Spring Form Validator API Example:-

Model class:

```
package com.app.model;

import java.util.List;

public class Employee {

	private Integer empId;
	private String empName;
	private String empGen;
	private String empProj;
	private String empAddr;
	private List<String> empLangs;

	public Employee() {
		super();
	}
	public Integer getEmpId() {
		return empId;
	}
	public void setEmpId(Integer empId) {
		this.empId = empId;
	}
	public String getEmpName() {
		return empName;
	}
	public void setEmpName(String empName) {
		this.empName = empName;
	}
	public String getEmpGen() {
		return empGen;
	}
	public void setEmpGen(String empGen) {
		this.empGen = empGen;
	}
```

```
    public String getEmpProj() {
        return empProj;
    }
    public void setEmpProj(String empProj) {
        this.empProj = empProj;
    }
    public String getEmpAddr() {
        return empAddr;
    }
    public void setEmpAddr(String empAddr) {
        this.empAddr = empAddr;
    }
    public List<String> getEmpLangs() {
        return empLangs;
    }
    public void setEmpLangs(List<String> empLangs) {
        this.empLangs = empLangs;
    }
    @Override
    public String toString() {
        return "Employee [empId=" + empId + ", empName=" +
empName + ", empGen=" + empGen + ", empProj=" + empProj
                + ", empAddr=" + empAddr + ", empLangs=" +
empLangs + "]";
    }

}
```

Controller:-

```
package com.app.controller;

import
org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.ModelMap;
import org.springframework.validation.Errors;
```

```
import
org.springframework.web.bind.annotation.ModelAttribute;
import
org.springframework.web.bind.annotation.RequestMapping;
import
org.springframework.web.bind.annotation.RequestMethod;

import com.app.model.Employee;
import com.app.validator.EmployeeValidator;

@Controller
public class EmployeeController {
    @Autowired
    private EmployeeValidator validator;

    //1. show Form
    @RequestMapping("/register")
    public String showPage(ModelMap map) {
        //Form Backing Object
        map.addAttribute("employee",new Employee());
        return "Register";
    }
    //2. on click submit
    @RequestMapping(value="/save",method=RequestMethod.POST)
    public String processData(
            @ModelAttribute Employee employee,
            Errors errors,
            ModelMap map)
    {
        String page=null;
        //1. call vlidator
        validator.validate(employee,errors);
        //2. check for errors exist
        if(errors.hasErrors()) {
            page="Register";
        }else {
            page="Data";
            map.addAttribute("emp",employee);
```

```
		}
		return page;
	}

}
```

Validator:-

```
package com.app.validator;

import java.util.regex.Pattern;

import org.springframework.stereotype.Component;
import org.springframework.util.StringUtils;
import org.springframework.validation.Errors;
import org.springframework.validation.Validator;

import com.app.model.Employee;
@Component
public class EmployeeValidator
	implements  Validator {

	@Override
	public boolean supports(Class<?> clz) {
		return Employee.class.equals(clz);
	}

	@Override
	public void validate(Object target, Errors errors) {
		//Down cast to Model class
		Employee e=(Employee)target;
		//Text Input Validation
		if(!Pattern.matches("[A-Z]{2,6}",e.getEmpName())){
			errors.rejectValue("empName", null, "Please
enter valid name");
		}
		//Radio Button
```

```
        //if(e.getEmpGen()==null ||
"".equals(e.getEmpGen())) {
        if(!StringUtils.hasText(e.getEmpGen())) {
            errors.rejectValue("empGen", null, "Please
choose one Gender");
        }
        //Text Area
        if(!Pattern.matches("[A-Za-z0-9\\s]{2,50}",
e.getEmpAddr())) {
            errors.rejectValue("empAddr", null, "Please
Enter Address");
        }
        //DropDown
        if(!StringUtils.hasText(e.getEmpProj())) {
            errors.rejectValue("empProj", null, "Please
choose one Project");
        }
        //Checkbox : List
        if(e.getEmpLangs()==null ||
e.getEmpLangs().isEmpty()) {
            errors.rejectValue("empLangs", null,"Please
choose one Language");
        }
    }
}
```

