

Teoria:

1. Explique para que serve o framework **MAVEN**?
2. Qual a finalidade do **Apache Tomcat**?
3. Qual a função do arquivo **pom.xml**?
4. Qual a função da pasta **/WEB-INF/views** no projeto Spring MVC?
5. O que são páginas de extensão **jsp**?
6. No Spring, para que serve a annotation **@Controller**?
7. O que são **rotas** em um projeto Spring MVC?
8. Qual a finalidade da classe **MvcConfiguration**?
9. O que é o bootstrap? Qual o benefício do uso do bootstrap para desenvolvimento frontend?
10. Onde devem ficar posicionados as referências para os elementos CSS e JS dentro de uma página JSP?

Prática:

1. Crie um projeto **MAVEN PROJECT** utilizando o Archetype **spring-mvc** com o nome de **contatosweb**, defina o padrão de pacotes como **br.com.cotiinformatica**
2. Crie 3 páginas JSP dentro da pasta **/WEB-INF/views** conforme abaixo:
 - a. **login_user.jsp**
 - b. **register_user.jsp**
 - c. **password_recover.jsp**
3. Para cada uma das páginas acima, crie um controlador, conforme abaixo (dentro do pacote **controller**)
 - a. **LoginUserController.java**
 - b. **RegisterUserController.java**
 - c. **PasswordRecoverController.java**
4. Mapeie em cada controlador as seguintes rotas para as páginas:
 - a. **login_user (/)**
 - b. **register_user (register_user)**
 - c. **password_recover (password_user)**
5. Desenhe as páginas utilizando a toolkit do bootstrap com formulários para login, cadastro e recuperação de senha, colocando os seguintes campos:
 - a. **login_user**
 - i. Campos: login de acesso e senha de acesso
 - b. **register_user**
 - i. Campos: nome, telefone, login de acesso, senha de acesso e confirmação da senha
 - c. **password_recover**
 - i. email de acesso
6. Faça a navegação entre as páginas por meio de links em HTML (<a href)
7. Crie um repositório no GITHUB chamado **contatosweb**
8. Configure na pasta do projeto os arquivos .gitignore e README e faça o COMMIT para o GITHUB.