

# Pass By Reference vs. Pass By Value



# php: linear arrays

```
<?php
```

```
$pickleNames = array(  
    'Gherkin',  
    'Polish',  
    'Hungarian',  
    'Swedish'  
);
```





# what will this print?

```
<?php
function foo($list) {
    $list[0] = 'Kosher';
}


$pickleNames = array(
    'Gherkin',
    'Polish',
    'Hungarian',
    'Swedish'
);

foo($pickleNames);
echo '<pre>';
print_r($pickleNames);
echo '</pre>'
```



# php

```
<?php
function foo($list) {
    $list[0] = 'Kosher';
}
```



```
$pickleNames = array(
    'Gherkin',
    'Polish',
    'Hungarian',
    'Swedish'
);
```

```
foo($pickleNames);
echo '<pre>';
print_r($pickleNames);
echo '</pre>'
```



# php

```
<?php
function foo($list) {
    $list[0] = 'Kosher';
}
```

```
$pickleNames = array(
    'Gherkin',
    'Polish',
    'Hungarian',
    'Swedish'
);
```

➔

```
foo($pickleNames);
echo '<pre>';
print_r($pickleNames);
echo '</pre>'
```





# php

➔

```
<?php
function foo($list) {
    $list[0] = 'Kosher';
}

$pickleNames = array(
    'Gherkin',
    'Polish',
    'Hungarian',
    'Swedish'
);

foo($pickleNames);
echo '<pre>';
print_r($pickleNames);
echo '</pre>'
```





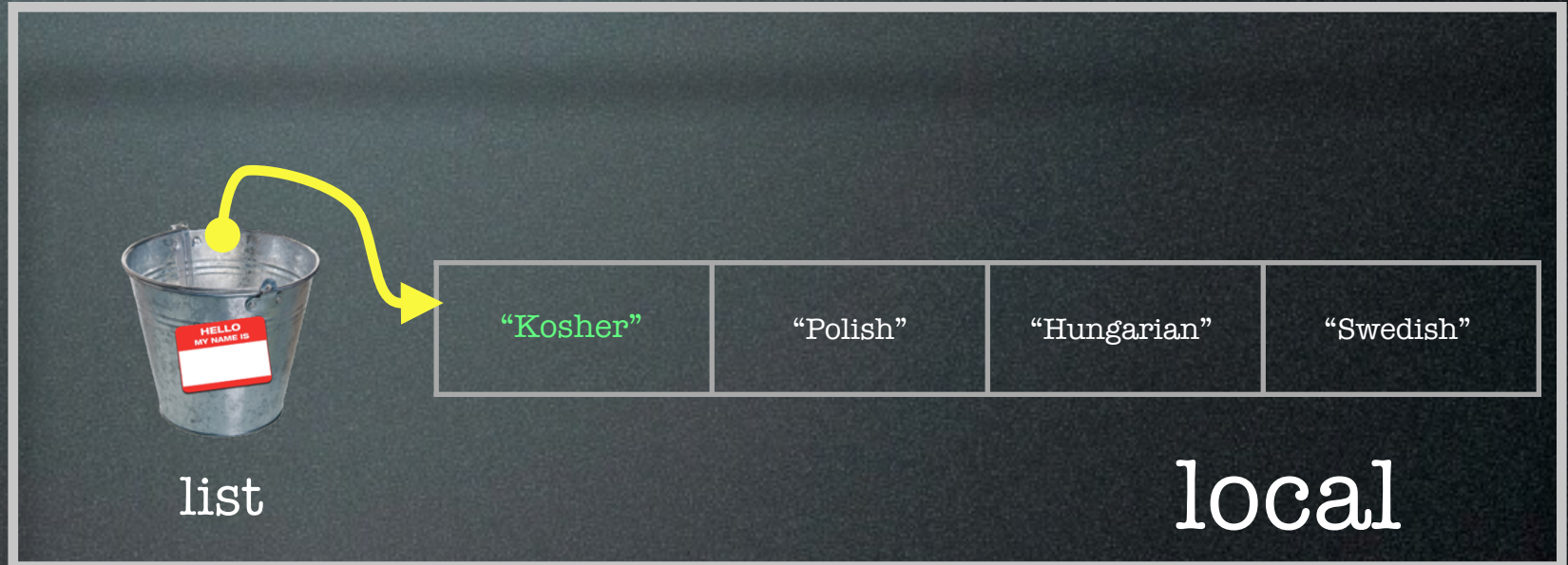
# php

➔

```
<?php
function foo($list) {
    $list[0] = 'Kosher';
}

$pickleNames = array(
    'Gherkin',
    'Polish',
    'Hungarian',
    'Swedish'
);

foo($pickleNames);
echo '<pre>';
print_r($pickleNames);
echo '</pre>'
```





