

JAVASCRIPT PATTERN

Part I

LINEARE SPAGHETTI

```
var i = 0;
if (i >= 5) {
    document.getElementById("box").innerHTML = "none";
}

for(var a = 15; a < 30; a--) {
    alert("zahl: "+a);
}

console.log("Hier steht Text");
```

FUNCTIONS

```
function hideBox(i) {  
    if (i >= 5) {  
        document.getElementById("box").innerHTML = "none";  
    }  
}
```

```
function myLoop() {  
    for(var a = 15; a < 30; a--) {  
        alert("zahl: "+a);  
    }  
}
```

```
var i = 0;  
hideBox(i);  
myLoop();  
console.log("Hier steht Text");
```

FUNCTIONS



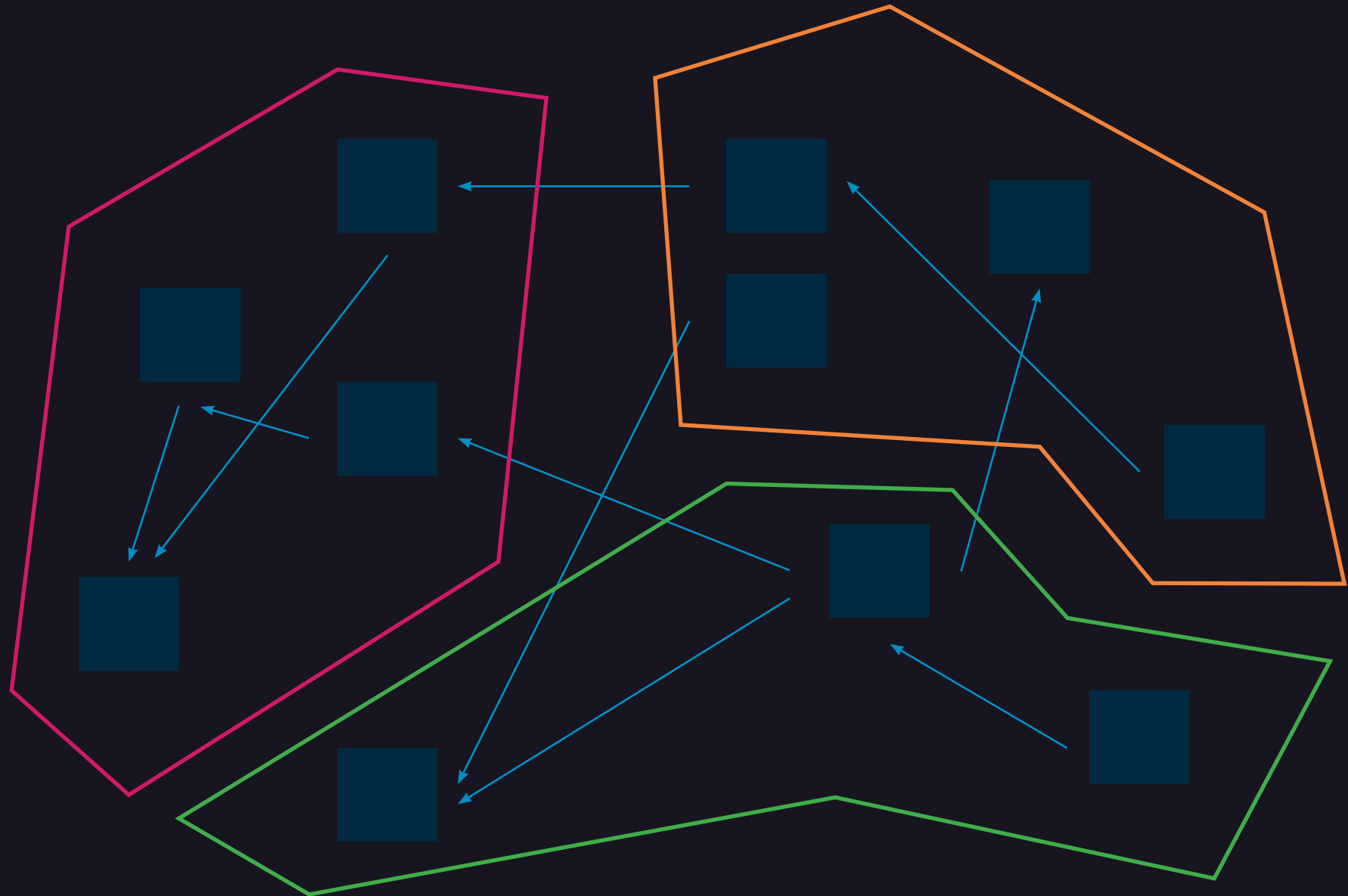
FUNCTIONS



