3บทความ สอนใช้งาน Arduino ตัวแปรใน Arduino

ตัวแปร	ขนาดของตัวแปร	ค่าของตัวแปร	การประกาศค่า
boolean	1 bit	true,false	boolean a = true;
byte	8 bit	0 - 255	byte a = 10;
char	8 bit	ASCII	char a= 'A';
int	16 bit	-32,768 ถึง 32,767	int a = 1000;
unsigned int	16 bit	0 ถึง 65,535	unsigned int a = 13;
float	32 bit	-3.4028235E+38 ถึง 3.4028235E+38.	float a =10.77;

ตัวแปร หมายถึง กลุ่มของ ตัวอักษร ตัวเลข และ เครื่องหมายใดๆ ที่รวมกันเป็นชื่อ เพื่อใช้กำหนด เป็นตัวแทนของค่าข้อมูลที่เราต้องการจะอ้างถึงในโปรแกรม ทั้งนี้ก็เนื่องจากว่าในการทำงานของโปรแกรมจริงๆนั้นจะใช้ค่าตัวเลขที่ผู้ใช้กำหนดให้ มาทำการประมวลผล ซึ่งในการเขียนโปรแกรมถ้าเราต้องเขียนโปรแกรมโดยกำหนดเป็นค่าตัวเลขให้กับโปรแกรมตรงๆเลย ก็จะ
ทำให้โปรแกรมที่เราเขียนขึ้นเต็มไปด้วยค่า ตัวเลขต่างๆมากมาย ซึ่งยากต่อการอ่าน ยากต่อการทำความเข้าใจ และยากต่อการตรวจสอบความถูกต้อง และอาจทำให้เกิดความผิดพลาดได้ง่ายด้วย ดังนั้นทุกภาษา จึงยอมให้มีการกำหนดชื่อ ขึ้นมาใช้งานแทน ค่าตัวเลข เพื่อให้เขียนโปรแกรมได้สะดวกและง่ายต่อการอ่าน ทำความเข้าใจ ได้มากยิ่งขึ้น

ซึ่งลักษณะของข้อมูล อาจมีทั้งแบบที่เป็นค่าซึ่งสามารถเปลี่ยนแปลงได้ (variable) หรือ อาจเป็นแบบที่มีค่าคงที่ไม่สามารถเปลี่ยนแปลงได้ (constant) ในการประกาศใช้งานตัวแปร จำเป็นต้องประกาศชนิดของตัวแปร หรือบางครั้งอาจมีการกำหนดค่าเริ่มต้นให้กับตัวแปรด้วยก็ได้

ชนิดและประเภทของตัวแปร

ถ้าหากว่าเราจะเปรียบเทียบว่า ตัวแปร คือ ภาชนะสำหรับบรรจุสิ่งของ และ ข้อมูล คือ สิ่งของที่เราต้องการจะเก็บ จะเห็นได้ว่า สิ่งของต่างๆรอบๆตัวเรานั้น จะมีคุณสมบัติที่แตกต่างกันไป ดังนั้นในการเลือกภาชนะสำหรับใช้บรรจุสิ่งของ เราก็
จำเป็นต้องเลือกชนิดของภาชนะให้มีความเหมาะสมที่จะใช้เก็บสิ่งของ ด้วย ซึ่งสิ่งแรกที่ต้องพิจารณาคือ เราจะต้องรู้จักคุณสมบัติของสิ่งของที่ต้องการจะจัดเก็บ และ จุดประสงค์ การใช้งาน ก่อน จากนั้นจึงจัดหา

ภาชนะที่มี ขนาด และ รูปทรงของภาชนะ เหมาะสมที่จะใช้เก็บสิ่งของ เพื่อให้สามารถจัดเก็บ และนำสิ่งของออกมาใช้งานได้อย่างเขประหยัดมากที่สุด

