

Programación orientada a objetos

con JavaScript

Francisco Arce
www.pacoarce.com

¿Qué es el prototype?

en JavaScript



Francisco Arce
www.pacoarce.com

¿Qué es el prototype?

Las funciones son objetos que tienen propiedades (arguments, prototype y constructor) y métodos (call() y apply())

¿Qué es el prototype?

prototype es una propiedad que tiene toda función y es de tipo objeto. En un inicio es un objeto vacío.

¿Qué es el prototype?

Por medio de la propiedad prototype podemos añadir métodos y propiedades a la función de clase o constructor.

¿Qué es el prototype?

El prototipo solo se utiliza cuando utilizas a la función como función constructora.

Añadir propiedades al prototipo

JavaScript



Francisco Arce
www.pacoarce.com

Añadir propiedades al prototype

Cuando creas una función como constructora por medio de “new”, puedes hacer referencia al mismo por medio de “this” (la cual no es una variable).

Añadir propiedades al prototype

A las propiedades añadidas con “this” se les conoce como “own properties”.

A las propiedades añadidas con “prototype” se les conoce como “prototype properties”.

Añadir propiedades al prototype

TANQUE

Own properties
this.balas;
this.vidas;
this.energia;

Prototype properties
prototype.x;
prototype.y;
prototype.angulo;

Usar propiedades y métodos del prototipo

JavaScript



Francisco Arce
www.pacoarce.com

Usar propiedades y métodos del prototipo

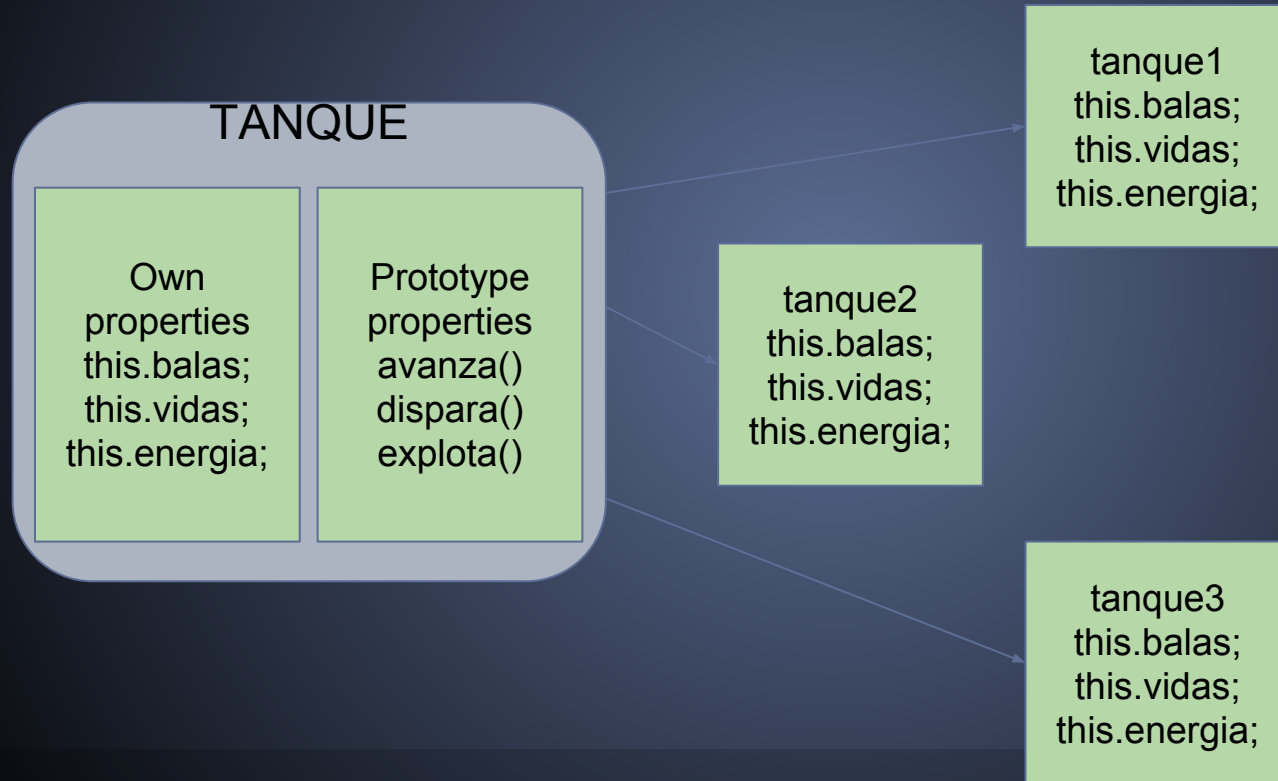
Como el prototipo es un objeto, cualquier modificación la pueden visualizar todas las instancias, incluso si fueron creadas antes de la modificación al prototipo.

