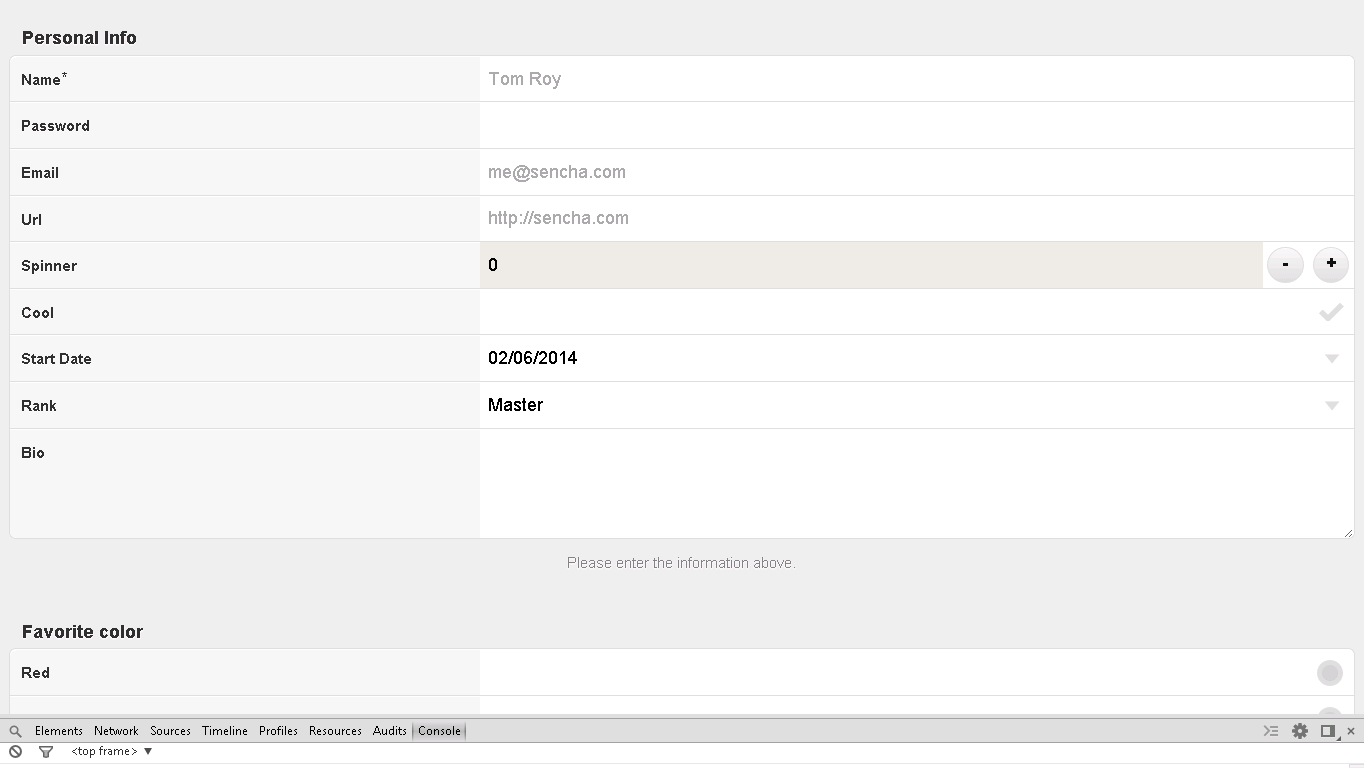
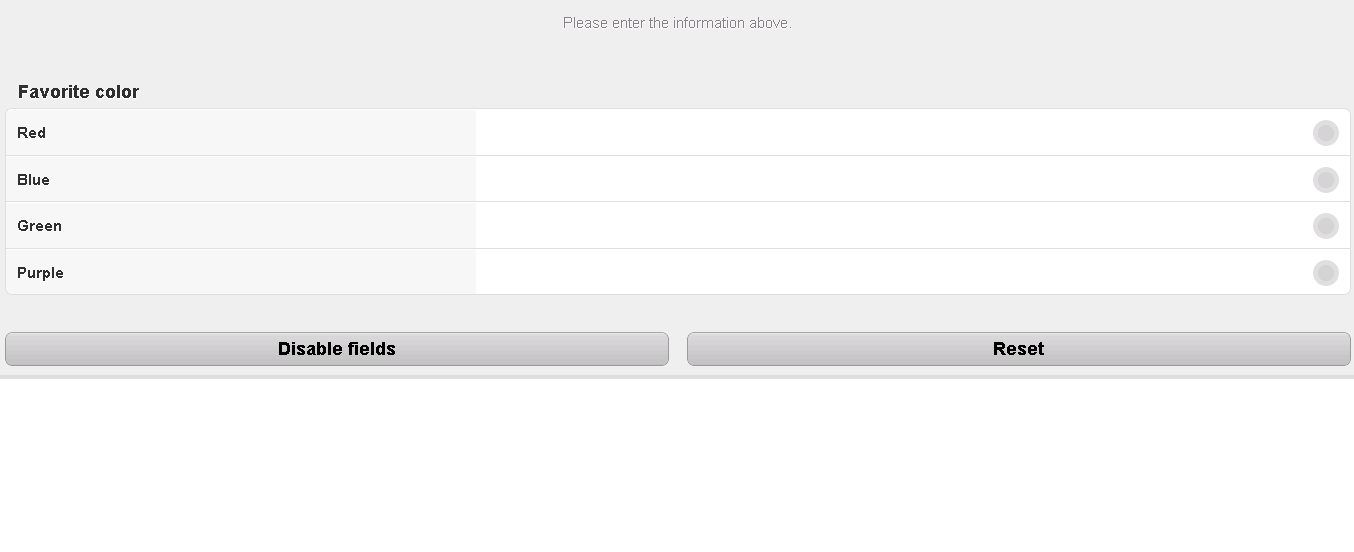
**Form panel:**