ในภาษาซีนั้น มีการกำหนด และ จำแนก ชนิดของตัวแปร ไว้เป็น 5 ชนิดด้วยกัน โดยแต่ละชนิดจะมีคุณสมบัติการใช้งานที่ต่างกัน เพื่อใช้ในการเก็บข้อมูลที่มีรูปแบบแตกต่างกัน คือ

char ใช้เก็บข้อมูลที่เป็นตัวอักษร (character) ใช้เก็บข้อมูลที่เป็นเลขจำนวนเต็มได้ 256 ค่า

int ใช้เก็บข้อมูลที่เป็นเลขจำนวนเต็ม (integer) ใช้เก็บข้อมูลที่เป็นเลขจำนวนเต็มได้ 65536 ค่า

float ใช้เก็บข้อมูลที่เป็นเลขทศนิยมแบบ Single Precision

double ใช้เก็บข้อมูลที่เป็นเลขทศนิยมแบบ Double Precision ซึ่ง สามารถเก็บค่าตัวเลขทศนิยมที่มีความละเอียดและถูกต้องของทศนิยมมากว่าแบบ float ถึง 2 เท่า

void ใช้เก็บตัวแปรที่ไม่มีค่า

คุณสมบัติเฉพาะของตัวแปร

สำหรับตัวแปรชนิดที่ใช้เก็บค่าเลขจำนวนเต็ม (char และ int) นั้น ในภาษาซี ไม่ได้มีการจำแนก ชนิดของตัวแปรเพื่อใช้เก็บค่าตัวเลขที่เป็น ค่าบวก หรือ ค่าลบ เป็นการเฉพาะ แต่ภาษาซี จะใช้วิธีการเพิ่มคำสั่ง สำหรับกำหนดคุณสมบัติเฉาพะให้กับตัวแปรไว้อีก 4 คำสั่ง
สำหรับใช้กำหนดคุณสมบัติของตัวแปรแบบนี้ ให้มีคุณสมบัติที่เฉพาะเจาะจงลงไปอีกเป็นต้นว่า จะใช้ตัวแปรในการเก็บค่าตัวเลขที่เป็นค่าบวกอย่างเดียว หรือต้องการเก็บค่าตัวเลขแบบคิดเครื่องหมายด้วย เพื่อให้ผู้ใช้สามารถปรับแก้คุณสมบัติในการใช้งานของ ตัวแปรให้มีคุณสมบัติใกล้เคียงกับความต้องการใช้งานมากขึ้นไปอีก และเพื่อจำกัดขอบเขตการใช้งานของ ตัวแปรให้ตรงกับจุดประสงค์มากยิ่งขึ้น และยังเป็นการช่วยให้ประหยัดจำนวนของหน่วยความจำที่ใช้สร้าง ตัวแปรด้วยและ
ทำให้โปรแกรมทำงานได้เร็วขึ้นกว่าเดิมอีกด้วย โดยในภาษาซีจะมีคำสั่ง ที่ใช้สำหรับระบุ คุณสมบัติเฉพาะของตัวแปรที่ใช้เก็บค่าเลขจำนวนเต็ม 4 คำสั่ง คือ

unsigned ใช้ระบุให้เก็บค่าเลขจำนวนเต็มในตัวแปรเฉพาะค่าที่เป็นบวกเท่านั้น

signed ใช้ระบุให้เก็บค่าเลขจำนวนเต็มในตัวแปรทั้งค่า บวก และ ลบ

short ใช้ระบุให้เก็บค่าเลขจำนวนเต็มในตัวแปรที่มีค่าน้อยกว่า int

long ใช้ระบุให้เก็บค่าเลขจำนวนเต็มในตัวแปรที่มีค่ามากกว่า int เป็น 2 เท่า

ชนิดข้อมูล	การเก็บข้อมูล	ขนาด
boolean	จริง (True) หรือ เท็จ (False)	1 บิต
char	ตัวเลข หรือตัวอักษร	1 ไบต์ ใส่ค่าได้ตั้งแต่ -128 ถึง 127
unsigned char	ตัวเลข หรือตัวอักษร	1 ไบต์ ใส่ค่าได้ตั้งแต่ 0 ถึง 255
byte	ไบต์	1 ไบต์ ใส่ค่าได้ตั้งแต่ 0 ถึง 255
int	ตัวเลขจำนวนเต็ม	2 ไบต์ ใส่ค่าได้ตั้งแต่ -32,768 ถึง 32,767
unsigned int	ตัวเลขจำนวนเต็ม	2 ไบต์ ใส่ค่าได้ตั้งแต่ 0 ถึง 65,535 (2^16) - 1)
long	ตัวเลขจำนวนเต็มที่มีความยาว	4 ไบต์ ใส่ค่าได้ตั้งแต่ -2,147,483,648 ถึง 2,147,483,647
unsigned long	ตัวเลขจำนวนเต็มที่มีความยาว	4 ไบต์ ใส่ค่าได้ตั้งแต่ 0 ถึง 4,294,967,295 (2^32 - 1)
float	ตัวเลขทศนิยมใช้ในการคำนวน	4 ไบต์ ใส่ค่าได้ตั้งแต่ 3.4028235E+38 ถึง -3.4028235E+38 มีทศนิยมได้ 6 ถึง 7 ตำแหน่ง
double (เฉพาะบอร์ด Arduino Due)	ตัวเลขทศนิยมที่มีความยาวและต้องการความแม่นยำ	8 ไบต์ ใช้ในการคำนวนที่ต้องการประสิทธิ์ภาพสูง
String	ข้อความ	ไม่ระบุ

