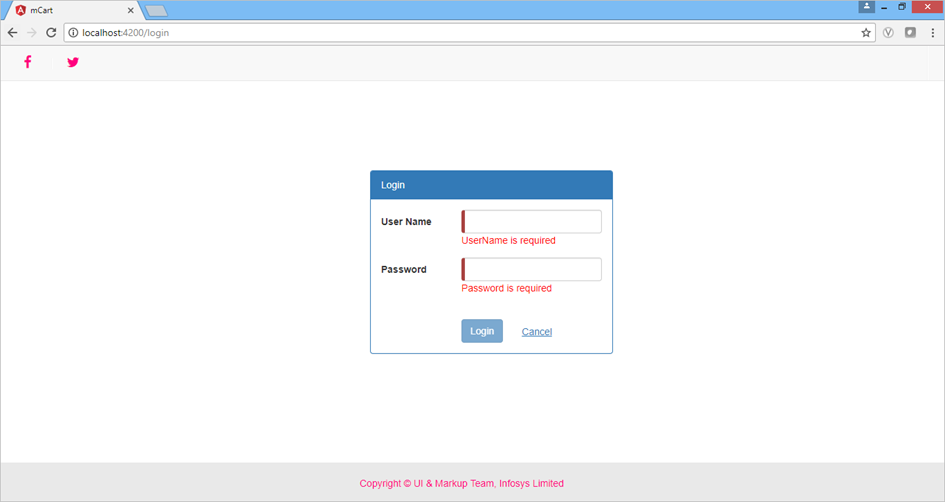
Now we will learn how to create forms in Angular.

We will see two styles of implementing forms and also how to perform validations.

mCart wants to create a login form with validations on username and password fields as shown below:



This can be done by creating either a ***template-driven***form or a ***reactive***form. Let us see each one of these in detail.

Forms are a crucial part of web applications through which we get a majority of data input from users. We can create two different types of forms in Angular

* **Template-driven forms** - Used to create small to medium-sized forms
* **Model-driven Forms or Reactive forms** - Used to create large forms

**Template-driven forms**

Template-driven forms use **ngForm**and **ngModel**directives to get information about the form and its controls and **ngSubmit**event to submit the forms.

* **ngForm:** Provides information about the current state of the form including a JSON representation of the form value and the validity state of the entire form
* **ngModel:** Provides two-way data binding between the view and component. It is also used to track the state and validity of the input field.
* **ngSubmit:** Fires an event specified by ngSubmit when the form is submitted

**Syntax:**

1. <form #formRef = "ngForm" (ngSubmit) = "onSubmit()">
2. <input class="form-control" name="controlname" [(ngModel)]="classVariable" #variable ="ngModel"/>
3. </form>

Here,

**#formRef**="ngForm" is a template variable that has an instance of *ngForm* or Angular from.

**(ngSubmit) = "onSubmit()" :** On submitting the form, onSubmit() method gets invoked.

**[(ngModel)] = "classVariable":** Binds the data entered in the input field of the form to the class variable specified in the corresponding **.ts** file

**#variable = "ngModel":** Another template variable which is bound to an instance of *ngModel.* This helps to track the state of the input field by checking if the user touched the control, if the value changed, or if the value became invalid.

You can store the input value accepted through the form and store it in the browser using [web storage API](https://lex.infosysapps.com/viewer/lex_10627728399173825000?ref=%252Ftoc%252Flex_12114348068735869000&courseId=lex_12114348068735869000).

You will learn to create a template-driven form in the next demo.