# php

```
<?php
function foo($list) {
    $list[0] = 'Kosher';
}
```

```
$pickleNames = array(
    'Gherkin',
    'Polish',
    'Hungarian',
    'Swedish'
);
```

➔

```
foo($pickleNames);
echo '<pre>';
print_r($pickleNames);
echo '</pre>'
```





# php

```
<?php
function foo($list) {
    $list[0] = 'Kosher';
}

$pickleNames = array(
    'Gherkin',
    'Polish',
    'Hungarian',
    'Swedish'
);

foo($pickleNames);
echo '<pre>';
print_r($pickleNames);
echo '</pre>'
```

output:

Array

```
(
    [0] => Gherkin
    [1] => Polish
    [2] => Hungarian
    [3] => Swedish
)
```





# php

- php uses **pass by value** by default
- **copies** parameters passed to functions



# javascript

```
<script type="text/javascript">
function foo(list) {
    list[0] = 'Kosher';
}

var pickleNames = [
    'Gherkin',
    'Polish',
    'Hungarian',
    'Swedish'
];

foo(pickleNames);
console.log(pickleNames);
</script>
```



# javascript

```
<script>  
function foo(list) {  
    list[0] = 'Kosher';  
}
```

```
➡ var pickleNames = [  
    'Gherkin',  
    'Polish',  
    'Hungarian',  
    'Swedish'  
];  
  
foo(pickleNames);  
console.log(pickleNames);  
</script>
```



# javascript

```
<script>  
function foo(list) {  
    list[0] = 'Kosher';  
}
```

```
var pickleNames = [  
    'Gherkin',  
    'Polish',  
    'Hungarian',  
    'Swedish'  
];
```

