

PHP JSON not returning value from php function

Asked 3 years, 10 months ago Modified 3 years, 10 months ago Viewed 77 times

 Part of [PHP](#) Collective



Say I have a function as following. `alert_danger` returns the error message in red box. `check_empty` checks if a value posted from form is empty or not.

-1



```
function alert_danger($msg){
    $alert = "<div class='alert alert-danger' id='responseBox'>".$msg."</div>";
    return $alert;
}

function checkEmpty($postValue, $msg){
    if($postValue == null){
        echo alert_danger($msg);
        exit();
    }
}
```

Now when I want to return the function value using `json` it's not returning the same. The following error is occurring:

```
// It returns this
$msg = alert_danger("Ah! Hello Adventurer, and welcome to the town of
Honeywood!");
echo json_encode(array('status' => $msg));

// But it does not returns this
$msg = checkEmpty($state, "Ah! Hello Adventurer, and welcome to the town of
Honeywood!");
echo json_encode(array('status' => $msg));
```

What seems to be the problem here?

Here is my `jQuery` if needed!

```
$(".action").click(function() {
    var form = $(this).closest("form");
    var type = form.find(".type").val();
    var dataString = form.serialize();
    var btnValue = $(".action").html();
    var btnElement = $(".action");
    var url = form.attr("action");
    $.ajax({
        type: "POST",
        dataType : "json",
        url: url,
        data: dataString,
        cache: true,
```

```

beforeSend: function(){
    $('.message').hide();
    $(".overlay").show();
    $(".wickedpicker").hide();
    btnElement.html('Please wait...');
},
success: function(json){
    $(''.message').html(json.status).fadeIn();
    // $('#content').html(json.result).fadeIn();
    $(".overlay").hide();
    $("html, body").animate({ scrollTop: $(".message").offset().top },
"slow");
    btnElement.html(btnValue);
    if(type == 'admin'){
        if($('.message').find('#responseBox').hasClass('alert-success')){
            setTimeout(function(){
                $(".overlay").hide();
                window.location.replace("dashboard.php");
            }, 1000);
        }
    }
}
});
return false;
});

```

PHP

php

jquery

json

ajax

function

Share

edited Feb 17, 2021 at 19:42

Improve this question

Follow

asked Feb 17, 2021 at 19:24



Shubham Jha

97 ● 1 ● 11

where is `alert_danger` function? How are you calling `checkEmpty` ? But the `checkEmpty` does not return any value... so assigning `$msg = checkEmpty` is meaningless – [Professor Abronsius](#) Feb 17, 2021 at 19:29

does `alert_danger` return a value? – [Professor Abronsius](#) Feb 17, 2021 at 19:35

- 1 Your function does not `return` any content. This would make `$msg` be `null` – [Twisty](#) Feb 17, 2021 at 19:37

@ProfessorAbronsius I have updated the question. Please check. I thought I did not need to mention `alert_danger()` function there so I didn't but since you demanded it I have updated it. – [Shubham Jha](#) Feb 17, 2021 at 19:43

@Twisty my function does return a value. The things are working perfect when I do not use JSON. But if I use JSON then it's not returning any value. – [Shubham Jha](#) Feb 17, 2021 at 19:45

2 Answers

Sorted by: Highest score (default)





Consider the following.

1



PHP

```
<?php
function checkEmpty($postValue, $msg){
    return $postValue == null ? array("status" => "error", "message" => "Empty
Value") : array("status" => $postValue, "message" => $message);
}

header('Content-Type: application/json');
echo json_encode(checkEmpty($state, "Ah! Hello Adventurer, and welcome to the
town of Honeywood!"));
?>
```

JavaScript

```
function redirectTo(url, time) {
    if (!url) {
        return false;
    }
    time = time != undefined ? time : 0;
    setTimeout(function() {
        window.location.href = url;
    }, time);
}

$(".action").click(function() {
    $(this).closest("form").submit();
});

$("form").submit(function(event) {
    event.preventDefault();
    var type = $(this).find(".type").val();
    var dataString = $(this).serialize();
    var btnValue = $(".action").html();
    var btnElement = $(".action");
    var url = $(this).attr("action");
    $.ajax({
        type: "POST",
        dataType: "json",
        url: url,
        data: dataString,
        cache: true,
        beforeSend: function() {
            $('.message').hide();
            $(".overlay").show();
            $(".wickedpicker").hide();
            btnElement.html('Please wait...');
        },
        success: function(json) {
            if (json.status == "error") {
                $(".message").html("<div class='alert alert-danger error'>" +
json.message + "</div>").fadeIn();
            } else {
                $('.message').html("<div class='alert alert-danger'>" + json.message +
"</div>").fadeIn();
            }
        }
    });
});
```

```

    $("html, body").animate({
        scrollTop: $(".message").offset().top
    }, "slow");
    btnElement.html(btnValue);
    if (type == 'admin') {
        if ($('.message').find('#responseBox').hasClass('alert-success')) {
            redirectTo("dashboard.php", 1000);
        }
    }
}
});
return false;
});

```

Typically, it is bad practice to use language X to generate code in language Y. Try decoupling the two languages by making data their only interface -- don't mingle the code.

<https://softwareengineering.stackexchange.com/questions/126671/is-it-considered-bad-practice-to-have-php-in-your-javascript>

You have to be careful to not confuse `echo` and `return`, they do very different things.

<https://www.php.net/manual/en/function.echo.php>

<https://www.php.net/manual/en/function.return.php>

Since you're passing back JSON data to the AJAX Call, I would advise wrapping your HTML inside the callback versus sending it back inside the JSON.

Share

edited Feb 17, 2021 at 22:38

answered Feb 17, 2021 at 19:43

Improve this answer



Twisty

30.8k ● 2 ● 29 ● 49

Follow



-1



I think you should take a look at your success function. I think it normally runs before the page loads. So, its possible none of the html your referencing in there exists yet. So move it out to a function like this:

```

success: function(json) {
    doSomething();
}

function doSomething(json){
    $( document ).ready(function() {
        console.log('page has loaded now modify your html with jquery'+json);
    }
}

```

Share Improve this answer Follow

answered Feb 17, 2021 at 22:48



[Jim VanPetten](#)

436 ● 3 ● 11

obviously we all use the shortcode now: `$(function() { console.log("ready!"); });`

– [Jim VanPetten](#) Feb 17, 2021 at 22:49