Register.jsp

```
<%@taglib prefix="form"
uri="http://www.springframework.org/tags/form" %>
<html>
<head>
<style type="text/css">
.err{
    color: red;
}
</style>
</head>
<body>
<h3>Welcome to Employee Register page</h3>
```

```
<form:form action="save" method="POST"
modelAttribute="employee">
<pre>
Employee Id     : <form:input path="empId"/>
Employee Name   : <form:input path="empName"/>
<form:errors path="empName"  cssClass="err"/>
Employee Gender: <form:radiobutton path="empGen"
value="Male"/>Male <form:radiobutton path="empGen"
value="Female"/>Female
<form:errors  path="empGen" cssClass="err"/>

Employee Address: <form:textarea path="empAddr"/>
<form:errors path="empAddr" cssClass="err"/>
Employee Project:
  <form:select path="empProj">
     <form:option value="">-SELECT-</form:option>
     <form:option value="P1">P1</form:option>
     <form:option value="P2">P2</form:option>
  </form:select>
<form:errors path="empProj" cssClass="err"/>
Employee Language:
<form:checkbox path="empLangs" value="English"/> English
<form:checkbox path="empLangs" value="Hindi"/> Hindi
<form:checkbox path="empLangs" value="Telugu"/> Telugu
<form:errors path="empLangs" cssClass="err"/>
  <input type="submit" value="Register"/>
</pre>
</form:form>
</body>
</html>
```

Data.jsp

```
Data is: ${emp}
```

app.properties

```
mvc.prefix=/WEB-INF/views/
mvc.suffix=.jsp
```

AppConfig.java:-

```
package com.app.config;

import
org.springframework.beans.factory.annotation.Autowired;
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.ComponentScan;
import org.springframework.context.annotation.Configuration;
import
org.springframework.context.annotation.PropertySource;
import org.springframework.core.env.Environment;
import
org.springframework.web.servlet.config.annotation.EnableWebM
vc;
import
org.springframework.web.servlet.view.InternalResourceViewRes
olver;

@Configuration
@PropertySource("classpath:app.properties")
@ComponentScan("com.app")
@EnableWebMvc
public class AppConfig {
	@Autowired
	private Environment env;
	@Bean
	public InternalResourceViewResolver vr() {
		InternalResourceViewResolver v=new 
InternalResourceViewResolver();
		v.setPrefix(env.getProperty("mvc.prefix"));
		v.setSuffix(env.getProperty("mvc.suffix"));
		return v;
	}
}
```

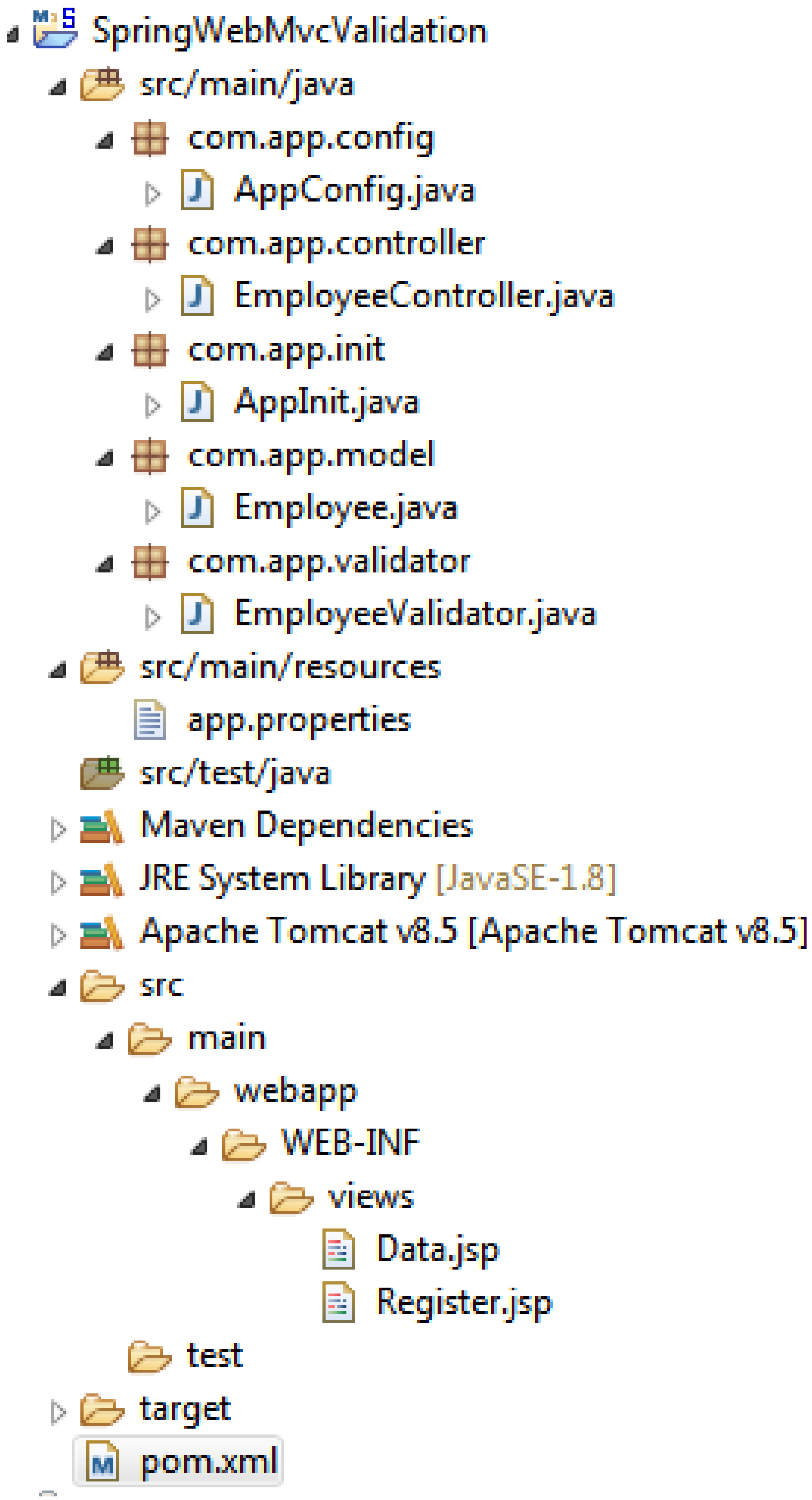
AppInit.java:-

```
package com.app.init;

import 
org.springframework.web.servlet.support.AbstractAnnotationConfigDispatcherServletInitializer;

import com.app.config.AppConfig;

public class AppInit extends 
AbstractAnnotationConfigDispatcherServletInitializer {

	@Override
	protected Class<?>[] getRootConfigClasses() {
		return new Class[] {AppConfig.class};
	}

	@Override
	protected Class<?>[] getServletConfigClasses() {
		// TODO Auto-generated method stub
		return null;
	}

	@Override
	protected String[] getServletMappings() {
		return new String[] {"/"};
	}

}
```

