Experiment - No- 9

Aim :- Design & develop any web appl^n Using Js.

Problem statement :
Create an application for bill payment record. using Angular Js.

S/w Req :-
Eclipse IDE / Notepad, modern web browser.

Theory :-

- Angular Js is open source web appl^n framework. It was intitially created in 2009 by misko hevery and adam Abrons.
- It presently kept by google. Its most recent adoption is 1.2.21 Angular Js is quillary system. for dynamic web apps.
- It gives you a chance to utilize HTMT as layout dialect & gives you a chance to strech out. HTML linguistic structure to express your appl^n parts painly & compactly.
- Its information official & reliance infusion take out a singnificant part of code. you as of now need to compose.

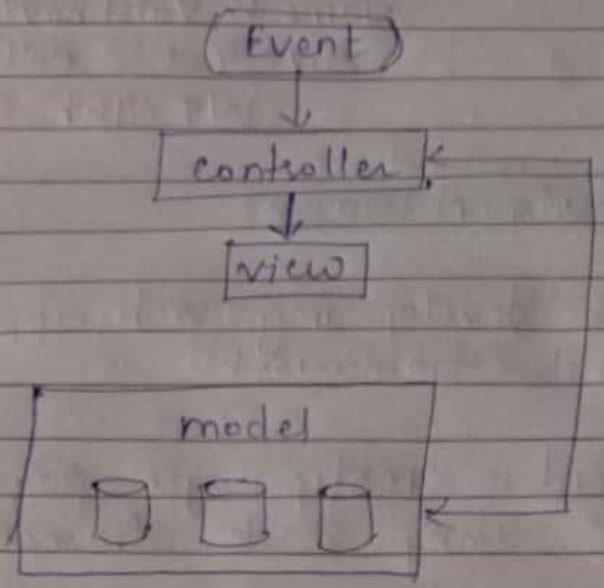


Fig. Basic MVC architecture

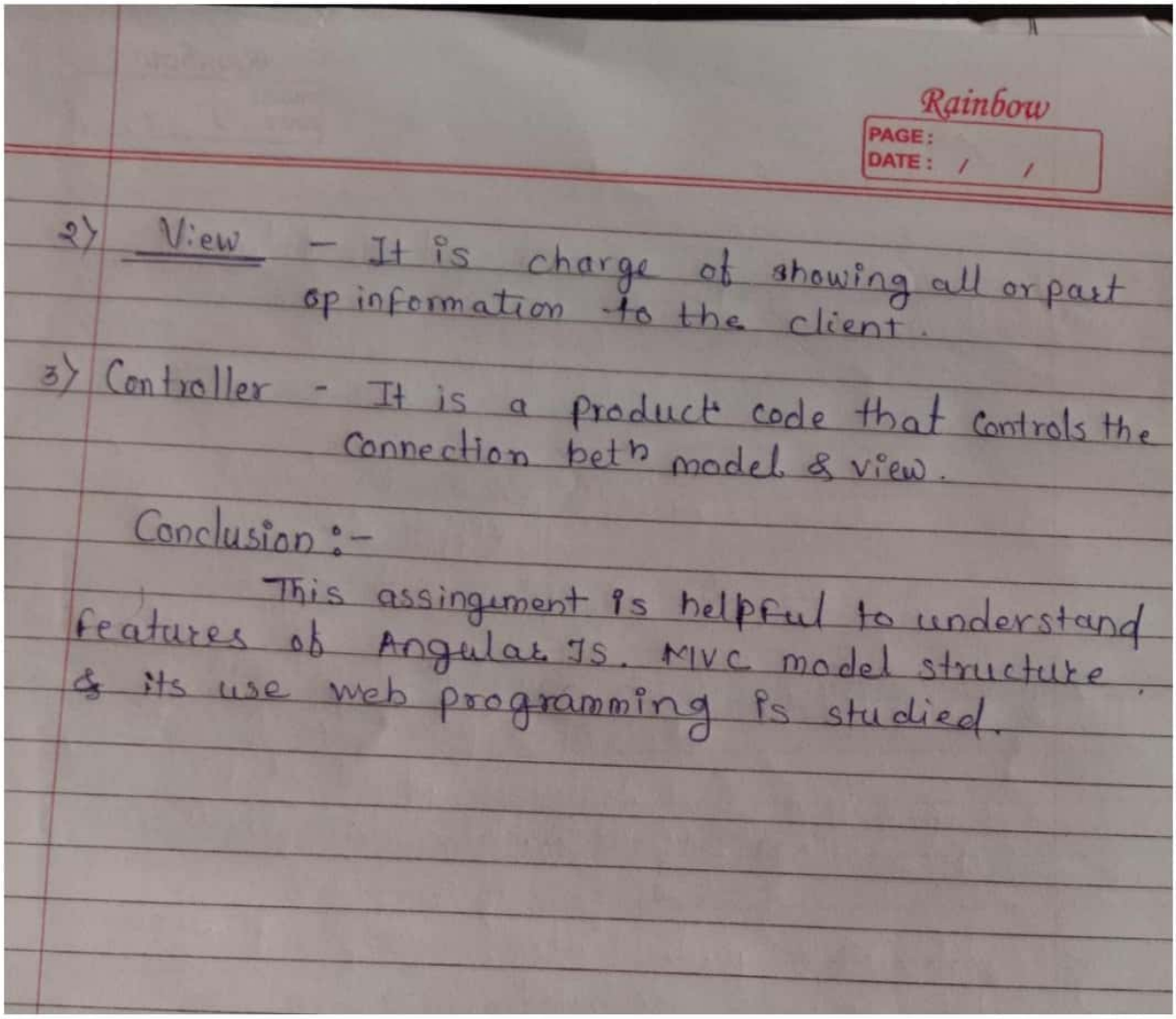
What is struts ?

