

# Sysbas1A

Debuggen

# Vandaag

- Bugs en errors
- Hoe vind je bugs
- Vermijden
- Oefenen

# Bugs en errors

- Typos
- Verkeerde data type of undefined
- Out of index
- Scope
- Volgorde
- Missende functies

# Bugs en errors -> Typos

- Typefouten
  - `elipse()` of `ellipse()` Dubbele l
- Missende hoofdletters
  - `xSpeed` of `xspeed` Hoofdletter

# Bugs en errors -> Verkeerde data type

- `"8" + 1 = ?` Een int (getal) optellen bij een string (woord) kan, maar geeft een ander antwoord dan je zou verwachten (namelijk 81)
- `undefined`

# Bugs en errors -> Out of index

```
let array = [0, 1, 3, 4, 1, 6, 7, 0, 2, 5]  
array[10] = ? Array index loopt vanaf 0
```

# Bugs en errors -> Scope

```
function setup() {  
  let variabele = 10;  
}  
function draw() { Kan niet bij de variabele omdat hij in een andere functie gedefinieerd is  
  console.log(variabele);  
}
```

# Bugs en errors -> Scope

```
function setup() {  
  let variabele = 10;  
}  
function draw() {  
  let variabele = 20;  
  console.log(variabele);  
}
```

Hier komt 20 uit, hij weet van het bestaan van de '10' niets af



# Bugs en errors -> Volgorde

```
console.log(variabele) // ERROR ReferenceError, variabele bestaat niet
let variabele;
function setup() {
  console.log(variabele) // undefined Variabele is nog leeg, wordt pas later gevuld
}
function draw() {
  variabele = 10;
  console.log(variabele) // 10
}
```

# Bugs en errors -> Missende functies

```
<script src="sketch.js"></script> <!-- Connect is not defined -->  
<script src="../oscLib.js"></script>
```

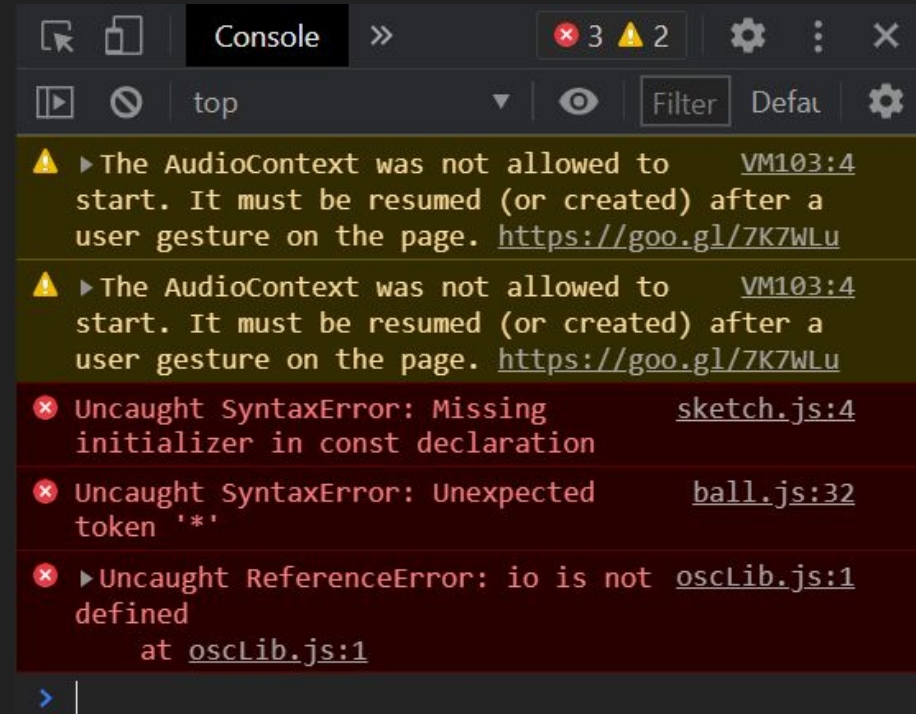
sketch.js heeft functies nodig die zijn gedefinieerd in oscLib.js  
hierom moet oscLib.js vóór sketch.js staan

# Vinden van bugs

- Error tracer
- Blindstaren
- Divide and conquer
- Print debugging
- Peer review

