

More Example

Example 2

The following Example helps a user to define the color priority. Below is the code snippet of a User Defined Function and, Test method to execute it .

```
//@UserDefined
function chooseColor(priority) {
  var color = "";
  if(priority == "High"){
    color = "Red";
  } else if(priority == "Medium"){
    color = "Yellow";
  } else if(priority == "Low"){
    color = "Green";
  }
  return color;
}

//@Test
function test(){
  chooseColor("Medium");}
```

Example 3

The following Example 3 explains using user defined functions and public functions and calling public functions from user defined functions, below is the code snippet of a user defined function and Test method to execute it .

```
input['type'] = {newid:"", newid: "", name:"string", first:"string",
id:"int"};
out['type'] = {newid: "", newid: "", name2:"string", first2:"string",
surname: "string", id2:"int"};input['type'] = {newid:"", newid: "",
name:"string", first:"string", id:"int"};
out['type'] = {newid: "", newid: "", name2:"string", first2:"string",
surname: "string", id2:"int"};

var myapp = {

  //@UserDefined
  doubleme: function(nu){
    return 2*nu;
  },

  //@Public
  readData: function(){
    print(input['data'].name+'---'+input['data'].first);

    out['data']= [
```

```

        {name2:"rg1", first2:"gu1", surname: "string1"},
        {name2:"rg2", first2:"gu2", surname: "string2"},
        {name2:"rg3", first2:"gu3", surname: "string3"}];
    },

    /*
    -----
    |  name2  | first2  |
    -----
    |  rg1    | gu1    |
    -----
    |  rg2    | gu2    |
    -----
    |  rg3    | gu3    |
    -----
    */
    //@Public
    writeData: function(){
        if (out['data']==null){
            out['data']=[];
        }

        index=0;
        for each (r in input['data']){
            print(r.name2+'--'+r.first2);
            currentId = 123 + index;
            currentName = r.name2 + index;
            currentFirst = r.first2 + index;
            out['data'].push({"id": currentId, "name2": currentName,
"first2":currentFirst});
            index++;
        }
    }

}

//@UserDefined
function combine(first, last){

    return first + ' ' + last;
}

//@UserDefined
function doubleme(nu){
    return 2*nu;
}

//@Test
function test(){
    input['data'] = {name:"rajeev", first:"gip"};
    myapp.readData();
    input['data'] = out['data'];// this is the straight transformation

```

```
    myapp.writeData();  
}
```