Usar propiedades y métodos del prototipo

Todo lo que se encuentre en el prototipo se pasa “por referencia”, por lo tanto, no se duplica en cada instancia.

En cambio en las “own properties” si se duplican en cada instancia.

Usar propiedades y métodos del prototipo



Propiedades del objeto y propiedades del prototipo

JavaScript



Francisco Arce
www.pacoarce.com

Prioridades del prototipo

JS primero busca en la función la propiedad, y si no existe la busca en el objeto prototype. Si hay una propiedad de la función y una propiedad en el objeto prototype, la propiedad de la función tiene precedencia.

Prioridades del prototipo

Si eliminas la propiedad de la función (own properties), aparecerá la propiedad prototype. A esto se le conoce como

“cadena de prototipos”

o

“property chain”

Sobreescribir propiedades de la función

JavaScript



Francisco Arce
www.pacoarce.com

Sobreescribir propiedades

Para que sepamos si la propiedad es de la función o del prototipo, tenemos el método `hasOwnProperty()`.

Sobreescribir propiedades

Si borramos una propiedad de la función, JavaScript nos regresará la propiedad del prototipo.

Visualizar el contenido del prototipo

JavaScript



Francisco Arce
www.pacoarce.com

Visualizar el contenido del prototipo

Por medio de un ciclo `for...in` podemos enlistar todas las propiedades de un objeto.

Por medio de la función `hasOwnProperty()` sabemos si son propiedades del objeto (`true`) o del objeto `prototype` (`false`).

Visualizar el contenido del prototipo

Para saber si una variable es enumerable (visible en el `for..in`) utilizamos el método `propertyIsEnumerable()`;

isPrototypeOf()

JavaScript



Francisco Arce
www.pacoarce.com

El método `isPrototypeOf()`

El método *`isPrototypeOf()`* nos indica si un objeto es prototipo de otro.

Regresa un valor booleano si son prototipos o un falso si no.

__proto__



Francisco Arce
www.pacoarce.com

Prototype

Todos los objetos en JavaScript tienen una propiedad prototype que es otro objeto.

En JavaScript, los objetos heredan por medio de la propiedad prototype.

El prototipo del objeto Object es el eslabón más alto en la cadena de prototipos.

Prototype

Cualquier objeto hereda de `Object.prototype`.

La relación entre el objeto y la clase que le hereda es “`__proto__`”

No es buena idea usarla porque no es compatible (IE)

Prototype

prototype pertenece a la función constructora

__proto__ pertenece a las instancias

Un sustituto de __proto__ es Object.

getPrototypeOf()

pero sólo corre en ES5, no en ES3.

Prototype

Francisco Arce
www.pacoarce.com

Modificar objetos preestablecidos



Francisco Arce
www.pacoarce.com

Modificar los objetos predefinidos

Podemos añadir propiedades o métodos a los objetos propios de JS como Array, Object, Function, etc.

Douglas Crockford

cadena de prototipos o “prototype chaining”

Links

<https://www.packtpub.com/packtlib/book/Web%20Development/9781847194145/5>