

Solution Objects and Prototypes

- Exercises are given every week on the PL page of the SCG website (<http://scg.unibe.ch/teaching/pl>)
- Solutions to each assignment must be sent to **mohammadreza.hazhirpasand@inf.unibe.ch**
- The solutions of the assignments are to be delivered before every Thursday at 11 PM. Solutions handed in later than the specified time will not be accepted. In case of serious reasons send an e-mail to **mohammadreza.hazhirpasand@inf.unibe.ch**

Exercise (6 points)

1. You should implement the `Person` prototype which has the following characteristics: (3 pts)
 - (a) The `name` property which is accessible by other objects of this prototype.
 - (b) The `password` property which cannot be accessed by other objects of this prototype.
 - (c) The `counter` property whose value is shared among all objects of this prototype.

You need to create two objects of this prototype in order to do the following tasks:

- (a) Show how to access the `name` property in both objects.
- (b) What output do you get if you try to access the `password` property? How can you correct it?
- (c) How to access the `counter` property? If you change the value of the `counter` property in one object, does it affect the property value on the second object?

Note: please only submit your JavaScript code

Answer:

```
var person = function (nm) {
  this.name = nm
  var secret = "Secret";
    person.shared_secret = 25;

  this.getSecret = function () {
    return secret;
  }
  this.setSecret = function () {
    secret = "not secret";
  }
  this.getName = function () {
```

```
        return this.name;
    }
    this.getAge = function() {
    return person.shared_secret;
    };
    this.growOlder = function () {
    person.shared_secret++;
    }

    return this;
};

var person1 = new person("mh1");
var person2 = new person("mh2");
console.log(person1.name);
console.log(person2.name);
console.log(person2.secret);
console.log(person2.secret);
console.log(person1.getSecret());
person1.setSecret();
console.log(person1.getSecret());
console.log(person2.getSecret());

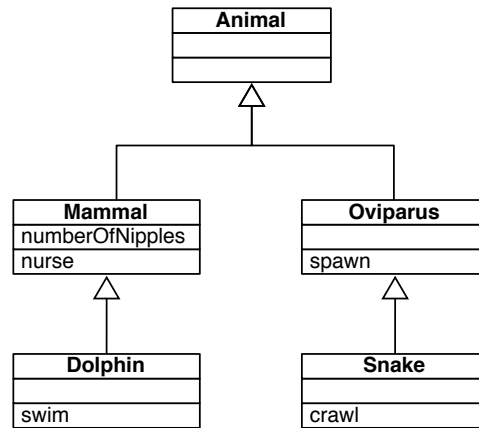
console.log(person1.getAge()); // 25
console.log(person2.getAge()); // 25
person1.growOlder();
console.log(person1.shared_secret)
console.log(person1.getAge()); // 26
console.log(person2.getAge()); // 26

console.log(person1.shared_secret);
```

2. A possible classification for animals is shown in Figure 1. When it comes to classify the platypus you realize that it nurses but it also spawns. Therefore, implement the class diagram shown in figure 1 including the poor platypus in Java Script. Use the [Platypus.html](#) file as skeleton for your implementation. (3 pts)

Answer:

```
var animal = {}
```



platypus



Figure 1: Animal classification

```
var mammal = Object.create(Animal);
mammal.numberOfNipples = 2;
mammal.nurse = function () {
  alert("I am nursing");
}
mammal.getNumberOfNipples = function(){
  alert(this.numberOfNipples);
}

dolphin = Object.create(mammal);
dolphin.numberOfNipples = 4;
dolphin.swim = function() {
  alert("I am swimming");
}

var oviparus = Object.create(Animal);
```

```
oviparus.spawn = function() {  
  alert("I am spawning");  
}  
  
var snake = Object.create(oviparus);  
snake.crawl = function() {  
  alert("I am crawling");  
}  
  
var platypus = Object.create(mammal);  
platypus.numberofNipples = 6;  
platypus.spawn = oviparus.spawn;  
  
function display(text) {  
  document.getElementById("output").innerHTML += text + "\n";  
}
```