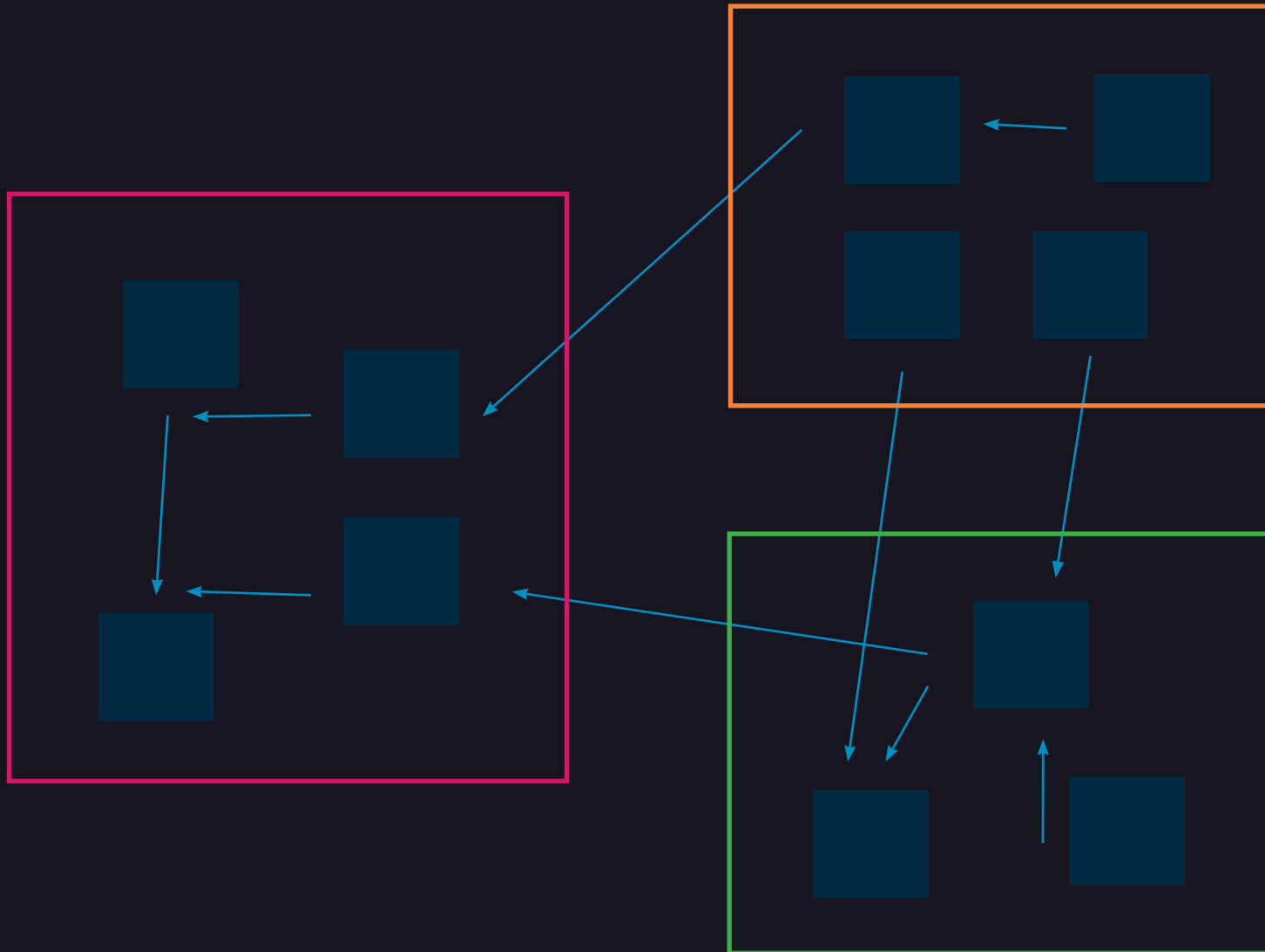
STRUKTURE



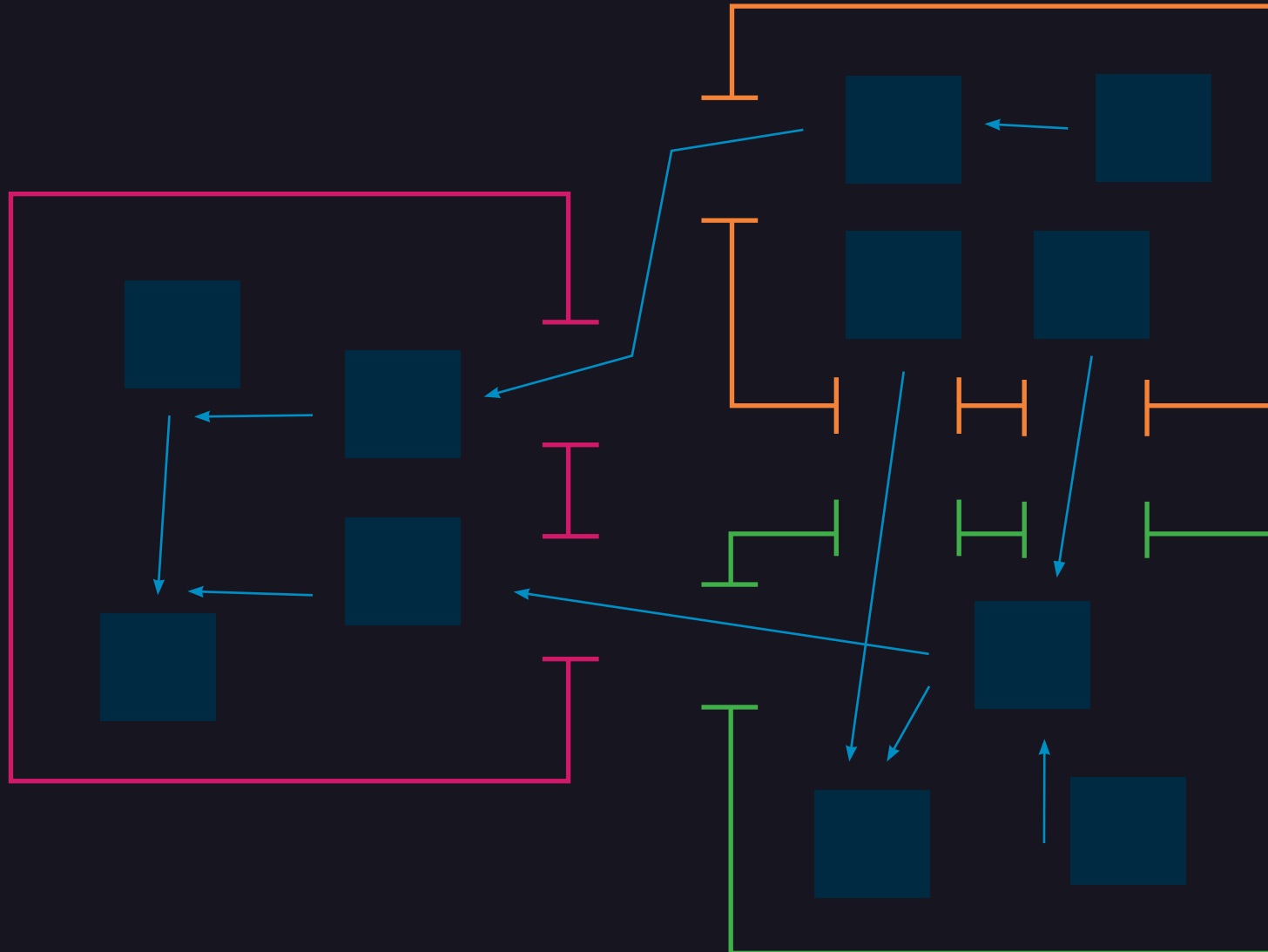
STRUKTURE



OBJECTE



SCHNITTSTELLEN



JS OBJECT ERZEUGEN

Keyword

|

Keyword

|

```
var Modul = new Object();
```

|

Name

|

Child

OBJECT EIGENSCHAFTEN

```
var Modul = new Object();  
  
Modul.eigenschaft = "wert";
```