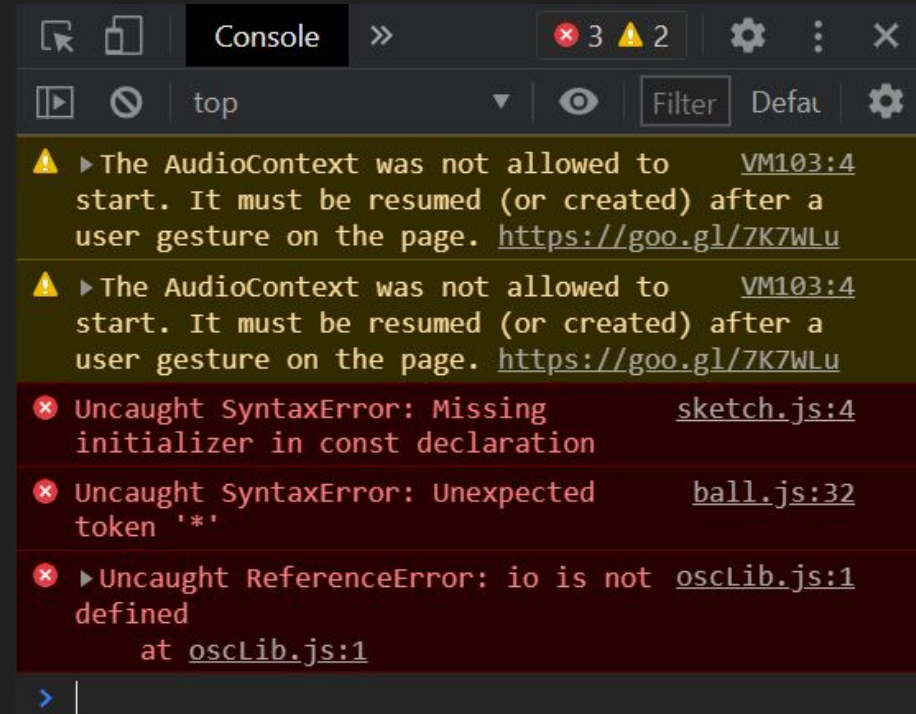
# Vinden van bugs -> Error tracer

- Terminal / F12 / Inspector
- Regelnummer
- Foutcode
- Beschrijving van error



# Vinden van bugs -> De error message

- Niet de hele error message
- Geen pad naar een file



The screenshot shows a browser's developer console with the following error messages:

```
▶ The AudioContext was not allowed to start. It must be resumed (or created) after a user gesture on the page. https://goo.gl/7K7WLu VM103:4
```

```
▶ The AudioContext was not allowed to start. It must be resumed (or created) after a user gesture on the page. https://goo.gl/7K7WLu VM103:4
```

```
✖ Uncaught SyntaxError: Missing initializer in const declaration sketch.js:4
```

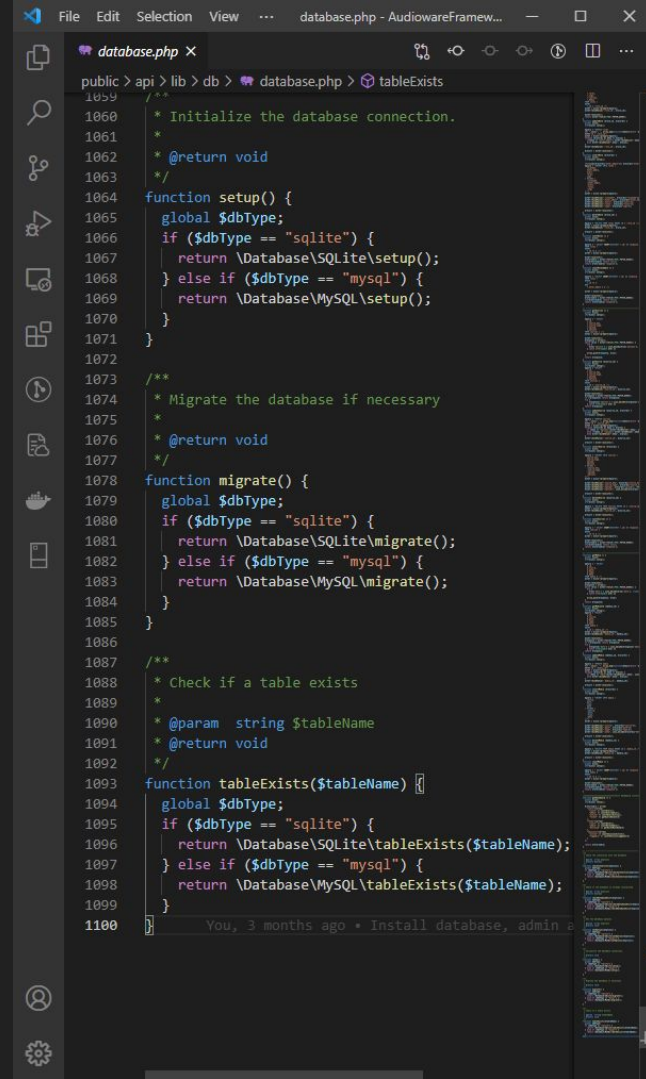
```
✖ Uncaught SyntaxError: Unexpected token '*' ball.js:32
```

```
✖ ▶ Uncaught ReferenceError: io is not defined oscLib.js:1  
    at oscLib.js:1
```

The console interface includes a 'Console' tab, a filter dropdown set to 'top', and a status bar showing 3 errors and 2 warnings.

