

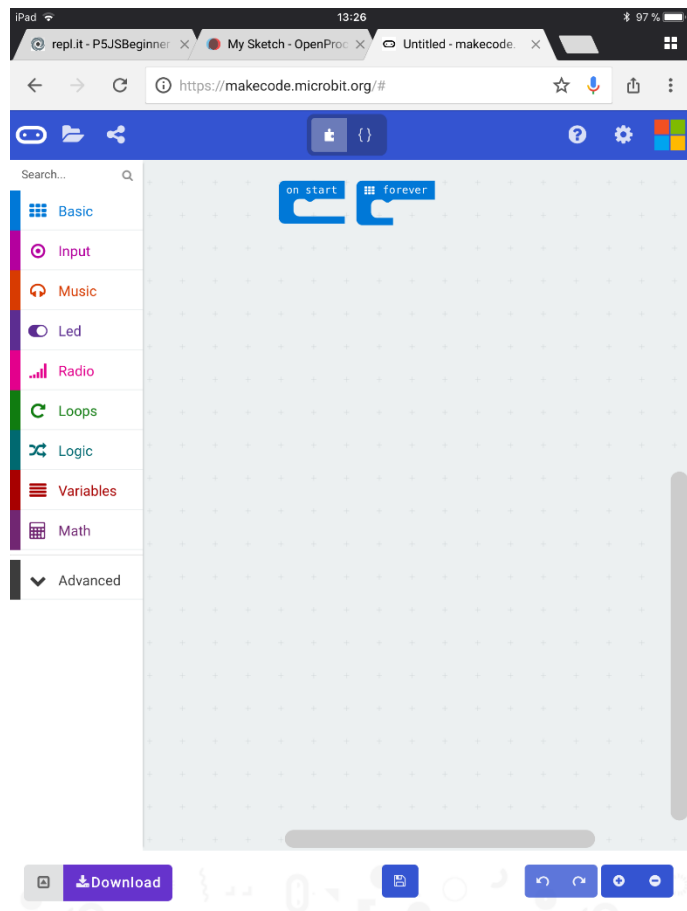
## JavaScript Blocks Editor (PXT)

### Programmets gränssnitt

Programmeringsblocken finns till vänster och är färgkodade. Det grå fältet är där man placerar blocken för att skapa kod.

I blå menyraden finns (från vänster till höger)

- Hemknapp
- Projektmap
- Dela programmet
- Symbolen för block programmering
- Symbolen för JavaScript kod
- Hjälp
- Inställningar
- Länk till Microsoft makecode.org



Längst ner finns (från vänster till höger)

- en pil som gör att man får fram simulatören.
- nedladdningsknappen
- spara ikonen (namnge projektet där)
- ångra och gör om
- zooma in/ut

## Block i *Basic*

I Basic finns bland andra block som styr LED displayen, skriver text och siffror. Det finns en **More knapp** när Basic klickas på. Där finns ytterligare block.

När programmet startas finns två block i programmeringsfältet.

**on start** används när något ska göras vid start och bara en gång.

**forever** används när man vill att något ska göras om och om igen.

**show number** används för att skriva heltal på displayen. Decimaltal fungerar inte.

**show icon** har ett antal förprogrammerade ikoner som man kan visa

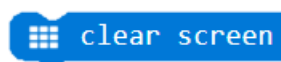
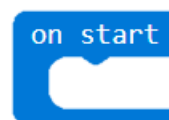
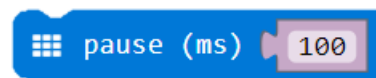
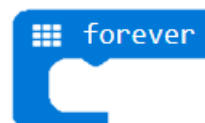
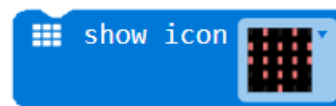
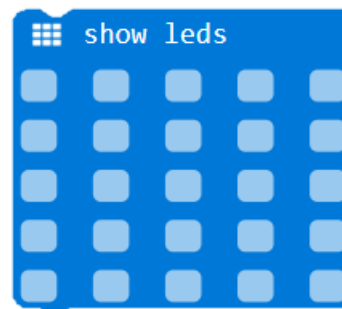
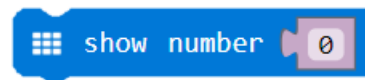
**show led** används när man vill skapa en egen figur på led displayen.

**show string** gör så att text scollar över displayen. Bokstäverna å, ä och ö kommer inte att synas utan där lämnas en lucka på displayen.

**pause** gör en paus i programmet innan nästa block körs. Enheten är millisekunder

**clear screen** tar bort allt från displayen (släcker alla LED)

**show arrow** visar pil för olika väderstreck. Den lilla pilen bredvid North visar att det finns fler val om den klickas.



