

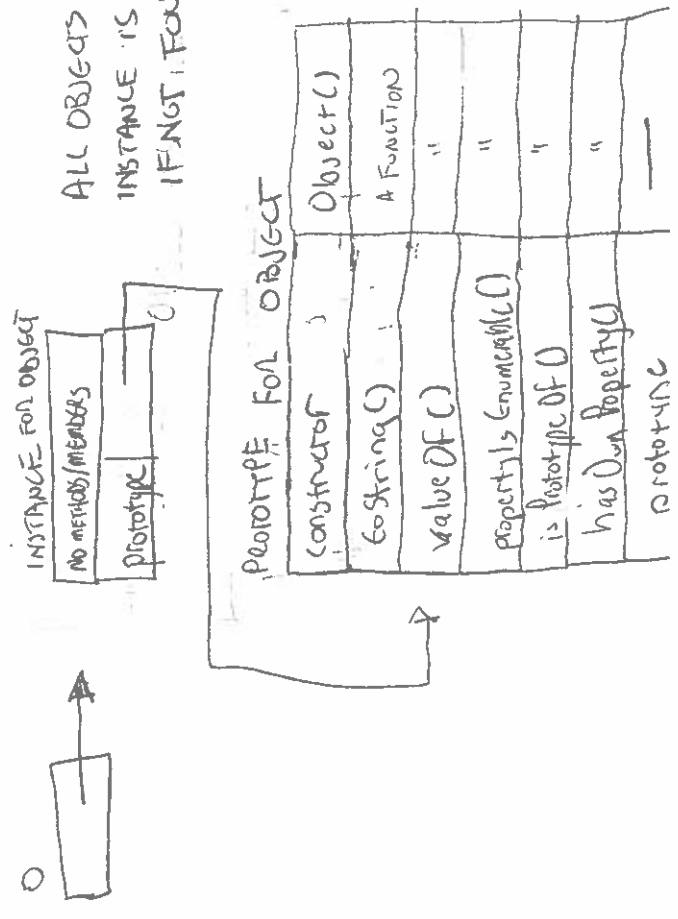
COMPLEX TYPES

JAVASCRIPT USES AN OOP TECHNIQUE CALLED PROTOTYPING

- CREATE PROTOTYPE OBJECT \approx CLASS
- CREATE INSTANCES OF PROTOTYPE \approx OBJECT
- INHERIT FROM PARENT VIA PROTOTYPE MEMBER

```
var o = new Object();
document.write(o.toString());
```

ALL OBJECTS HAVE A "PROTOTYPE" MEMBER
 INSTANCE IS CHECKED FOR MEMBER/METHOD ACCESSED
 IF NOT FOUND, THEN PROTOTYPE CHAIN IS CHECKED



CREATING CLASS / OBJECT :-

ALL FUNCTIONS IN JAVASCRIPT ARE CLASS CONSTRUCTORS BUT YOU CAN IGNORE THIS

function Person() {
 ← (CREATES A PERSON FUNCTION WHICH BECOMES A CLASS CONSTRUCTOR WHEN WE USE IT WITH NEW)

var p1 = new Person(); ← PERSON OBJECT

ADDING MEMBERS

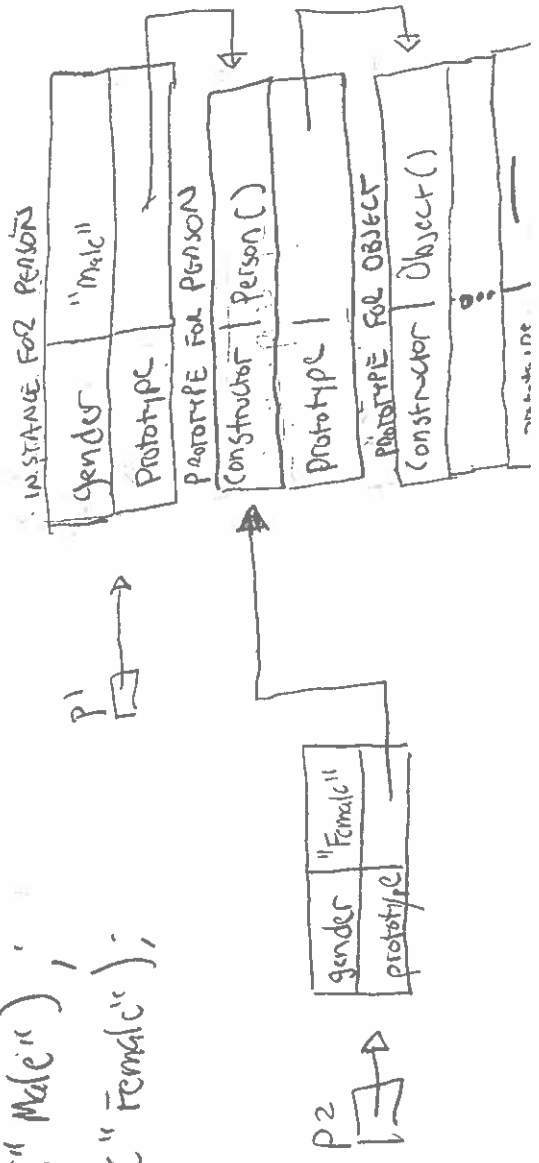
REDEFINED CONSTRUCTOR; THERE CAN BE ONLY ONE CONSTRUCTOR!