# Vinden van bugs -> Blindstaren

- Alles lezen van begin tot eind tot je eindelijk een keer de fout vind...
- Grote files niet te doen
- Kleine stukken code bruikbaar



The image shows a code editor window with a dark theme. The title bar at the top reads "database.php - AudiodwareFrame...". The editor displays PHP code for a database connection and migration utility. The code is organized into several functions: `setup()`, `migrate()`, and `tableExists()`. The `tableExists()` function is currently selected, indicated by a blue highlight and a mouse cursor. The code includes comments and docblocks, such as `/** Initialize the database connection. */` and `/** Migrate the database if necessary */`. The file path in the breadcrumb is `public > api > lib > db > database.php > tableExists`. The status bar at the bottom shows "You, 3 months ago • Install database, admin a".

```
public > api > lib > db > database.php > tableExists
1059 /...
1060  * Initialize the database connection.
1061  *
1062  * @return void
1063  */
1064  function setup() {
1065      global $dbType;
1066      if ($dbType == "sqlite") {
1067          return \Database\SQLite\setup();
1068      } else if ($dbType == "mysql") {
1069          return \Database\MySQL\setup();
1070      }
1071  }
1072
1073  /**
1074  * Migrate the database if necessary
1075  *
1076  * @return void
1077  */
1078  function migrate() {
1079      global $dbType;
1080      if ($dbType == "sqlite") {
1081          return \Database\SQLite\migrate();
1082      } else if ($dbType == "mysql") {
1083          return \Database\MySQL\migrate();
1084      }
1085  }
1086
1087  /**
1088  * Check if a table exists
1089  *
1090  * @param string $tableName
1091  * @return void
1092  */
1093  function tableExists($tableName) {
1094      global $dbType;
1095      if ($dbType == "sqlite") {
1096          return \Database\SQLite\tableExists($tableName);
1097      } else if ($dbType == "mysql") {
1098          return \Database\MySQL\tableExists($tableName);
1099      }
1100  }
```

# Vinden van bugs -> Divide and conquer

- Kleine stukjes uitvoeren en testen of het probleem zich daar voordoet

```
draw() {  
  for (let step = 0; step < this.stepsAmount;  
    // const state = this.states[step];  
    // if (running && step == this.currentStep) {  
    //   if (state) fill('#f00')  
    //   else fill('#faa')  
    // }  
    // else {  
    //   if (state) fill('#aaa')  
    //   else fill('#fff')  
    // }  
    | const posX = this.posX + (this.rectSize +  
    // rect(posX, this.posY, this.rectSize, t  
    // this.play(step)  
    }  
  }  
}
```

You, seconds ago • Uncommitted changes

# Vinden van bugs -> Print debugging

- Overal een print tussen zetten zodat je weet wat er gebeurt
- `console.log(variabele);`

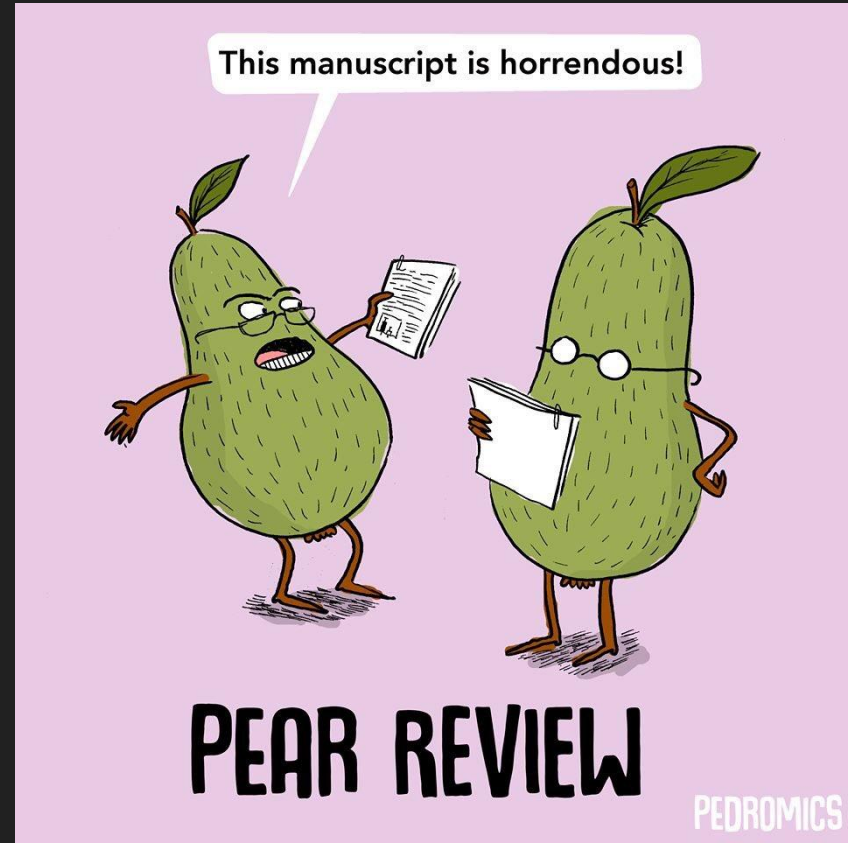
```
draw() {  
  for (let step = 0; step < this.stepsAmount; step++) {  
    const state = this.states[step];  
    console.log("state", state)  
    console.log("running", running)  
    console.log("step", step)  
    console.log("this.currentStep", this.currentStep)  
    if (running && step == this.currentStep) {  
      if (state) fill('#f00')  
      else fill('#faa')  
    }  
    else {  
      if (state) fill('#aaa')  
      else fill('#fff')  
    }  
    const posX = this.posX + (this.rectSize + this.padding) * step  
    console.log("rect", posX, this.posY, this.rectSize, this.rectSize)  
    rect(posX, this.posY, this.rectSize, this.rectSize)  
    this.play(step)  
  }  
}
```