## Block i Input


Kod som placeras i blocken med den här formen  kommer bara att köras när det kommer vald input.

EX:

```

on button A pressed
  show string "HEJ!"
  pause (ms) 1000
  show icon [heart icon]
  
```

När knapp A trycks in kommer HEJ! att scrolla över displayen, paus i en sekund, ett hjärta visas.

Blocken med den här  formen används för att ta reda på ett värde. Det kan vara ett numeriskt värde eller ett Boolean värde (true eller false).

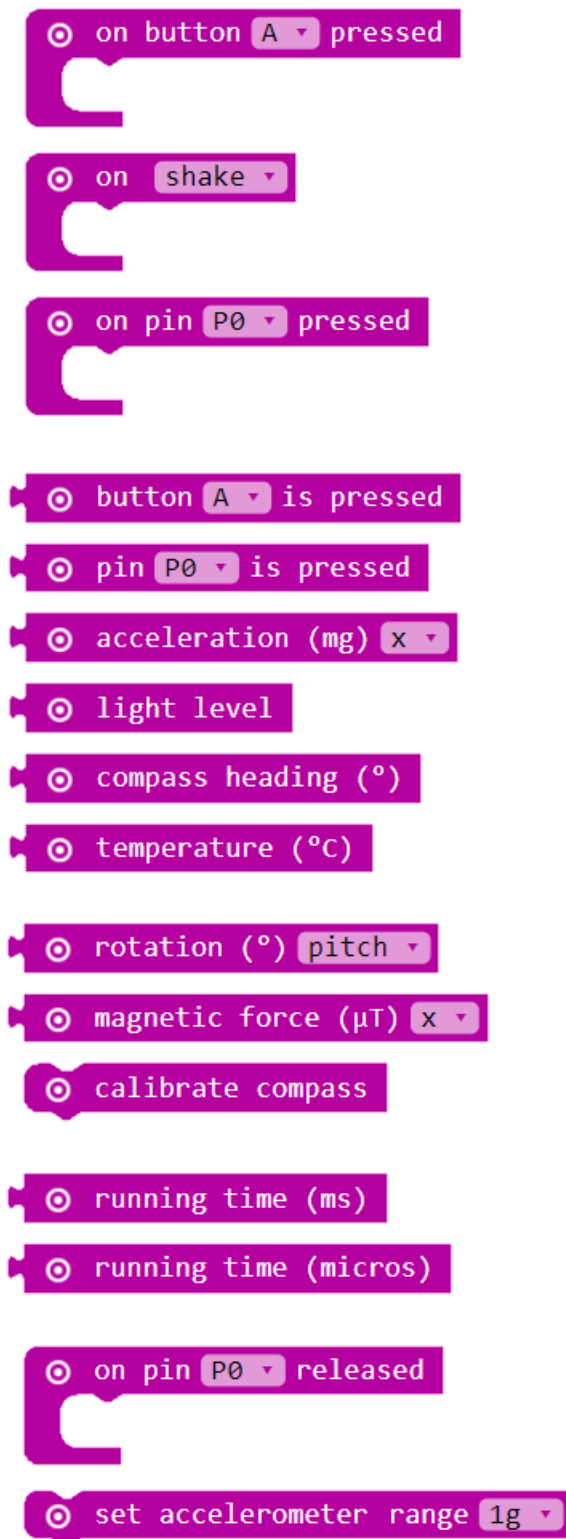
EX:

```

forever
  show number [compass heading (°)]
  pause (ms) 2000
  
```

Visar värdet mellan 0 och 360 beroende av åt vilket håll micro:bit vrids. Pausar 2 sekunder innan värde visas igen.

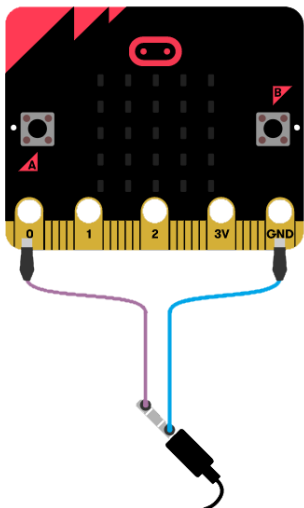
Genom att markera blocket, så det blir en gul rand runt, högerklicka och välja Help kan information fås om vilka värden som returneras.



