

# Using Simple Macro

macro as find-and-replace

x ← a

y ← b

```
(macro (swap x y)
  (let ([tmp x])
    (set! x y)
    (set! y tmp)))
```

```
(define a 1)
```

```
(define b 2)
```

```
...
```

```
(println a)
```

```
(println b)
```

→  
→  
→ 2

→  
→  
→ 1

```
(let ([tmp a])
```

```
(set! a b)
```

```
(set! b tmp)))
```



# Using Simple Macro

macro as find-and-replace

x ← a

y ← tmp

```
(macro (swap x y)
  (let ([tmp x])
    (set! x y)
    (set! y tmp)))
```

(define a 1)

(define tmp 2)

...

(println a)

(println tmp)

>>> 1

>>> 1

(let ([tmp a])

(set! a tmp)

(set! tmp tmp)))