}{t} \rightarrow t = \frac{S}{v}$$

VARIABLES

รถยนต์คันหนึ่งวิ่งด้วยอัตราเร็ว v 80 กิโลเมตรต่อชั่วโมง จากเมือง A ไปยังเมือง B
ที่อยู่ห่างกัน S 200 กิโลเมตร ถ้าออกเดินทางเวลา 06.00 น. จะถึงปลายทางเวลาเท่าใด

$$t = \frac{S}{v} \rightarrow t = \frac{200}{80}$$

VARIABLES

รถยนต์คันหนึ่งวิ่งด้วยอัตราเร็ว v 80 กิโลเมตรต่อชั่วโมง จากเมือง A ไปยังเมือง B
ที่อยู่ห่างกัน S 200 กิโลเมตร ถ้าออกเดินทางเวลา 06.00 น. จะถึงปลายทางเวลาเท่าใด

$$t = \frac{200}{80} \rightarrow t = 2.5$$

VARIABLES

เก็บค่า



VARIABLES

TYPE

ชนิด

ชื่อ

NAME



VARIABLES

TYPE NAME;

TYPE NAME = VALUE;



TYPE ชนิด

VARIABLES

- จำนวนเต็ม
- จำนวนจริง
- ตัวอักษร



TYPE ชนิด

- จำนวนเต็ม

int 16 bits

long 32 bits

VARIABLES

- จำนวนจริง
- ตัวอักษร

TYPE ชนิด

- จำนวนเต็ม

```
int x = 3;
```

```
long y = 5689241;
```

VARIABLES

- จำนวนจริง
- ตัวอักษร

TYPE ชนิด

– จำนวนจริง

float 16 bits

double 32 bits

VARIABLES

– จำนวนเต็ม

– ตัวอักษร

TYPE ชนิด

– จำนวนจริง

```
float a = 2.3;
```

```
double b = 83.56248;
```

VARIABLES

– จำนวนเต็ม

– ตัวอักษร

TYPE ชนิด

– ตัวอักษร

char 8 bits

VARIABLES

- จำนวนเต็ม
- จำนวนจริง

TYPE ชนิด

– ตัวอักษร

```
char plus = '+';
```

VARIABLES

- จำนวนเต็ม
- จำนวนจริง

TYPE ชนิด

– ตัวอักษร (หลายตัว)

VARIABLES

```
char gumLumJai[] = "susso";
```

ARRAY

– จำนวนเต็ม
– จำนวนจริง

unsigned

VARIABLES

ไม่เอาค่าติดลบ

```
unsigned int x = 0;
```

VARIABLES

ชื่อ NAME

การตั้งชื่อตัวแปร

VARIABLES

การตั้งชื่อตัวแปร

ชื่อ NAME

ห้ามขึ้นต้นด้วยตัวเลข	5star , 4you , 1st_variable
ห้ามมีอักขรพิเศษ ยกเว้น _ (underscore)	O945+45 , Hi! , _doctor\$
ห้ามมีช่องว่างในชื่อ	Can I , Is true , love u so much
ตัวพิมพ์เล็ก/ใหญ่ตัวเดียวกัน ถือเป็นตัวอักษรต่างกัน	Dog ≠ dog , X ≠ x
ห้ามตั้งซ้ำกับคำสั่งวน	Int enum goto

VARIABLES

การตั้งชื่อตัวแปร

ชื่อ NAME

auto	else	long	switch
break	enum	register	typedef
case	extern	return	union
char	float	short	unsigned
const	for	signed	void
continue	goto	sizeof	volatile
default	if	static	while
do	int	struct	_Packed
double			

VARIABLES

ชื่อ NAME

Lower Camel Case

VARIABLES

Lower Camel Case

ชื่อ NAME

ตัวแรกตัวเล็ก ถ้ามีคำต่อกันจะขึ้นต้นด้วยตัวใหญ่

Is_True → isTrue



VARIABLES

TYPE

ชนิด

ชื่อ

NAME



ACTIVITY

ประกาศตัวแปรรูกลำ

_Final Desk#3 @gmail docker Static



VARIABLES (เสริม)

การประกาศค่าคงที่



VARIABLES (เสริม)

การประกาศค่าคงที่

เก็บไว้ในตัวแปร
(Memory Constant)

const

const type name = value;

Const float PI = 3.14;

VARIABLES (เสริม)

การประกาศค่าคงที่

นิยามโดยใช้#define

(Define Constant)

การนิยาม, การกำหนด, จำกัดความ

```
#define name value
```

```
#define PI 3.14
```

VARIABLES (เสริม)

การแปลงชนิดข้อมูล



int to float

```
int main() {  
    int in = 9;  
    float out = in;  
}
```

```
int main() {  
    int in = 9;  
    float out = (float)in;  
}
```

Out = 9.000000000000000

int to char

```
int main() {  
    int in = 56;  
    char out = in;  
}
```

```
int main() {  
    int in = 56;  
    char out = (char)in;  
}
```

Out = '8'

ASCII TABLE

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	\$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[ENG OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]

int to char[]

snprintf();

```
int main() {  
    int in = 12345;  
    char out[5];  
    snprintf(out, 5, "%d", in);  
    return(0);  
}
```

sprintf();

```
int main() {  
    int in = 12345;  
    char out[5];  
    sprintf(out, "%d", in);  
    return(0);  
}
```

Out = "12345"

float to int

```
int main(){  
    float in = 3.85f;  
    int out = in;  
}
```

```
int main(){  
    float in = 3.85f;  
    int out = (int)in;  
}
```

Out = 3

float to char[]

snprintf();

sprintf();

เหมือน int to char[]

char to int

```
int main() {  
    const char str[] = "12345";  
    char c = 's';  
    int x, y;  
  
    sscanf(str, "%d", &x); // Using sscanf  
    printf("\nThe value of x : %d", x);  
  
    y = (int)(c); // Using typecasting  
    printf("\nThe value of z : %d", y);  
  
    return 0;  
}
```

VARIABLES (เสริม)

ตัวแปรแบบ Global Local



VARIABLES (เสริม)

ตัวแปรแบบ Global Local

Global นอก function

Local ใน function

```
#include<stdio.h>
```

```
int x = 3;
```

```
int main(){  
    int y = 7;  
    return 0;  
}
```