- on button A pressed
- on shake
- on pin P0 pressed
- button A is pressed
- pin P0 is pressed
- acceleration (mg) x
- light level
- compass heading (°)
- temperature (°C)
- rotation (°) pitch
- magnetic force (µT) x
- calibrate compass
- running time (ms)
- running time (micros)
- on pin P0 released
- set accelerometer range 1g

## Block i Music

För att spela musik kopplas högtalare till pin 0 och GND på microbit.



```

play tone Middle C for 1 beat
ring tone (Hz) Middle C
rest(ms) 1 beat
start melody dadadum repeating once
music on melody note played
Middle C
1 beat
tempo (bpm)
change tempo by (bpm) 20
set tempo to (bpm) 120
    
```

## Block i LED

**plot** och **unplot** tänders respektive släcker en led specificerad av dess x- och y-koordinater. (0,0) är led i övre högra hörnet.

**toggle** betyder växla

**EX:**

```

on button A pressed
  toggle x 0 y 0
  
```

när knapp A trycks in tänds led (0,0) om den är av, när A trycks in igen släcks den, osv.

**point** tar reda på om viss led är av eller på. Returnerar boolean värde (true eller false).

**plot bar graph** betyder stapeldiagram

**EX:**

```

forever
  plot bar graph of light level
  up to 255
  
```

Fler led lyser när yttre ljus ökar. Om det är mörkt i rummet lyser en led och det är full ljusstyrka lyser alla led.

**brightness** är ljusstyrkan på led på displayen i ett intervall 0 – 255 där 0 är av och 255 är full ljusstyrka.

**stop animation** stoppar den animation som pågår på displayen

**EX:**

```

forever
  show string "Hello!"
  on button A pressed
    show number 0
  on button B pressed
    stop animation
    show number 0
  
```

Texten Hello! scollar över displayen för alltid. När knapp A trycks in kommer siffran 0 att visas först när hela textsträngen är klar. När knapp B trycks in kommer siffran 0 att visas direkt oberoende om hela Hello! är visad eller ej.

**led enable** styr om displayen ska vara på eller av.

```

plot x 0 y 0
unplot x 0 y 0
toggle x 0 y 0
point x 0 y 0
plot bar graph of 0
up to 0
plot x 0 y 0 brightness 255
brightness
set brightness 255
stop animation
led enable false
  
```

## Block i Radio

Radio används för att sända information mellan två eller fler micro:bits.

### EX:

Koden nedan laddas upp på två micro:bit. Vid start sätts en grupp

```

on start
  radio set group 156

on button A pressed
  radio send string "hej"

on radio received receivedString
  show icon
  pause (ms) 1000
  show string receivedString
  
```

så att bara de micro:bits som har samma grupp kommer att vara med i sändningen. Värdet kan vara 0—255.

När knapp A trycks ner sänds en text "hej". Nästa block av kod säger att när textsträng tas emot ska en bild visas på displayen, pausa en sekund och visa sen vilken text som mottagits. Blocket

`receivedString` skapas automatiskt i variabler när dessa block används.

Använd markera/högerklicka/help för att läsa mer om respektive block.

```

radio send number 0

radio send value "name" = 0

radio send string ""

on radio received receivedNumber

on radio received name value

on radio received receivedString

radio set group 1

radio set transmit power 7

radio set transmit serial number true

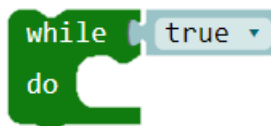
radio write received packet to serial
  
```

## Block i Loops

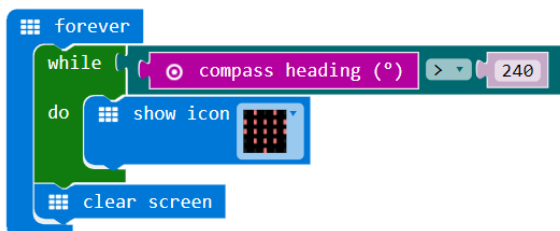
**repeat 4 times** kör koden som läggs i blocket det antal gånger som specificerats



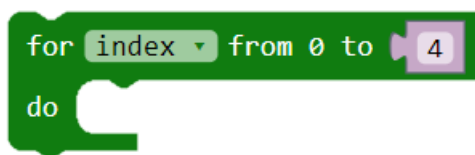
**while do** kör koden som läggs i blocket under förutsättning att det villkor som satts är sant



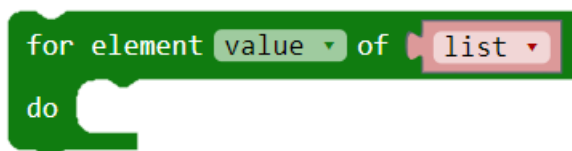
**EX:**



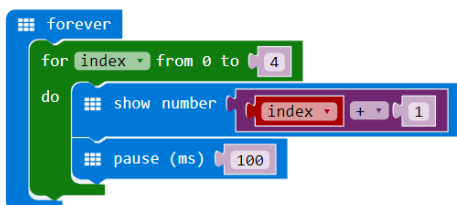
Forever körs hela tiden. While (medans) kompassens riktning är större än 240 grader ska ett hjärta visas på displayen. Om inte ska displayen rensas.