Struts is a framework that advances the utilization of the model view controller engineering for planning substantial scale applications.

Steps / Design :

1] Create directory structure
2] Create I/p page
3] Provide entry of controller in file
4] Load Jar files
5] Start server & deploy project or create .war file paste it in webpages folder & run.

Conclusion : Hence, we have successfully tested struts framework & tested the results



Program:

```
<html ng-app="billpayApp">
<!-- SCRIPTS TO BE ADDED IN HEAD TAG --
> <head>
<title>Bill Payment Record using angular and bootstram
framework</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<!-- ACCESSING ANGULARJS BY CDN METHOD-->
```

```
<script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.4/angular.min
.js"></script>
<!-- ACCESSING STYLESHEET FOR DESIGN [OPTIONAL PART CAN BE SKIP]-->
<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.mi
n.css">
<!-- MODEL PART-->
<script>
var model = {
customer: "Student",
items: [{
bill: "Electricity",
status: false
},
{
bill: "Internet(Wi/fi)",
status: false
},
{
bill: "Parking Charges",
status: false
},
{
bill: "Phone",
status: true
},
{
```

```
bill: "House Tax",
status: true
}
]
}
varbillpayApp = angular.module("billpayApp", []);
billpayApp.controller("billpayctrl", function($scope)
{ $scope.billpay = model;
$scope.dueBills = function() {
var items = $scope.billpay.items;
var counter = 0;
items.forEach((item) => {
if (!item.status) {
counter++;
}
})
return counter;
} $
scope.redFlag =
function() {
return $scope.dueBills() <= 2 ? "label-success" : "label-danger";
}
$scope.addBills = function(billName)
{ obj = {
bill: billName,
status: false
} $
scope.billpay.items.push(obj);
```

```
} $
scope.removeBills =
function(rmvBills) {
$scope.billpay.items.splice($scope.billpay.items.indexOf(rmvBills), 1);
}
});
</script>
</head>
<!-- HTML BODY PART-->
<body ng-controller="billpayctrl">
<div class="container">
<div class="page-header">
<h1>{{billpay.customer}}'s Bill's remained to Be Paid -
<span class="lable" ng-class="redFlag()" ng-hide="dueBills()==0">
{{dueBills()}}
</span>
</h1>
</div>
<h3><center><b>Add extra biller fields if any</center></b></h3>
<div class="panel">
<div class="input-group">
<input class="form-control" ng-model="billName" />
<span class="input-group-btn">
<button class="btnbtn-danger" ngclick="
addBills(billName)">+ADD+</button>
</span>
</div>
<table class="table table-striped">
```

```
<thead>
<tr>
<th>Bill Name</th>
<th>Status</th>
<th>Status</th>
<th>Close</th>
</tr>
</thead>
<tbodyng-model="rmvBills">
<trng-repeat="item in billpay.items" ng-model="item">
<td>{{item.bill}}</td>
<td><input type="checkbox" ng-model="item.status" /></td>
<td>{{item.status}}</td>
<td>
<button type="button" ng-click="removeBills(item)">&times;</button>
</td>
</tr>
</tbody>
</table>
</div>
</div>
</div>
</body>
</html>
```