This example shows how to add method to class

You have two choice

1. Create instance method
2. Create prototype method

Difference

**Instance method** is slow in performance as it will **allocate memory for every object**

**Prototype method** is provide better performance as it will stored in **separate memory block (virtual table)** and every object will have same copy of function

Instance method can make use of private variable eg if you have declare var x in function person

Instance method can have access to this variable

Prototype method can not access variable declared in function with var key word as scope of var will be within the function

<script>

"use strict"

function person(nm,ag)

{

if(this instanceof person)

{

this.name=nm;

this.age=ag;

this.speak=function(){return " hello "}

alert(this.constructor)

return this;

}

else

throw new TypeError("Check this is not a function");

}

person.prototype.walk=function(){

return " can walk ";

}

var obj1=new person("raj",80);

console.log(obj1);

document.write(obj1);

var x=new person("mona",90);

console.log(x);

document.write(x);

document.write(x.speak());

document.write(x.walk());

</script>

Try this------------------

<script>

"use strict"

function person(nm,ag)

{

**var x="private data";**

if(this instanceof person)

{

this.name=nm;

this.age=ag;

this.speak=function(){return " hello "+ x}

alert(this.constructor)

return this;

}

else

throw new TypeError("Check this is not a function");

}

person.prototype.walk=function(){

return " can walk ";

}

var obj1=new person("raj",80);

console.log(obj1);

document.write(obj1);

var x=new person("mona",90);

console.log(x);

document.write(x);

document.write(x.speak());

document.write(x.walk());

</script>