

# LESSON 15: SENDING AND RECEIVING DATA

## SENDING DATA

We often send data to a service from one of these situations:

1. We \_\_\_\_\_ data in our script based on settings we know we need for the service such as filters or sorts for a GET ALL request.
2. We use some variable from the \_\_\_\_\_ in order to obtain a specific request such as a GET One, PUT, or DELETE.
3. We \_\_\_\_\_ data from a \_\_\_\_\_ to send along with an POST or PUT request.

### 1. *Compiling Data Manually*

```
$.ajax("user_lego/", {  
    type: "GET",  
    data: {  
        filter: 'user=1',  
        sort: [  
            'type.name',  
            'lego.name'  
        ]  
    },  
    ...  
});  
  
var req_data = {  
    filter: 'user=1',  
    sort: [  
        'type.name',  
        'lego.name'  
    ]  
};  
$.ajax("user_lego/", {  
    type: "GET",  
    data: req_data,  
    ...  
});
```

# [Scripting]

## 2. Using System Variables

```
<a href="#" data-item="3" class="lego-thumb">...</a>
```

```
$(".lego-thumb").on("click", function(e){  
    e.preventDefault();  
    var $target = $(e.target);  
    var itemId = $target.attr("data-item");  
    $.ajax("lego/" + itemId, {  
        ...  
    });  
});
```

## 3. Serializing Form Data

Assuming you start with a form whose fields' names all match the corresponding entity properties...

```
$("#form-id").on("submit", function(e){  
    e.preventDefault();  
    var $form = $(e.target);  
    var form_data = $form.serialize();  
    $.ajax("lego/", {  
        type: "POST",  
        data: form_data,  
        ...  
    });  
});
```

# [Scripting]

## RECEIVING DATA

We most often receive data from a service in the form of JSON text. We first must \_\_\_\_\_ it to an actual JSON object and then we can use it for \_\_\_\_\_ or to populate a \_\_\_\_\_.

```
var __1__ = $.parseJSON(__2__);
```

1. Desired name for parsed data variable.
2. JSON string variable to parse.

```
$.ajax("user_lego/", {  
    ...  
    success: function(data, xhr) {  
        data = $.parseJSON(data);  
        $.get("lego-tiles.tpl.html", function(tplStr) {  
            var tpl = Handlebars.compile(tplStr);  
            var tplHTML = tpl(data);  
            $("#tiles").html(tplHTML);  
        });  
    }  
});
```