function Person(gender)

{
 this.gender = gender; ← "THIS" TARGETS MEMBERS/METHODS OF THE PERSON CLASS
 }

var p1 = new Person("Male");

var p2 = new Person("Female");



ADDING METHODS

```
function Person (gender)
```

```
{
```

```
  this.gender = gender;
```

```
  this.sayHi = function () { document.write("Hi"); };
```

INLINE

```
  this.sayBye = sayBye;
```

```
}
```

TWO WAYS OF DEFINING METHOD

```
function sayBye()
```

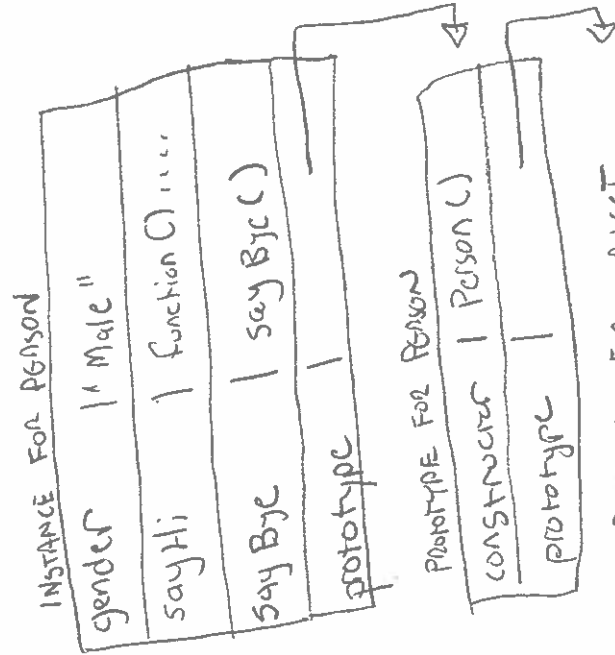
```
{
```

```
  document.write("Bye");
```

```
}
```

```
var p1 = new Person("Male");
```

```
p1.sayHi();
```



PROTOTYPE FOR PERSON

INHERITING FROM PARENT CLASS

```
function Student (gender) {
  this.gender = gender;
  this.sayHi = function () { document.write ("Yo!"); };
}
```

```
Student.prototype = new Person();
```

↳ WIRE PERSON INTO STUDENT'S PROTOTYPE CHAIN

```
Student.prototype.constructor = Student;
// MAKE SURE STUDENT DOESN'T INHERIT PERSON'S CONSTRUCTOR
```

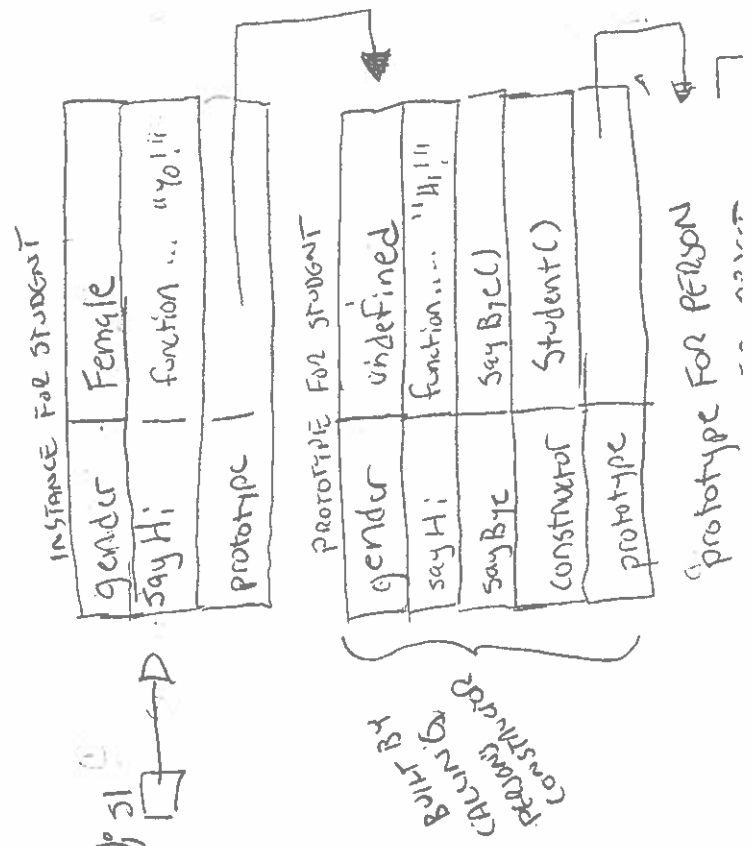
```
var s1 = new Student ("Female");
```

```
s1.sayHi();
```

```
s1.sayBye();
```

Output

Yo!
Bye!



ASK CLASS

```

Student.prototype.sayHi = sayBye
Student.class = 2010;
var s1 = new Student("Male");

```

```

s1.gpa = 4.0; // ADDS TO INSTANCE

```

```

s1.sayHi = function () { document.write("Howdy!"); };

```

```

Student vs s1: sayHi(); Bye! Howdy!
Student vs. s1: sayBye(); Bye! Bye
Student vs. s1: gpa; undefined | 4.0
Student vs. s1: class; 2010

```

