

## What is lightning Component Attribute : Aura:attribute

**Lightning Component Attribute** (aura:attribute) are like member variables on a class in Apex. `<aura:attribute>` tag is used to add an attribute to the [lightning component](#) and [Salesforce lightning App](#).

- Each attribute must have “**name**” and “**type**”.
- To make attribute as required use `required="true"` in the tag.
- To make default value use `default="string or integer"`.

### Attribute naming Rules.

An attribute name must follow these naming rules.

- Must begin with a letter or an underscore.
- Must contain only alphanumeric or underscore characters.

## What is lightning **Component** Attribute : Aura:attribute

In this [Salesforce tutorial](#), we are going to learn about Lightning Component attributes and how can to use `<aura:attribute>` tag in Salesforce lightning app and lightning component.

- *In this Salesforce example, we are going to create an application and a component called “addition”. Here we will add two integers and will find the sum of two integers. And also we add styling to the Salesforce lightning component.*
- Navigate to [Developer console](#) and create new Lightning component.

Here we have to define two tags they are `<aura:component>` and `<aura:attribute>`. This lightning Component Attribute will have name and type.

```
<aura:component >  
  <h1> Addition of
```

```

<aura:component >
  <h1> Addition of Two Numbers</h1>
  <aura:attribute name="num1" type="integer" default="100" />
  <aura:attribute name="num2" type="integer" default="20" />
  <aura:attribute name="sum" type="integer" />
  <br></br>
  <div class="addC" >
    {!v.num1} + {!v.num2} = {!v.sum}
  </div>
  <br></br>
  <p>
    <ui:button label="submit" press="{!c.add}" />
  </p>
</aura:component>

```

```

File Edit Debug Test Workspace Help <> Component name
addition.cmp firstlightningController.js lightning.app additionController.js additionHelper.js addition.css
1 <aura:component >
2 <h1> Addition of Two Numbers</h1>
3 <aura:attribute name="num1" type="integer" default="100" />
4 <aura:attribute name="num2" type="integer" default="20" />
5 <aura:attribute name="sum" type="integer" />
6 <br></br>
7 <div class="addC" >
8     {!v.num1} + {!v.num2} = {!v.sum}
9 </div>
10 <br></br>
11 <p>
12     <ui:button label="submit" press="{!c.add}" />
13 </p>
14
15 </aura:component>

```

As shown above, we have selected type as integer because we are storing numbers. And if we want to provide default values enter default="value". To display two numbers we have to use particular expression like `{!v.num1} + {!v.num2} = {!v.sum}`.

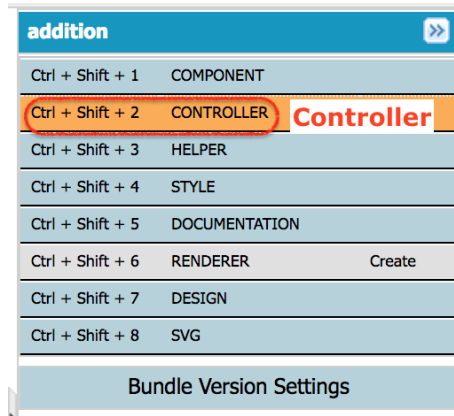
- We also defined a button called submit. When a submit button is clicked the result will be displayed on the screen.
- The sum value will not be displayed in the output when we click on submit because we have not defined what should happen when submit button is clicked.

```

File Edit Debug Test Workspace Help <> Controller
addition.cmp firstlightningController.js lightning.app additionController.js * additionHelper.js addition.css
1 ( {
2     add:function(component) {
3         var addval=component.get("v.num1")+component.get("v.num2");
4         component.set("v.sum", addval);
5     }
6 } )

```

Here we have to add a call a method called `{!c.add}`. Now click on controller in the component and add the controller as shown below.



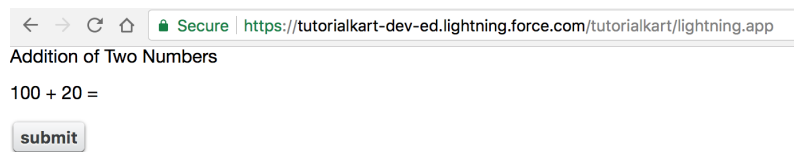
When we click on **Controller**, Controller file will be created automatically. We want to call `{!c.add}` function.

```
{
  add:function(component) {
    var addval=component.get("v.num1")+component.get("v.num2");
    component.set("v.sum",addval);
  }
})
```

- Click on **Save** button.

## lightning **Component** Attribute output.

When we preview the lightning application, the output will be as follow.