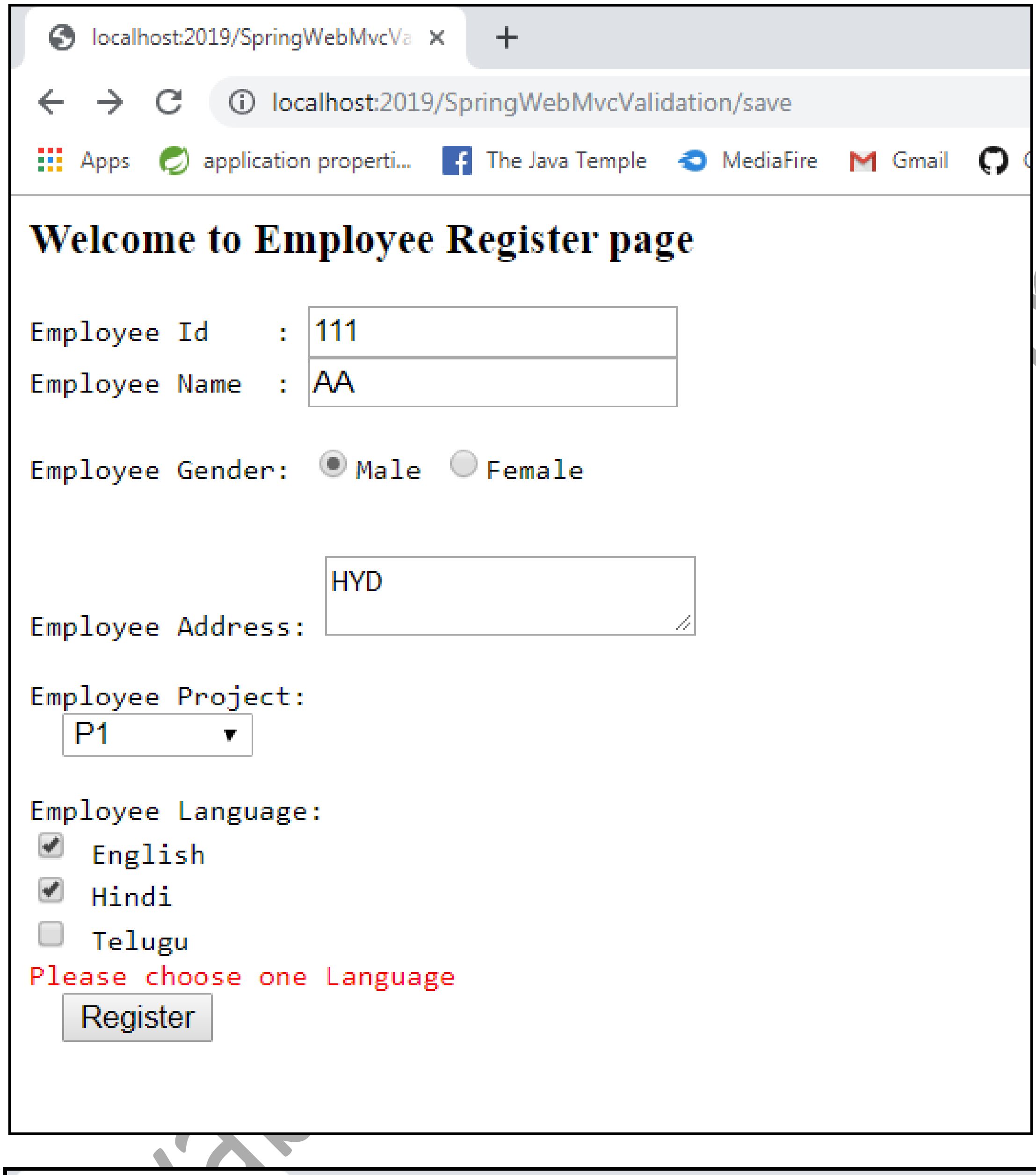
pom.xml:-

```
<dependencies>
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-webmvc</artifactId>
            <version>5.1.8.RELEASE</version>
        </dependency>
    </dependencies>

    <build>

        <plugins>
            <plugin>

    <groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-compiler-plugin</artifactId>
                <version>3.8.1</version>
                <configuration>
                    <source>1.8</source>
                    <target>1.8</target>
                </configuration>
            </plugin>

            <plugin>
                <artifactId>maven-war-plugin</artifactId>
                <version>2.4</version>
                <configuration>

    <failOnMissingWebXml>false</failOnMissingWebXml>
                </configuration>
            </plugin>
        </plugins>

    </build>
```