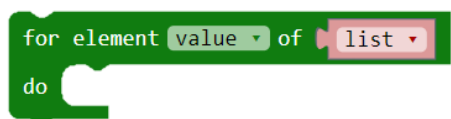
**for index from 0 to 4** står för "så länge variabeln i har ett värde mellan 0 och 4" kör den kod som finns i blocket.



**EX:**



från 0 till 4 gör att det som står i blocket kommer att göras 5 gånger. Ett nummer ska visas som har värdet på index plus 1, pausa 0,1 sekund. Siffrorna som kommer att visas är 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 .....



används för att göra något med varje element i tex en lista.

## Blocken i Logic

**if then** om (det block som placeras bredvid if) är sann kommer det som står i then att utföras. Genom att klicka på kugghjulet kan även else if (annars om) och else (annars) läggas till.

### I Logic finns tecken för olikheter

**$3 < 4$**

Detta utläser vi som "3 är mindre än 4"

**$4 > 2$**

Detta utläser vi som "4 är större än 2"

Det finns också två tecken som betyder "större än eller lika med" och "mindre än eller lika med"

**$X \leq 4$**

Detta utläser vi som att "x är mindre än eller lika med 4".

**$X \geq 2$**

Detta utläser vi som att "x är större än eller lika med 2".

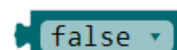
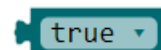
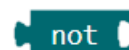
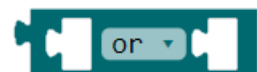
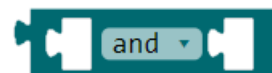
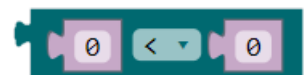
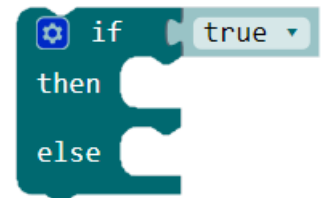
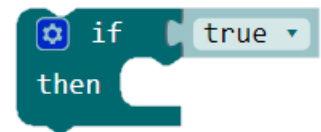
**$5 \neq 8$**

Detta utläser vi som att "5 är **inte** lika med 8".

JÄMFÖRELSE returnerar en BOOLEAN (sant eller falskt)

**and** värde ett OCH värde två måste vara sanna för att sant ska returneras

**or** värde ett ELLER värde två måste vara sanna för att sant ska returneras





## Block i Variables

**En variabel har ett namn och ETT värde/innehåll. Värdet förändras om det senare i programmet anges ett nytt värde till den variabeln**

I Variabler finns en variabel skapad. Den har namnet *item*.

När variabeln skapats kommer det ett nytt block med det namn som angetts.

*item* färdig variabel klar att användas

**set (item) to** värdet som ges till variabeln lagras i detta block tilldelar variabeln ett värde. Kan vara ett fast numeriskt värde eller värdet som tilldelats en annan variabel.

**change (variabel) by** \_ tar det aktuella värdet på variabeln och ändrar det med värdet infört i blocket till höger

Make a Variable

item ▾

set item ▾ to 0

change item ▾ by 1

## Block i Math

### Räkneoperationer

Blocken används för att genomföra räkneoperationer av två värden. Kan användas för värden eller variabler och returnerar beräknat värde. Fem olika räkneoperationer kan utföras.

Addition (+),

subtraktion(-),

multiplikation (x),

division (÷) och

^ vilket står för "upphöjt till"

**pick random 0 to \_** väljer ett slumpmässigt tal mellan 0 och valt tal

**remainder of 0 ÷ 1** dividerar ett tal med ett annat och returnerar den rest som blev kvar. EX:  $66 \div 10$  returnerar 6.

**min of \_ and \_** finner minsta värde av två insatta värden

**max of \_ and \_** finner största värdet av två insatta värden

**absolute of \_** finner absolutvärdet. Absolutvärdet är distansen från 0. Exempel: 2 och -2 har båda absolutvärdet 2

**text from char code** Alla bokstäver har ett numeriskt värde i programmering. Vanligast är att använda koden i ASCII-tabell <http://www.asciitable.com/> När knapp A trycks in kommer bokstaven D att visas.

