Prototypal Inheritance

An introduction to Prototypal Inheritance

The extend method of UnderscoreJs adds all properties inherited via the prototype as own properties.

```
var proto = { protoProperty: 'proto' };
var o = Object.create( proto );

var c = _.extend({}, o);

console.log( 'o.hasOwnProperty( "protoProperty" )', o.hasOwnProperty( "protoProperty" ) );
// o.hasOwnProperty( "protoProperty" ) false

console.log( 'c.hasOwnProperty( "protoProperty" )', c.hasOwnProperty( "protoProperty" ) );
// c.hasOwnProperty( "protoProperty" ) true
```

While some shallow copy implementations behave in the same way, others completely omit properties coming from the prototype chain.

More often than not, the prototype should not be considered at all. In case of cloning objects with a prototype, make sure you consider how the clone method treats prototypes.