Folder System:



```
SpringWebMvcValidation
  src/main/java
    com.app.config
      AppConfig.java
    com.app.controller
      EmployeeController.java
    com.app.init
      AppInit.java
    com.app.model
      Employee.java
    com.app.validator
      EmployeeValidator.java
  src/main/resources
    app.properties
  src/test/java
  Maven Dependencies
  JRE System Library [JavaSE-1.8]
  Apache Tomcat v8.5 [Apache Tomcat v8.5]
  src
    main
      webapp
        WEB-INF
          views
            Data.jsp
            Register.jsp
    test
  target
  pom.xml
```


localhost:2019/SpringWebMvcVa
localhost:2019/SpringWebMvcValidation/save
Apps
application properti...
The Java Temple
MediaFire
Gmail
Register

Welcome to Employee Register page

Employee Id : 111

Employee Name : AA

Employee Gender: ● Male ○ Female

Employee Address: HYD

Employee Project: P1

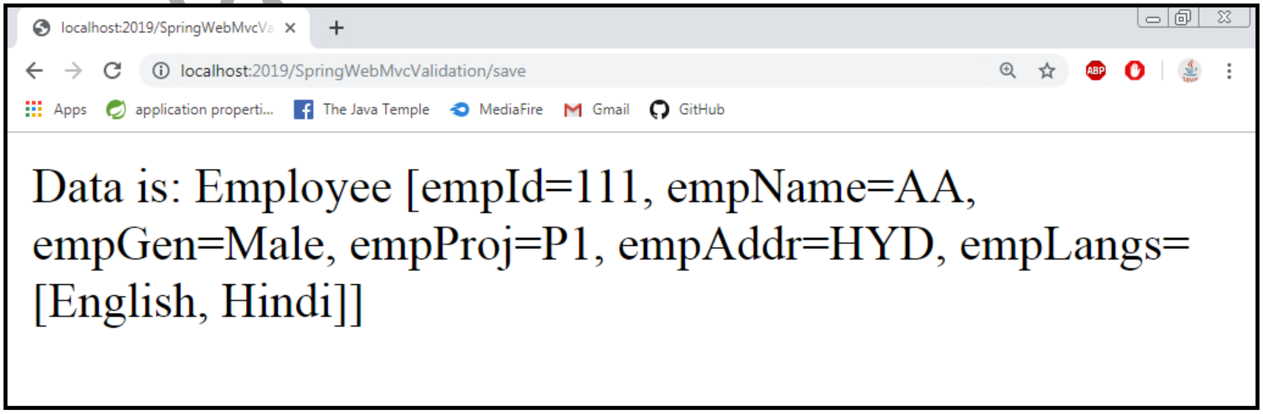
Employee Language:

☑ English

☑ Hindi

☐ Telugu

Please choose one Language


localhost:2019/SpringWebMvcVa
localhost:2019/SpringWebMvcValidation/save
Apps
application properti...
The Java Temple
MediaFire
Gmail
GitHub

```
Data is: Employee [empId=111, empName=AA,
empGen=Male, empProj=P1, empAddr=HYD, empLangs=
[English, Hindi]]
```