➔ 

```
foo(pickleNames);  
console.log(pickleNames);  
</script>
```





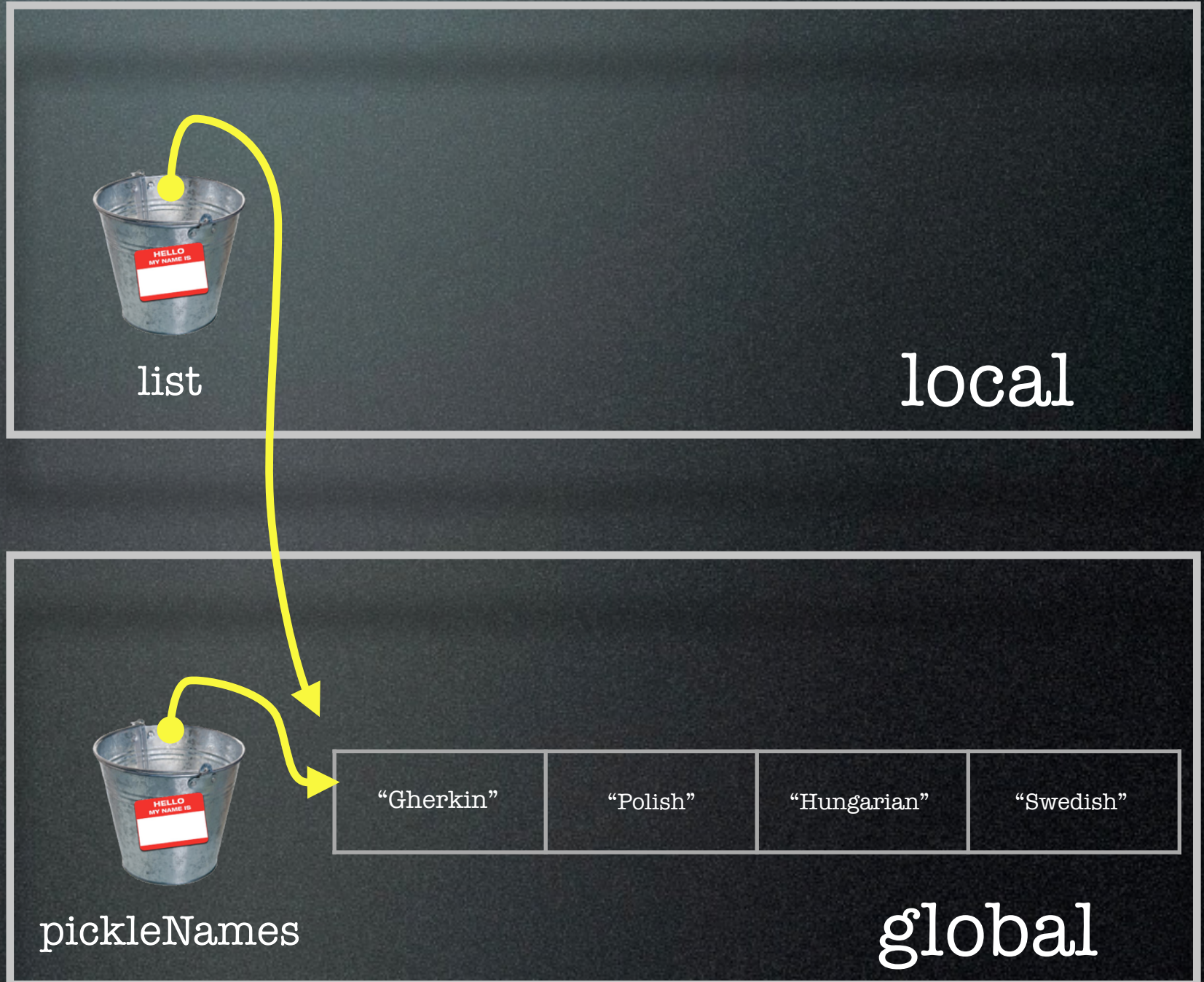
# javascript

➔

```
<script>
function foo(list) {
  list[0] = 'Kosher';
}

var pickleNames = [
  'Gherkin',
  'Polish',
  'Hungarian',
  'Swedish'
];

foo(pickleNames);
console.log(pickleNames);
</script>
```