การประกาศตัวแปร เซ็ตค่าตัวแปร

การประกาศตัวแปรจะเหมือนกับภาษา C โดยปกติ คือ

TYPE KEY;

โดย TYPE เป็นชนิดของข้อมูล ส่วน KEY เป็นชื่อตัวแปร การประกาศตัวแปรข้างต้นคือการประกาศตัวแปรแบบไม่กำหนดค่า ดังนั้นค่าปกติที่อ่านจากตัวแปรจะเป็น 0

TYPE KEY = VAL;

จากตัวอย่างด้านบน เป็นลักษณะของการประกาศตัวแปรแบบกำหนดค่า เมื่ออ่านค่าของตัวแปรออกมาจะได้เป็นค่าที่กำหนดไว้ตอนประกาศ

ตัวอย่างการประกาศตัวแปร

int i;

int a = 10, b = 20;

จากตัวอย่าง จะเห็นว่าเราสามารถกำหนดชนิดของข้อมูลให้ทีเดียวหลายๆตัวแปรก็ได้ โดยใช้เครื่องหมาย , คั่นไว้

```
int i;
```

```
int a = 10, b = 20;
```

```
i = a + b;
```

จากตัวอย่างด้านบน เราสามารถกำหนดค่าให้กับตัวแปรเมื่อไรก็ได้ โดยใช้เครื่องหมายเท่ากับ = เป็นตัวเชื่อม ชื่อตัวแปรจะอยู่ทางซ้าย และจะกำหนดค่าเป็นอะไร ให้อยู่ทางขวา ค่าที่อยู่ทางขวาจะถูกนำมาใส่ในค่าที่อยู่ทางซ้ายเสมอ

```
int i = 10, a;
```

```
a = i;
```

จากตัวอย่างด้านบน จะเห็นว่า a ไม่ได้กำหนดค่าไว้ตอนประกาศ ทำให้ค่าที่อ่านได้จาก a คือ 0 แต่บรรทัดถัดมา มีการกำหนดค่าให้ a เท่ากับ i ซึ่งตอนประกาศ i ได้ประกาศไว้ว่าค่าเท่ากับ 10 เมื่อนำมาใส่ a ค่าที่อ่านได้จาก a จึงเป็น 10 ด้วยเช่นกัน

```
boolean is = false;
```

```
is = !is;
```

จากตัวอย่างด้านบน มีการประกาศตัวแปร boolean ซึ่งเป็นตัวแปรทางลอจิก มีค่าเป็น True (1) หรือ False (0) ได้เท่านั้น ในบรรทัดแรกได้ประกาศว่าตัวแปร is เป็นตัวแปรชนิด boolean และมีค่าเป็น false หรือลอจิก 0 บรรทัดต่อมา ได้มีการกำหนดให้ is เท่ากับ !is การที่เครื่องหมายนิเสธไปอยู่หน้าตัวแปร หมายถึงการกลับเป็นค่าตรงข้าม จากบรรทัดแรก ตัวแปร is มีค่าเป็น false เมื่อเจอ !is ค่าจึงถูกกลับเป็น true และถูกนำไปเซ็ตในตัวแปร is ทำให้สุดท้ายแล้วตัวแปร is มีค่าเป็น true

```
String text = "Myarduino";
```

จากตัวอย่างด้านบน ได้มีการประกาศตัวแปรชื่อ text เป็นชนิด String เมื่ออ่านค่าที่ได้จาก text จึงได้ค่าออกมาเป็น "Myarduino" เลย

