

A map from strings to any value except `undefined`.

It's not derived from a class.

```
{  
    key1 : val1,  
    ...  
}
```

Keys are automatically strings so they don't require quotes unless they are reserved words.

```
obj["quoted-key"]
```

```
obj.key
```

Accessing a non-existent key results in `undefined`.

Accessing a key of `undefined` results in a `TypeError`.

Through assignment on the keys.

Object.prototype

Using a special function.

```
if (typeof Object.create !== 'function') {  
    Object.create = function (o) {  
        var F = function () {};  
        F.prototype = o;  
        return new F();  
    }  
}
```

- For updates only the object's descendants get the new member, not the ancestors.
- For deletion only the object is affected.

- Functions are accessed, even when only data is desired. A conditional with `typeof` reflection can be used to filter undesired members.
- The entire prototype chain is accessed. `hasOwnProperty` can be used to discriminate inherited members.

Use `delete` on a member. It does not affect ancestors.

Use a single global object variable for all the program's state.