This example shows normal form for personal info, which will cover different component’s which can be used in form.

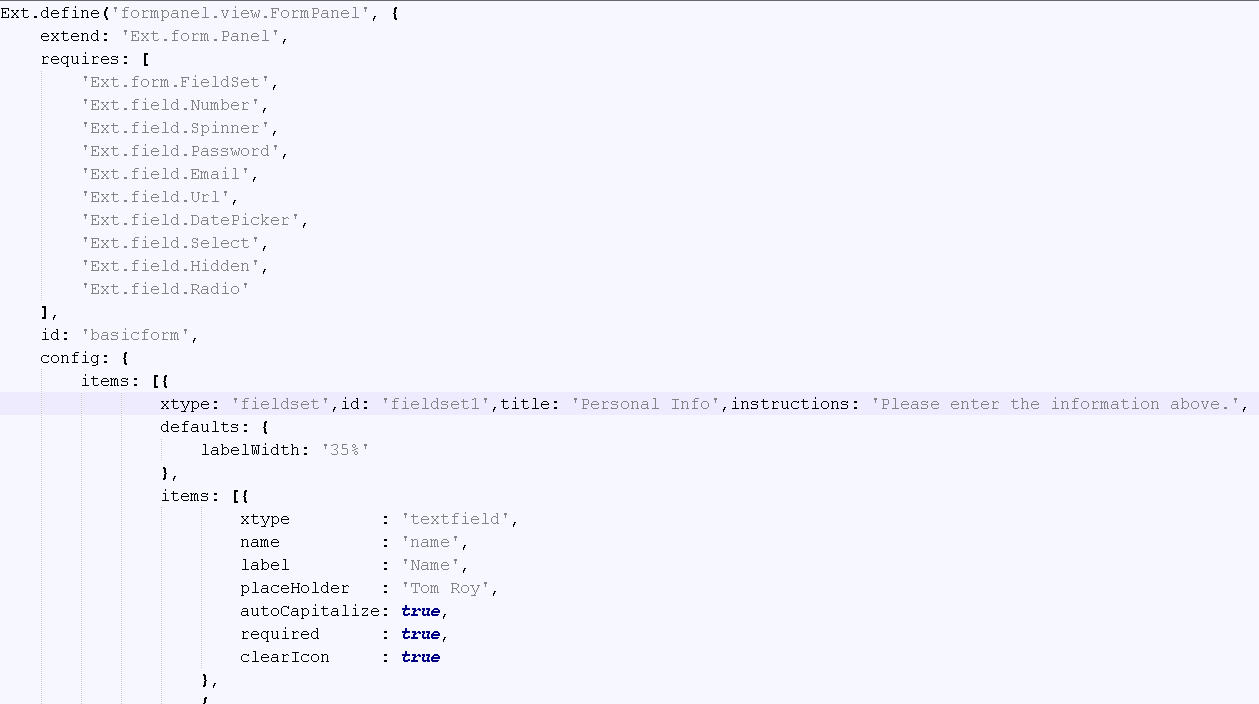


**Snapshot 1.1**



**Snapshot 1.2**

**Following is code snippet for form panel example of FOS:**

****

**Snapshot 1.3**

* Form panel application extends superclass Panel to use in panel in this application.

**Ext.form.Panel**

* This requires following classes to use in application:

Ext.form.FieldSet - To use field Set in application.

Ext.field.Number - - To use Number in application

Ext.field.Spinner - To use spinner in application with which you can increase and

decrease number by clicking plus or minus button

Ext.field.Password - To use password property in application which hides password you provide by showing black dots instead of text for security.

Ext.field.Email – To use Email field in form.

Ext.field.Url - To use URL field in form.

Ext.field.DatePicker –To use date picker which will allow you to select date at run time.

Ext.field.Select – To use selectfield in form which allows you to select one field from

Given list.

Ext.field.Hidden -

Ext.field.Radio - To use Radio field in form which allows you to select only field at time

Form given options.

* ID is provided to every sencha class to identify it individually, you can give any name as required.

**id: 'basicform',**

* Any sencha code starts with Config{} block which contains whole actual code.
* Coding starts with directly adding adding item as form requires fieldset at base:

items: [

{

xtype: 'fieldset',

id: 'fieldset1',

title: 'Personal Info',

instructions: 'Please enter the information above.',

defaults: {

labelWidth: '35%'

},

* Above is default base item which shows xtype i.t type of component to be used as fieldset, it is provided with id, title and intrustion for whole form, you can change it as per requirement.
* Label width is assign as 35% for complete form.

