

jQuery Quiz II

1. Which, if any, of the following 3 code fragments are equivalent? Explain why they are different, if they are. Explain why they can have different parameters and be equivalent, if they are equivalent.

```
//code fragment 1
$("li").each(function(idx, e) {
    $(e).css("color", "yellow"); });
```

```
//code fragment 2
$("li").each(function() {
    $(this).css("color", "yellow"); });
```

```
//code fragment 3
$("li").each(function(idx) {
    $(this).css("color", "yellow"); });
```

The each() method specifies a function to run for each matched element, and it takes a callback function.

The syntax for the each() method is :

```
$(selector).each(function(index,element){
    //code to be executed
});
```

So, in the code fragments above, all codes are doing the same thing. In code fragment 1 and 2, both are referring the current element in the loop. In code fragment 1, the "this" keyword is passed as a second argument to the each() function and is accessed through the "e" parameter in the callback function. In code fragment 2, the "this" keyword is used directly in the callback function. In code fragment 3, the additional parameter added in the callback function is not being used in the callback function, that means it doesn't affect the behavior of the code and can be omitted without changing the result. Therefore, all three code fragments are equivalent.

2. Write a jQuery expression to find all *divs* on a page that include an unordered list in them, and make their text color be blue.

```
<div>no ul here </div>
<div>
  This does contain a ul.
  <ul>
    <li>the first item</li>
    <li>the second item</li>
  </ul>
</div>
<script>

  $(document).ready(function() {
    $("div:has(ul)").css("color","blue");});
</script>
</body>
```

3. Write jQuery code to append the following *div* element (and all of its contents) dynamically to the *body* element.

```
<div><h1>jQuery Core</h1></div>
```

HTML:

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>JS Bin</title>
</head>
<body>

</body>
</html>
```