

## Különleges karakterek

;	Alt Gr + ,?;
=	Shift + 7
()	Shift + 8 és 9
{ }	Alt Gr + B és N
[ ]	Alt Gr + F és G
” (macskaköröm)	Shift + 2
' (apoztróf)	Shift + 1
\	Alt Gr + Q
& (ÉS)	Alt Gr + C
(VAGY)	Alt Gr + W
< > (kisebb-nagyobb)	Alt Gr + Í és Y

## Rajzoló függvények

size(szélesség, magasság);	ablak mérete
background(szürke);	háttér színezése, szürke: 0-255
background(piros, zöld, kék);	háttér, színek 0-255

stroke(szürke);	vonall színe, szürke: 0-255
stroke(piros, zöld, kék);	vonall színe, színek 0-255
stroke(piros, zöld, kék, áttetszőség);	vonall színe halványan, 0-255
strokeWeight(vastagság)	vonall vastagsága
noStroke();	ne legyen körvonal

fill(szürke);	kitöltés színe, szürke: 0-255
fill(piros, zöld, kék);	kitöltés, színek 0-255
fill(piros, zöld, kék, áttetszőség);	kitöltés halványan, 0-255
noFill();	üres legyen belül



# Alakzatok

<code>point(x, y);</code>	pont
<code>line(x1, y1, x2, y2);</code>	vonal
<code>ellipse(x, y, szélesség, magasság);</code>	ellipszis vagy kör
<code>ellipseMode(MÓD);</code>	hogyan rajzolja az ellipszist
RADIUS	x y középpont, többi a sugara
CENTER	x y középpont, többi átmérő
CORNER	x y bal felső sarok, többi átmérő
CORNERS	x1 y1 x2 y2, befoglaló sarkai
<code>triangle(x1, y1, x2, y2, x3, y3);</code>	háromszög
<code>quad(x1, y1, x2, y2, x3, y3, x4, y4);</code>	négyszög
<code>rect(x1, y1, x2, y2);</code>	téglalap vagy négyzet
<code>rect(x1, y1, x2, y2, kerekítés);</code>	lekerekített sarkú téglalap
<code>rect(x1, y1, x2, y2, BF, JF, JA, BA);</code>	lekerekített téglalap sarkonként
<code>rectMode(MÓD);</code>	hogyan rajzoljon téglalapot
CORNER	x1 y1 bal felső sarok, szél., mag.
CORNERS	x1 y1 egyik sarok, x2 y2 másik
CENTER	x1 y2 középpont, szél., mag.
RADIUS	x1 y2 közép, ettől mért szél. mag.
<code>beginShape();</code>	alakzat rajz kezdete
<code>texture(kép);</code>	alakzat mintája
<code>vertex(x1, y1, x2, y2);</code>	alakzat vonal, csak itt érvényes!
<code>endShape();</code>	alakzat rajzolásának vége
<code>endShape(CLOSE);</code>	alakzat vége, zárja be



## Elérhető adatok

width, height	ablak szélessége, magassága
frameRate	hány képkocka/másodperc

mouseX, mouseY	egér X, Y koordinátái
mouseButton	megnyomott egérgomb
LEFT, RIGHT	lehetséges egérgombok

key	utolsó használt billentyű
keyCoded	ha KEY == CODED akkor a különleges billentyű kódja
UP, DOWN, LEFT, RIGHT	iránybillentyűk
ALT, CONTROL, SHIFT	speciális billentyűk
keyPressed	meg van-e nyomva egy billentyű

## Események

void mouseClicked(){ }	egér kattintás történt
void mousePressed(){ }	egér gomb lenyomódott
void mouseReleased(){ }	egér gomb elengedődött
void mouseWheel(MouseEvent e){ println(e.getCount()); }	egér görgetése, e.getCount() mondja meg hogy mennyit görgetett és merre
void mouseMoved(){ }	egér megmozdult
void mouseDragged(){ }	egér gomb lenyomva és húzva

void keyPressed(){ }	billentyű megnyomódott
void keyReleased(){ }	billentyűt elengedtek
void keyTyped(){ }	billentyűt leütöttek kivéve Shift, Alt, Ctrl



# Nyelvtan

függvény amit mi csinálunk

```
void setup(){}
```

void	setup	()	{}
visszatérési érték, mit ad (alma, körte, 42)	függvény neve	paraméterek	függvény törzse, ide jönnek a parancsok

függvény amit használunk, paraméter nélkül

```
noStroke();
```

noStroke	()	;
függvény neve	zárójelek (shift + 8-9)	parancs vége (Alt Gr + ,?;)

függvény amit használunk, egy paraméterrel

```
stroke(255);
```

stroke	(	255	)	;
függvény neve	paraméter kezdődik	paraméter	paraméter vége	parancs vége

függvény amit használunk, több paraméterrel

```
size(800, 600);
```

size	(	800	,	600	)	;
függvény neve	paraméterek kezdődnek	első paraméter	paraméter elválasztása	második paraméter	paraméterek vége	lezáró pontosvessző



# Mintakód

```
// A fekete háttérből fehér lesz majd újra fekete
int szam = 0;
void setup(){
    size(800, 600);
}

void draw(){
    background(szam);
    szam++;
    if(szam >= 255){
        szam = 0;
    }
}
```

```
// A fekete háttérből fehér lesz majd újra fekete
int szam = 0;
void setup(){
    size(800, 600);
}

void draw(){
    background(szam);
    szam++;
    if(szam >= 255){
        szam = 0;
    }
}
```



# Osztályok

```
class Sajatosztaly{
    int egyikTagvaltozo;
    String nev;

    // Paraméter nélküli konstruktor
    Sajatosztaly(){
        egyikTagvaltozo = 0;
        masikTagvaltozo = "";
    }
    // konstruktor paraméterekkel
    Sajatosztaly(int szam, String nev){
        egyikTagvaltozo = szam;
        this.nev = nev;
    }

    void draw(){
        text(nev, egyikTagvaltozo, 0);
    }
}

Sajatosztaly gyariPeldany = new Sajatosztaly();
Sajatosztaly egyediPeldany =
    new Sajatosztaly(70, "osztály neve");
egyediPeldany.draw();
```