This particular Item fieldset contains sub items which will on this fieldset.

**Following code snippet shows sub items added on fiedset:**



**Snapshot 1.4**

* First item is **text field** which has following properties:

xtype : 'textfield' – Defines component as textfield

name : 'name' -> Gives name to texfield

label : 'Name'- assign label to textfield

placeHolder : 'Tom Roy' – Shows default text in textfield

autoCapitalize: true – Auto capitalises text field

required : true – shows that particular field is mandatory

clearIcon : true – Enables user to clear text field to default text.

* **Passwordfield :**

{

xtype: 'passwordfield' - Defines component as password field

name : 'password'- Gives name to password field

label: 'Password'- Gives name to password field (You can change label)

},

* **Emailfield**

{

xtype : 'emailfield' - Defines component as email field

name : 'email', - Gives name to email field

label : 'Email' - Gives label to password field (You can change label)

placeHolder: 'me@sencha.com' - Shows default text in email field

clearIcon : true - – Enables user to clear email field to default text

},

* **urlfield**

{

xtype : 'urlfield' - Defines component as url field

name : 'url, - Gives name to email field

label : 'Url' - Gives label to password field (You can change label)

placeHolder: 'http://sencha.com' - com' - Shows default text in url field

clearIcon : true - Enables user to clear url field to default text

},

* **spinnerfield**

{

xtype : 'spinnerfield' - Defines component as 'spinnerfield

name : 'spinner' - Gives name to spinner field

label : 'Spinner', - Gives label to spinner field (You can change label)

minValue : 0 – sets minimum value of spinner to 0

maxValue : 10 – sets minimum value of spinner to 0

stepValue : 1 – assigns step value to 1 between one spin of spinner.

cycle : true – assigns cycle to spinner.