* การกำหนดค่าแบบข้อความให้กับตัวแปร จะต้องอยู่ภายใต้เครื่องหมาย "" เท่านั้น มิฉนั้นโปรแกรมจะแสดงข้อความผิดพลาดออกมา

สรุปชนิดของตัวแปรใน Arduino ที่ใช้บ่อยๆ

boolean ใช้เก็บค่าข้อมูล เพียง 2 จำนวน คือ TRUE (จริง) และ FALSE (เท็จ)

char ใช้เก็บค่าข้อมูลขนาด 8 บิต ใช้สำหรับเก็บค่ารหัสของตัวอักษร ซึ่งสามารถกำหนดเป็นค่า หรือ เขียนตัวอักษรไว้ภายใต้เครื่องหมาย ฟันเดียวก็ได้ เช่น 'A' หรือ 0x41 หรือ 65

byte ใช้เก็บค่าข้อมูลขนาด 8 บิตที่เป็นค่าจำนวนเต็มแบบไม่คิดเครื่องหมาย เหมือนกันกับunsigned char ในภาษาซี) ซึ่งสามารถเก็บค่าข้อมูลได้ 256 ค่า คือ 0-255

int หรือ Integer ใช้เก็บค่าข้อมูลขนาด 16บิต ที่เป็นค่าจำนวนเต็ม แบบคิดเครื่องหมาย โดยสามารถใช้เก็บข้อมูลได้ 65536 ค่า คือ -32768 ถึง +32767

unsigned int ใช้เก็บค่าข้อมูลขนาด 16บิต ที่เป็นค่าจำนวนเต็ม แบบไม่คิดเครื่องหมาย โดยสามารถใช้เก็บข้อมูลได้ 65536 ค่า คือ 0-65535

long ใช้เก็บค่าข้อมูลขนาด 32บิต ที่เป็นค่าเลขจำนวนเต็มแบบคิดเครื่องหมาย โดยสามารถใช้เก็บข้อมูลได้ 4294967296 ค่า คือ -2,147,483,648 ถึง 2,147,483,647

unsigned long ใช้เก็บค่าข้อมูลขนาด 32 บิต ที่เป็นค่าเลขจำนวนเต็มแบบไม่คิดเครื่องหมาย โดยสามารถใช้เก็บข้อมูลได้ 4294967296 ค่า คือ 0 ถึง 4,294,967,295

float ใช้เก็บค่าข้อมูลที่เป็นเลขทศนิยมแบบคิดเครื่องหมายขนาด 32 บิต โดยสามารถเก็บค่าได้ ระหว่าง 3.4E-38 ถึง 3.4E+38 (-3.4028235E+38 ถึง 3.4028235E+38)

double ใช้เก็บค่า เลขทศนิยมเช่นเดียวกันกับ float แต่มีค่าความละเอียดกว่า float ถึง 2 เท่า สามารถเก็บค่าได้มากถึง 1.7E+308

void เป็นตัวแปรแบบที่ไม่มีการเก็บค่าใดๆ คือ ไม่มีค่านั้นเอง

arrays เป็นตัวแปรที่ใช้เก็บข้อมูลหลายๆค่าไว้ในตัวแปรตัวเพียงชื่อเดียว แต่มีตัวเลขสำหรับชี้ตำแหน่งการเก็บข้อมูลต่างกัน โดยตัวเลขที่ใช้ทำหน้าที่เป็นตัวชี้ตำแหน่งของข้อมูล เรียกว่า Index Number โดยค่าลำดับของข้อมูลในตัวแปร array ตำแหน่งแรกจะมีค่าเป็น ศูนย์เสมอ

string เป็นตัวแปรใช้เก็บข้อความ หรือ ตัวอักษรหลายๆตัว ซึ่ง string ก็คือ array ของตัวแปร แบบ char นั่นเอง

pointer เป็นตัวแปรที่ไม่ได้ใช้เก็บข้อมูล แต่ใช้เก็บค่าตำแหน่งแอดเดรสของหน่วยความจำที่ใช้ สร้างเป็นตัวแปรสำหรับเก็บข้อมูล ซึ่งตัวแปรแบบนี้จะใช้ทำหน้าที่เป็นตัวชี้ไปยังตำแหน่งแอดเดรส ของตัวแปรอื่นๆอีกที่หนึ่ง