You, seconds ago • Uncommitted changes



# Vinden van bugs -> Peer review

- Overheen lezen



# Voorkomen

- Syntax beter kennen
- Variabel namen
- Opsplitsen
- Overzicht door middel van indents
- Comments

# Voorkomen -> Syntax beter kennen

`count(array)`

`sizeof(array)`

`len(array)`

`array.length`

`array.length()`

# Voorkomen -> Variabel namen

- Naam net zo veel betekenis als de waarde van de variabele
- camelCase
  - Alle woorden een hoofdletter behalve de eerste
- PascalCase
  - Alle woorden hebben een hoofdletter
- snake\_case
  - spaties vervangen door een underscore ( \_ )
- Gebruik één hoofdtaal (Engels)

```
let r = 10
```

```
let rIsDeRadiusVanDeEersteBal = 10
```

# Voorkomen -> Opsplitsen

In plaats van:

```
if ( (mouseX > this.posX && mouseX < this.posX + this.rectSize) && (mouseY > this.posY && mouseY < this.posY + this.rectSize)) {  
  console.log("De muis is ingedrukt op " + mouseX + ", " + mouseY);  
}
```

Doe je dit:

```
let insideXBounds = (mouseX > this.posX) && (mouseX < this.posX + this.rectSize);  
let insideYBounds = (mouseY > this.posY) && (mouseY < this.posY + this.rectSize);  
if (insideXBounds && insideYBounds) {  
  console.log("De muis is ingedrukt op " + mouseX + ", " + mouseY);  
}
```

# Voorkomen -> Overzicht

```
draw() {  
  for (let step = 0; step < this.stepsAmount; step++) {  
    const state = this.states[step];  
    if (running && step == this.currentStep) {  
      if (state) {  
        fill('#f00');  
      }  
      else {  
        fill('#faa');  
      }  
    }  
    else {  
      if (state) {  
        fill('#aaa');  
      }  
      else {  
        fill('#fff');  
      }  
    }  
    const posX = this.posX + (this.rectSize + this.padding) * step;  
    rect(posX, this.posY, this.rectSize, this.rectSize);  
    this.play(step);  
  }  
}
```

```
draw() {  
  for (let step = 0; step < this.stepsAmount; step++) {  
    const state = this.states[step];  
    if (running && step == this.currentStep) {  
      if (state) {  
        fill('#f00');  
      }  
      else {  
        fill('#faa');  
      }  
    }  
    else {  
      if (state) {  
        fill('#aaa');  
      }  
      else {  
        fill('#fff');  
      }  
    }  
    const posX = this.posX + (this.rectSize + this.padding) * step;  
    rect(posX, this.posY, this.rectSize, this.rectSize);  
    this.play(step);  
  }  
}
```

# Voorkomen -> Comments

- Beschrijven een regel of een stuk code

Oefenen



# Oefenen

- Download de opdracht op <https://home.hku.nl/~jochem.vaniterson/sysbas/>
- Los in duos deze bugs op en noteer:
  - Wat was de bug
  - Waar vond je de bug
  - Hoe heb je de bug gevonden
  - Hoe heb je de bug opgelost
- Aan het einde komen de antwoorden met uitleg online

# Sysbas1A

Debuggen