},

* **checkboxfield**

{

xtype: 'checkboxfield' - Defines component as checkbox field

name : 'cool' - Gives name to checkbox field

label: 'Cool' - Gives label to checkbox field (You can change label)

},

* **datepickerfield -**   
   {

xtype: 'datepickerfield' - Defines component as datepicker field

destroyPickerOnHide: true -

name : 'date' - Gives name to datepicker field

label: 'Start Date' - Gives label to datepicker field (You can change label)

value: new Date() – assigns new date object

picker: {

yearFrom: 1990

} – picks yeas staring from 1990

},

* **selectfield**

{

xtype: 'selectfield' - Defines component as datepicker field

name : 'rank' Gives name to select field

label: 'Rank' - Gives label to select field (You can change label)

options: - provides options to select field

[

{

text : 'Master',

value: 'master'

},

{

text : 'Journeyman',

value: 'journeyman'

},

{

text : 'Apprentice',

value: 'apprentice'

}

]

},

{

xtype: 'textareafield' - Defines component as text area field to enter text

name : 'bio' - Gives name to text area field

label: 'Bio' - - Gives label to textarea field (You can change label)

}

* Here, we are using other fieldset to add radio buttons



**Snapshot 1.5**

{

xtype: 'fieldset', - Defines component as Fieldset

id: 'fieldset2' – Id is provide to field set to recognise it uniquely.

title: 'Favorite color' – Gives title for field set

defaults: - Assign default properties to field set and its sub items

{

xtype : 'radiofield' – It is sub xtype for radiofield inside fieldset2

labelWidth: '35%' – aggsing with to radio buttons.

},

items: [

{

name : 'color', - Gives colour as name to radio button

value: 'red' – Gives value to radio button

label: 'Red'- Gives label to radio button.

},

{

name : 'color’ - Gives colour as name to radio button,

label: 'Blue' – Gives value to radio button

value: 'blue' – Gives label to radio button

},

{

name : 'color' - Gives colour as name to radio button

label: 'Green' – Gives value to radio button

value: 'green' – Gives label to radio button

},

{

name : 'color' - Gives colour as name to radio button

label: 'Purple' – Gives value to radio button

value: 'purple' – Gives label to radio button

}

]

},

* Here, we are adding one more container to add disable button and rest bottuon within one container.



**Snapshot 1.6**

* Following code shows one disable button in added in a container to diable whole form and freeze the form unable editing it.
* Code is for when you clck disable button itwill call function disable() to disable complete form
* Functions will check whether form is disalbled or not if disabled it will show button lable as enable to again enable the form by calling enable function and vice versa.

**{**

**xtype: 'container',**

**defaults: {**

**xtype: 'button',**

**style: 'margin: .5em',**

**flex : 1**

**},**

**layout: {**

**type: 'hbox'**

**},**

**items: [**

**{**

**text: 'Disable fields',**

**scope: this,**

**hasDisabled: false,**

**handler: function(btn){**

**var fieldset1 = Ext.getCmp('fieldset1'),**

**fieldset2 = Ext.getCmp('fieldset2');**

**if (btn.hasDisabled) {**

**fieldset1.enable();**

**fieldset2.enable();**

**btn.hasDisabled = false;**

**btn.setText('Disable fields');**

**} else {**

**fieldset1.disable();**

**fieldset2.disable();**

**btn.hasDisabled = true;**

**btn.setText('Enable fields');**

**}**

**}**

**},**

* Here reset button is added to same container to reset all fields of form by calling reset function()



**Snapshot 1.7**

{

text: 'Reset',

handler: function()

{

Ext.getCmp('basicform').reset();

} }

**CHECKPOINTS**:

1. Check carefully with your indentations as if you will miss any of the braces by mistake may reflect in compilation error.
2. See that your .js file starts with Ext.Define block and check that ends at the end of the program.
3. More than one items are separated by comma (,) separator and main program block of define is ended by semi colon (;)
4. Any value is assigned using colon (:) for example ui: 'confirm' and value is written inside single coats.
5. Items block starts with square brackets [] , which contains multiple members and also sub items.
6. Don’t forget to add entries of your files in app.js.

**Source Code**