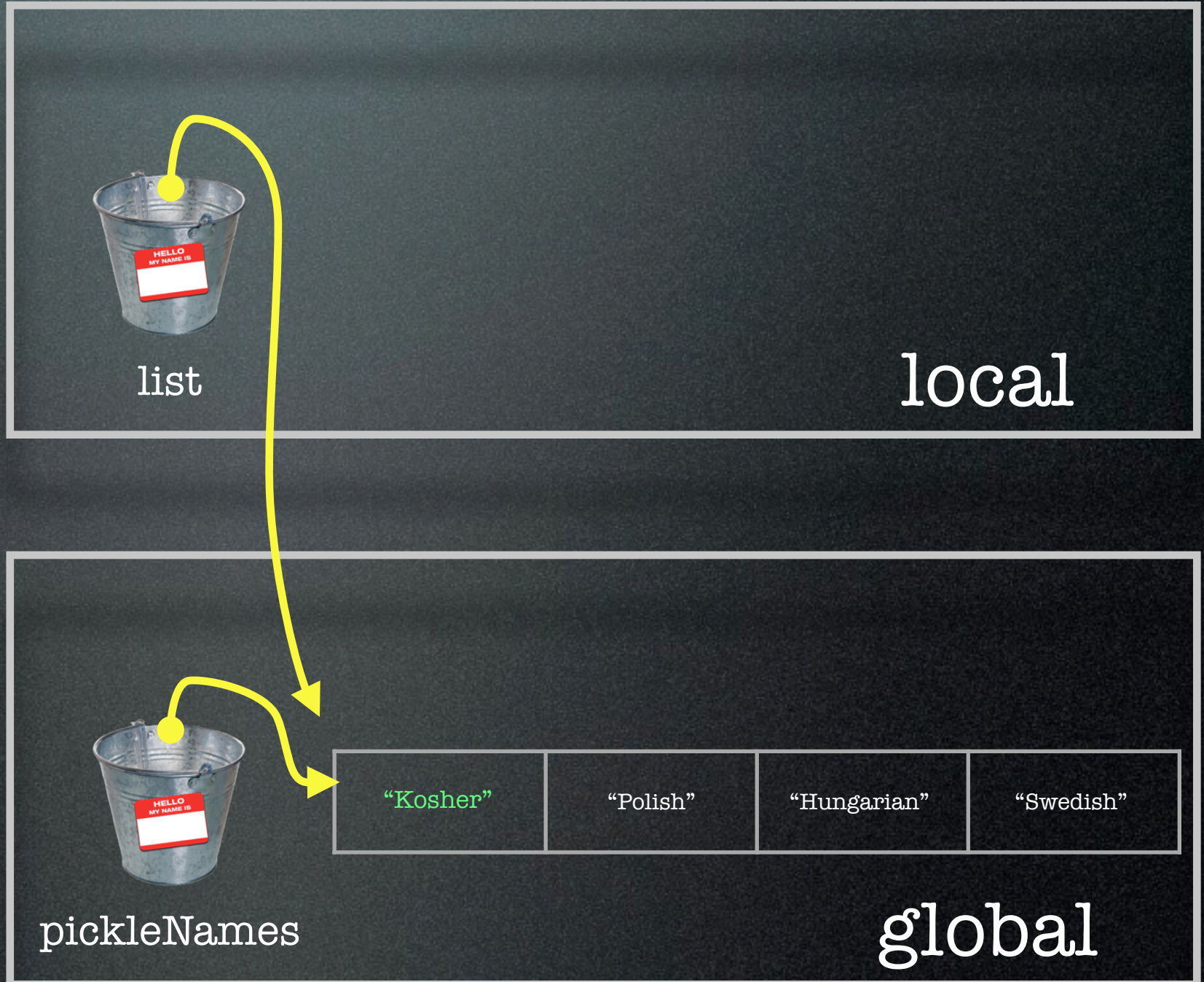
# javascript

➔

```
<script>
function foo(list) {
  list[0] = 'Kosher';
}

var pickleNames = [
  'Gherkin',
  'Polish',
  'Hungarian',
  'Swedish'
];

foo(pickleNames);
console.log(pickleNames);
</script>
```





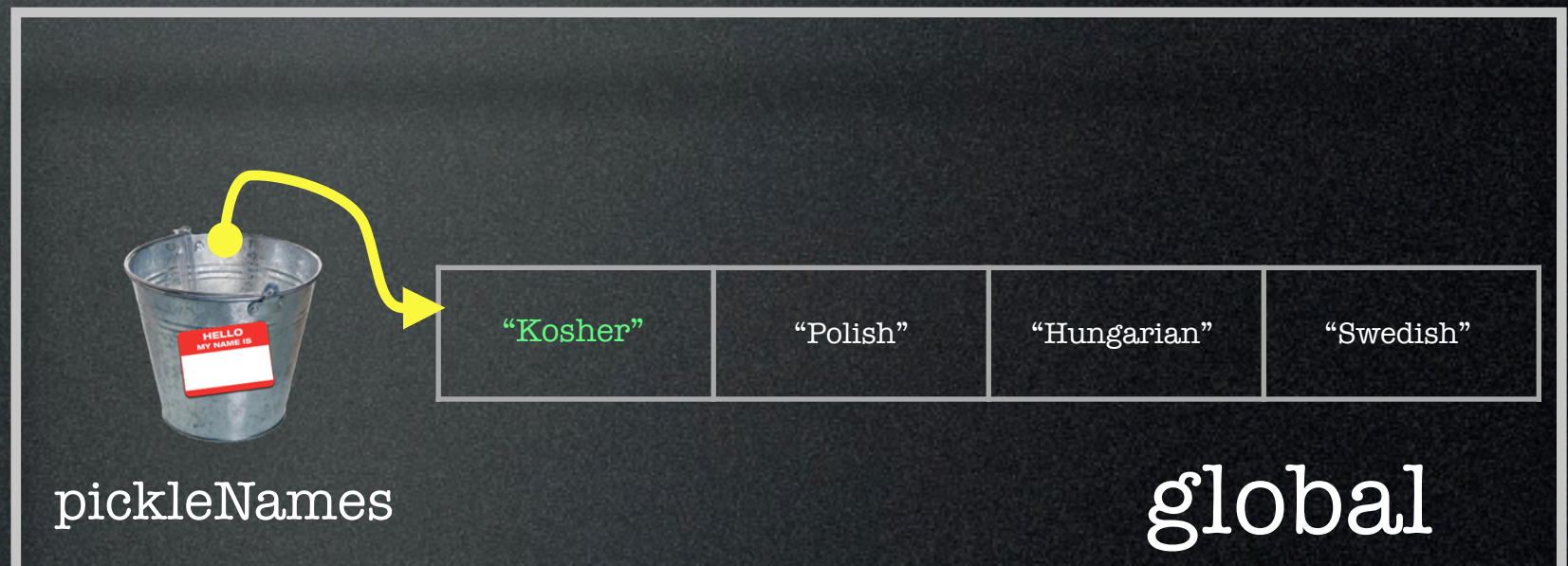
# javascript

```
<script>  
function foo(list) {  
    list[0] = 'Kosher';  
}
```

```
var pickleNames = [  
    'Gherkin',  
    'Polish',  
    'Hungarian',  
    'Swedish'  
];
```

