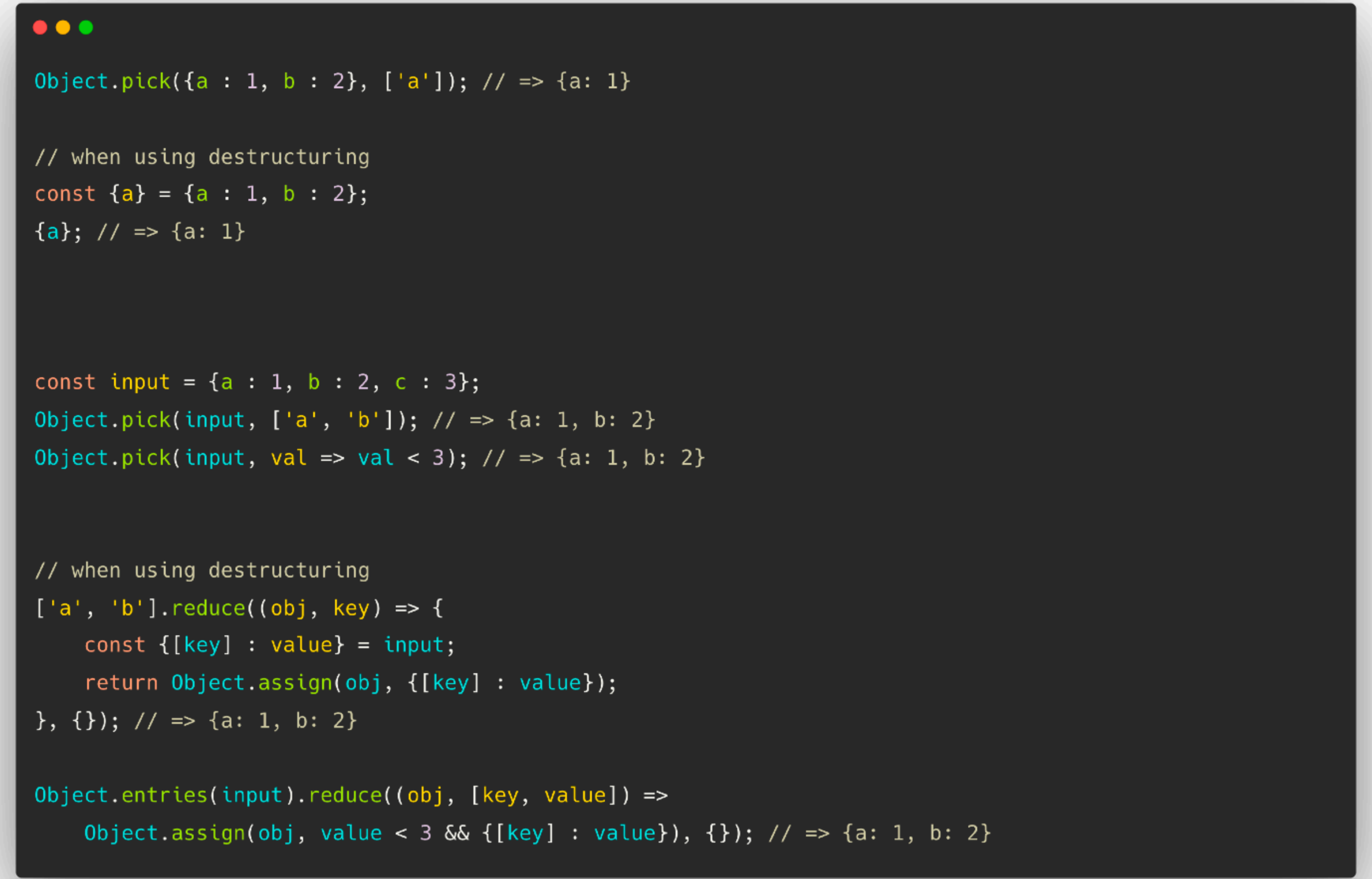
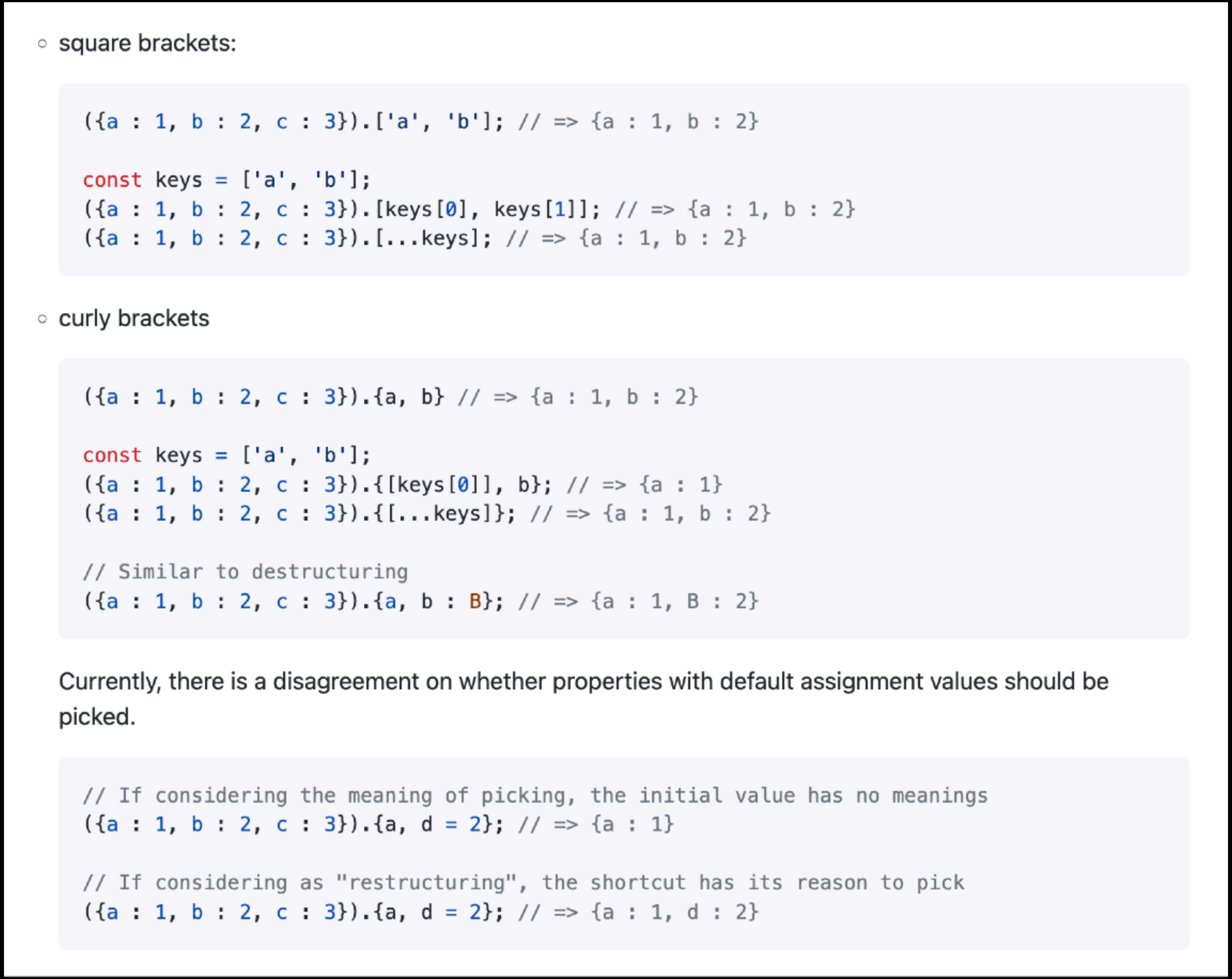


```javascript
Object.pick({a : 1, b : 2}, ['a']); // => {a: 1}

// when using destructuring
const {a} = {a : 1, b : 2};
{a}; // => {a: 1}



const input = {a : 1, b : 2, c : 3};
Object.pick(input, ['a', 'b']); // => {a: 1, b: 2}
Object.pick(input, val => val < 3); // => {a: 1, b: 2}


// when using destructuring
['a', 'b'].reduce((obj, key) => {
    const {[key] : value} = input;
    return Object.assign(obj, {[key] : value});
}, {}); // => {a: 1, b: 2}

Object.entries(input).reduce((obj, [key, value]) =>
    Object.assign(obj, value < 3 && {[key] : value}), {}); // => {a: 1, b: 2}
```

Visions



- square brackets:

```
({a : 1, b : 2, c : 3}).['a', 'b']; // => {a : 1, b : 2}

const keys = ['a', 'b'];
({a : 1, b : 2, c : 3}).[keys[0], keys[1]]; // => {a : 1, b : 2}
({a : 1, b : 2, c : 3}).[...keys]; // => {a : 1, b : 2}
```

- curly brackets

```
({a : 1, b : 2, c : 3}).{a, b} // => {a : 1, b : 2}

const keys = ['a', 'b'];
({a : 1, b : 2, c : 3}).{[keys[0]], b}; // => {a : 1}
({a : 1, b : 2, c : 3}).{[...keys]}; // => {a : 1, b : 2}

// Similar to destructuring
({a : 1, b : 2, c : 3}).{a, b : B}; // => {a : 1, B : 2}
```

Currently, there is a disagreement on whether properties with default assignment values should be picked.

```
// If considering the meaning of picking, the initial value has no meanings
({a : 1, b : 2, c : 3}).{a, d = 2}; // => {a : 1}

// If considering as "restructuring", the shortcut has its reason to pick
({a : 1, b : 2, c : 3}).{a, d = 2}; // => {a : 1, d : 2}
```