The look and feel of the output will be changes by adding style to lightning component. Click on Style and add the following css code given below.

```
.THIS {
  background-color:
```

```
.THIS {
  background-color: white;
  font-family:sans-serif;
  text-align:center;
}

body.THIS {
  border: 2px solid grey;
}

H1.THIS {
  background-color: white;
  font-weight:strong;
  width: 20%;
  border: 1px solid green;
  padding: 10px;
  font-family:sans-serif;
  text-align:center;
  margin:auto;display:block;
}

body.THIS {
  text-align:center;
}

p.THIS {
  text-align: center;
}

button.THIS {
  position: absolute;
  top: 50%;
}

addC.THIS {
  margin:auto;
  display:block;
}
```

**Output :**

## Addition of Two Numbers

$$100 + 20 = 120$$

submit

**Conclusion** : In this Salesforce tutorial, we have learned about lightning Component Attribute and created Salesforce lightning component for addition of two integer. In our upcoming [Salesforce training](#), we will learn about Salesforce lightning Component composition with an example.

Home : [Salesforce Admin Tutorials](#)

↳ [Salesforce Administration](#)

Section 1 : [Getting Started](#)

↳ [What is Salesforce.com](#)

↳ [What is login.Salesforce.com](#)

↳ [Enabling Caching and Automcomplete at Salesforce login](#)

↳ [Overview of ERP and CRM](#)

↳ [What is Cloud Computing](#)

↳ [Why Salesforce CRM is #1 on Demand](#)

Section 2 : [Designing your Data Model](#)

Section 3 : [Building your Data Model](#)

↳ [Understanding Salesforce.com application](#)

↳ [How to create an App in SFDC?](#)

↳ [How to create custom object in SFDC](#)

↳ [Creating fields using different field types in Salesforce.com](#)

↳ [Overview on Salesforce Object Relationships](#)

↳ [How to create Master Detail relationship in SFDC?](#)

↳ [Cannot create Master Detail Relationship?](#)

↳ [How to create Lookup relationship in salesforce.com?](#)

↳ [How to create Many to Many Relationship in Salesforce?](#)

↳ [What is Schema Builder](#)

Section 4 : [Implementing Business Logics](#)

↳ [What is a Formula Filed?](#)

↳ [What are Cross Object Formulas?](#)

‡ What are Validation Rules?

‡ Roll up Summary Fields

‡ Field Dependencies

## Section 5 : Setting up User Interface

‡ Page Types and Page Elements

‡ What are page Layouts?

‡ Home page layouts and components

‡ Why Documents are used in Salesforce

‡ List Views

## Section 6 : Customizing Standard Salesforce Application

‡ How to Customise Standard Salesforce application

## Section 7 : Salesforce Security Model

‡ Overview on Salesforce security Model

‡ How to create New Users in SFDC?

‡ What is SAML?

‡ Public groups in Salesforce

‡ Role Hierarchies in Salesforce.com

‡ How to create and manage Profiles

‡ What is Field Level Security?

‡ How to use Permission Sets?

‡ Control Record visibility using Organisation wide Defaults

‡ Control Record visibility using Role Hierarchy

‡ Control Record visibility using Sharing Rules

‡ What are Record Types and how can we configure

‡ What are Page Types and page Elements

## Section 8 : Data Management

‡ Different Data Management Tools

‡ How to install Apex Data Loader in macOS and Windows

‡ How to Insert a record using Apex Data Loader

‡ How to Update and Insert records Apex Data Loader

‡ How to Delete and Export records Apex Data Loader

## Section 9 : Salesforce.com Audit

‡ Field History Tracking

▸ Setup Audit Trail

## Section 10 : Automate Business Process

▸ How to create New Email Template in Salesforce

▸ Salesforce Workflow Rules Overview

▸ Creating Workflow rules with Rule Criteria & Workflow actions

▸ Configuring Approval Process

▸ Assignment rules for Leads and Cases

## Section 11 : Reports and Dashboards

▸ What is a report in Salesforce and how we create them?

▸ Salesforce Reports and Dashboards Overview

▸ How to use report builder?

▸ How to create Summary reports?

▸ How to create matrix reports?

▸ How to create salesforce Dashboard, Different Dashboard Components, Customizing Dashboard Components, How to create Dynamic Dashboard in Salesforce, Deleting and Printing Salesforce Dashboards

▸ Creating Dashboards with Dashboard Components for Tabular and Joined Reports

## Section 12 : Salesforce Service Cloud

▸ Overview on Salesforce Service Cloud

## Section 13 : Portal and Sites

## Section 14 : AppExchange

▸ What is Salesforce AppExchange.

## Section 15 : Managing Sandboxes

▸ What is Salesforce Sandbox?.

▸ Different Sandboxes and Sandbox Environment Types.

▸ How to create Salesforce Sandbox template?.

▸ How to create Salesforce Sandbox?

▸ How to login Salesforce Sandbox?

## Section 16 : Salesforce lightning

▸ How to create Salesforce custom domain?

▸ What is Salesforce lightning Experience?

▸ What is Salesforce lightning component Framework?

▸ Creating first Salesforce lightning App

▸ Styling Salesforce lightning App

↳ Lightning Component : aura:attribute tag

↳ What is Salesforce DX?

↳ Create Salesforce DX project

↳ Salesforce Dev Hub Setup step-by-step

↳ Creating Scratch Org