➔ 

```
foo(pickleNames);  
console.log(pickleNames);  
</script>
```





# javascript

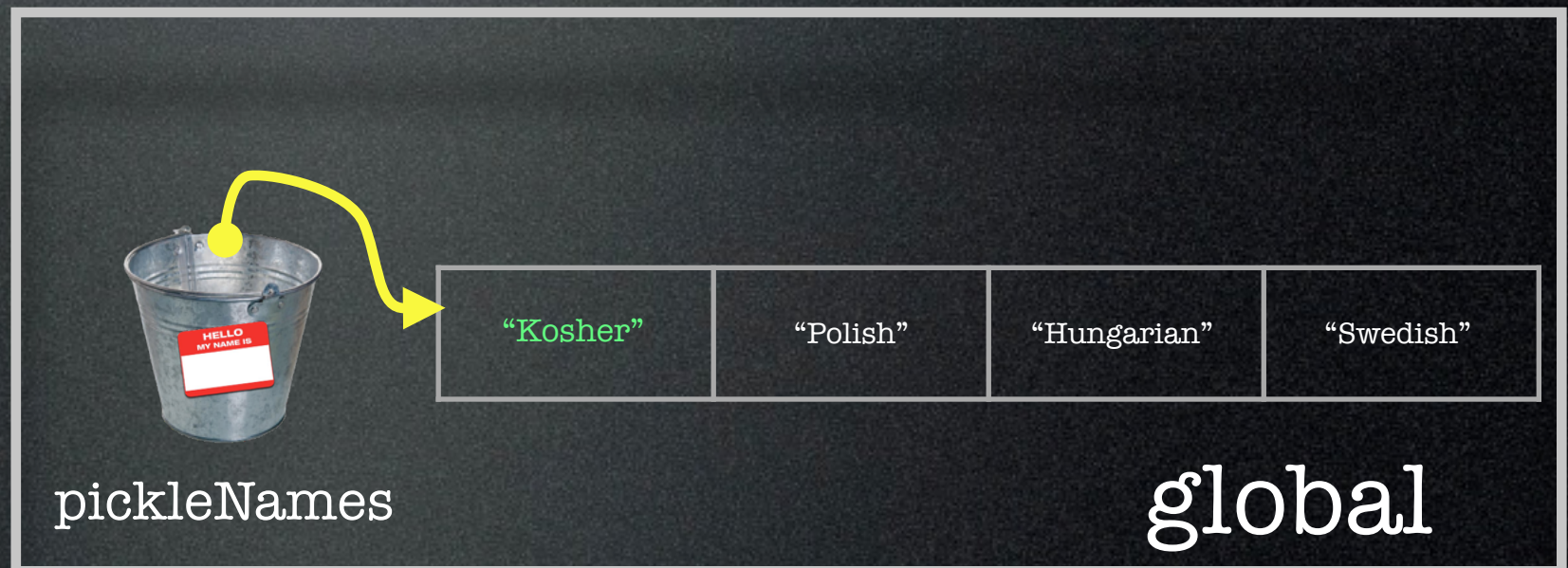
```
<script>
function foo(list) {
    list[0] = 'Kosher';
}

var pickleNames = [
    'Gherkin',
    'Polish',
    'Hungarian',
    'Swedish'
];

foo(pickleNames);
console.log(pickleNames);
</script>
```

output:

```
[
  'Kosher',
  'Polish',
  'Hungarian',
  'Swedish'
]
```





# javascript

- javascript uses **pass by reference** by default
- a **reference** to non-atomic values are passed as parameters



# javascript

- actually, every variable in javascript is best thought of as a reference to a single object
- if you want to clone/copy some data, you must do it explicitly



# php pass by reference

- you can pass by reference in PHP with the & operator



# php pass by reference

```
<?php
function foo(&$list) {
    $list[0] = 'Kosher';
}
```

```
$pickleNames = array(
    'Gherkin',
    'Polish',
    'Hungarian',
    'Swedish'
);
```

```
foo($pickleNames);
echo '<pre>';
print_r($pickleNames);
echo '</pre>'
```

ampersand says:  
this parameter is  
pass-by-reference

this should now work  
like the javascript  
example



# php iteration and pass by reference

```
<?php
```

```
$pickleNames = array(  
    'Gherkin',  
    'Polish',  
    'Hungarian',  
    'Swedish'  
);  
  
foreach ($pickleNames as $name) {  
    $name =  
}  
  
echo '<pre>';  
print_r($pickleNames);  
echo '</pre>'
```