Objects

Class 08

Agenda

- Feedback på sidste mini ex – Generativity
- Objektorienteret programmering (OOP)
- Pause
- Sample Code
- Mini ex 07
- Øvelse i par

Feedback i grupper

- Præsenter og beskriv dit program
- Hvilke regler har du brugt i dit program?

- Diskuter, hvordan programmet udtrykker generativitet (fx. authorship/nonhuman creation/ autonomous system/Complexity/Emergent behaviour etc)

Grundlæggerne af OOP

Ole-Johan Dahl (1931-2002) og Kristen Nygaard (1926-2002)

OOP

- OOP er det mest dominerende programmeringsparadigme indenfor den kommercielle og industrielle industri.
- Programmeringsparadigme = En fundamental måde at programmere en computer på.
    - NB: Et programmeringssprog kan godt understøtte flere paradigmer.

Objekter

- Lad os skrive et program for vores morgenrutine:
    1. Stå op
    2. Spis morgenmad
    3. Børst tænder
    4. Tag bussen til uni
- Objektet: Mig, en person
- Properties: Køn, hårfarve, øjenfarve, højde, vægt osv. = Variabler
- Abilities: Sove, vågne, spise, gå. = Funktioner

OOP inddeler koden i **klasser**

Defining object: Car

- Car data/attributter:
    - Car color
    - Car x location
    - Car y location
    - Car x speed
- Setup:
    - Initialize car object: color, size, location and speed
- Draw:
    - Fill background
    - Display car object
    - Drive car object

Non-OOP

```
color c = color(0);
float x = 0;
float y = 100;
float speed = 1;

void setup() {
  size(200,200);
}

void draw() {
  background(255);
  move();
  display();
}

void move() {
  x = x + speed;
  if (x > width) {
    x = 0;
  }
}

void display() {
  fill(c);
  rect(x,y,30,10);
}
```

Fra dette…

…til dette!

OOP

```
Car myCar;

void setup()  {
  myCar = new Car();
}

void draw()  {
  background(255);
  myCar.drive();
  myCar.display();
}
```

Hvad består classes af?

```
// Simple non OOP Car
color c;
float xpos;
float ypos;
float xspeed;

void setup() {
  size(200,200);
  c = color(255);
  xpos = width/2;
  ypos = height/2;
  xspeed = 1;
}

void draw() {
  background(0);
  display();
  drive();
}

void display () {
  rectMode(CENTER);
  fill(c);
  rect(xpos,ypos,20,10);
}

void drive() {
  xpos = xpos + xspeed;
  if (xpos > width) {
    xpos = 0;
  }
}
```

```
class Car {
  color c;
  float xpos;
  float ypos;
  float xspeed;

  Car() {
    c = color(255);
    xpos = width/2;
    ypos = height/2;
    xspeed = 1;
  }

  void display() {
    rectMode(CENTER);
    fill(c);
    rect(xpos,ypos,20,10);
  }

  void drive() {
    xpos = xpos + xspeed;
    if (xpos > width) {
      xpos = 0;
    }
  }
}
```

→ The class name

→ Data

→ Constructor

→ Functionality

Pause

Hvordan tegner Winnie biler?

```
3   // step 1. Declare Objects
4   var car = [];
```

```
12  function setup() {
13   createCanvas(windowWidth, windowHeight);
14
15  //step 2. Initialize object
16   car[0] = new Car(color(255,0,0), 10, 10, 100, 20);
17   car[1] = new Car(color(0,0,255), 15, 20, 300, 10);
18
19  }
```

```
21  function draw() {
22   background(bg);
23
24
25  //step 3. Use object
26   for (var i = 0; i <car.length; i++) {
27
28    car[i].drive();
29    car[i].display();
30   }
31  }
```

Class name

- function Navn { klassens kode }
- Skal stå udenfor setup() & draw() – hvorhenne er ligemeget
- Klassenavne har stort forbogstav, så vi kan kende forskel på dem og variabler

```
43  function Car(getcolor, speed, xpos, ypos, size) {
44
45    this.getcolor = getcolor;
46    this.speed = speed;
47    this.pos = new createVector(xpos, ypos);
48    this.size = size;
49
```