OBJECT METHODEN

```
var Modul = new Object();  
Modul.eigenschaft = "wert";  
  
Modul.methode = function () {  
    alert(Modul.eigenschaft);  
};
```

OBJECT METHODEN

```
var Modul = new Object();  
Modul.eigenschaft = "wert";  
Modul.methode = function () {  
    alert(Modul.eigenschaft);  
};  
  
Modul.methode();
```

... ODER SO

```
var Modul = {  
    eigenschaft : "wert",  
    methode : function () {  
        alert(Modul.eigenschaft);  
    }  
};  
Modul.methode();
```

OBJECT = OBJECT = OBJECT

```
var Modul = new Object();  
Modul.eigenschaft = "wert";  
Modul.methode = function () {  
    alert(Modul.eigenschaft);  
};  
  
Modul.methode();
```

```
var Modul = {  
    eigenschaft : "wert",  
    methode : function () {  
        alert(Modul.eigenschaft);  
    }  
};  
Modul.methode();
```

VERSCHACHTELUNG

window

(globales Objekt)



Modul

(Objekt)



eigenschaft

(String)



methode

(Function)

ALLES PUBLIC ...!

window

(globales Objekt)



Modul

(Objekt)



eigenschaft

(String)

methode

(Function)



PRIVATER FUNKTIONS-SCOPE

Functions Declaration

|

```
(function () {  
    /* ... */  
})();
```

|

direkter Functions Aufruf

PRIVATE OBJEKTE

```
var Modul = (function () {  
  
    /* ... private Objekte ... */  
    var privatVariable = "my Name";  
  
    /* Gebe öffentliche API zurück: */  
    return {  
        publicMethod : function () {  
            return privatVariable;  
        }  
    };  
  
})();
```

SYNTAX FUNCTION

```
var myFunc = function () { /*...*/ };
```

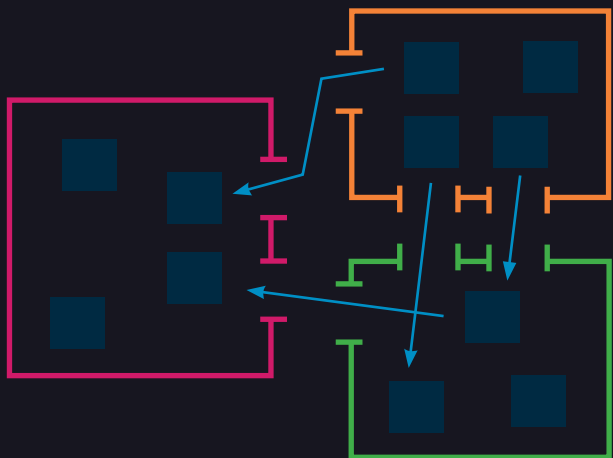
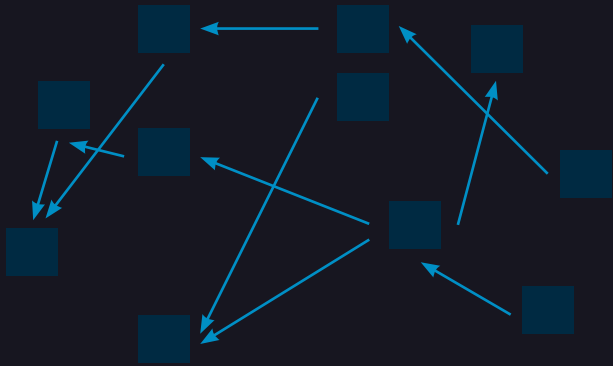
JS URSCHLEIM

```
var myFunc = function () { /*...*/ };  
  
myFunc.prototype.methode = function () {  
    /* ... */  
};
```

INSTANZEN

```
function Katze () {}  
  
Katze.prototype.pfoten = 4;  
Katze.prototype.miau = function () {  
    alert("Miau!");  
};  
  
var maunzi = new Katze();  
var schnucki = new Katze();  
  
maunzi.miau();
```

ZUSAMMENFASSUNG



```
var Modul = {  
  eigenschaft : "wert",  
  methode : function () {  
    alert(Modul.eigenschaft);  
  }  
};  
Modul.methode();
```

```
var myFunc = function () { /*...*/ };  
myFunc.prototype.eigenschaft = "wert";  
myFunc.prototype.methode = function () {  
  /* ... */  
};
```