Class data

- Class data (eller attributter, properties) er en samling af variabler.

```
43  function Car(getcolor, speed, xpos, ypos, size) {
44
45    this.getcolor = getcolor;
46    this.speed = speed;
47    this.pos = new createVector(xpos, ypos);
48    this.size = size;
49
```

Constructor

- Denne funktion skaber objektet: Vi instruerer computeren i, hvordan objektet skal sættes op.
- Kaldes ved at bruge “**new**” operator’en, som laver et nyt objekt.

Constructor

```
12  function setup() {
13    createCanvas(windowWidth, windowHeight);
14
15  //step 2. Initialize object
16    car[0] = new Car(color(255,0,0), 10, 10, 100, 20);
17    car[1] = new Car(color(0,0,255), 15, 20, 300, 10);
18
19  }
```

```
43  function Car(getcolor, speed, xpos, ypos, size) {
44
45    this.getcolor = getcolor;
46    this.speed = speed;
47    this.pos = new createVector(xpos, ypos);
48    this.size = size;
49
```

Functionality

- Vi kan tilføje funktionalitet til objektet:

```
50   this.drive = function() {
51
52     this.pos.x = this.pos.x + this.speed;
53
54     if (this.pos.x > width) {
55         this.pos.x = 0;
56     }
57
58   }
59
60   this.display = function() {
61     noStroke();
62     fill(this.getcolor);
63       rect(this.pos.x,this.pos.y,this.size,this.size);
64   }
65
66 }
67
```

p5.dom object

```
3   // step 1. Declare Objects
4   var car = [];
5   var button;
```

```
12  function setup() {
13    createCanvas(windowWidth, windowHeight);
14    button = createButton('add');
15    button.mousePressed(add);
```

```
22  function draw() {
23    background(bg);
24    button.position(0,0);
```

```
35  function add() {
36
37    car[car.length] = new Car(color(random(155,255)), random(2,10), random(10,20), random(10,500), random(30));
38
39    //append can be used as well, which has the same effect as line 37
40    //append(car,  new Car(color(random(155,255)), random(2,10), random(10,20), random(10,500), random(30)));
41  }
```

Opsamling på sample code

Netscape - [FORM]

File Edit View Go Bookmarks Options Directory Window Help

Home Edit Reload Open Print Find

Location: http://archive.rhizome.org/anthology/form-art/www.c3.hu/collection/form/

FORM ART COMPETITION

who: Alexei Shulgin

what: Form

where: c3

browser: Netscape 3.0

Contact

THANKS TO

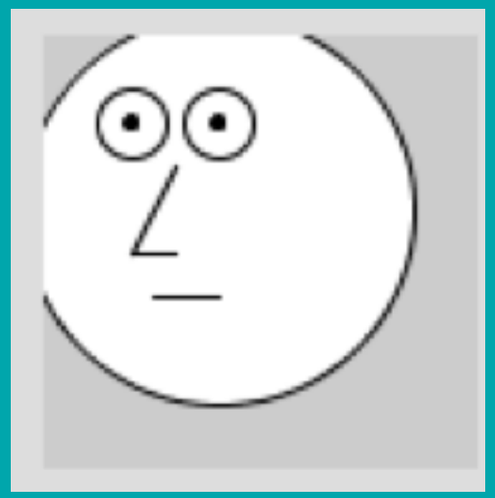
Document: Done

Mini ex 7: Form Art

Imagine you are going to apply to the Form Art Competition, and you are required to create an interactive form art (with p5.dom library and use of objects).

This mini exercise addresses objects in two ways: 1/ the general notion of object in object-oriented programming language 2/ HTML5 objects in p5.dom. What are the advantages in using objects and what might be the limitations? What's your experience in using objects?

Øvelse



- Gå sammen to og to
- Find ét af jeres første programmer frem og omskriv koden med objekter
- Hjælp: https://p5js.org/examples/objects-objects.html