**Q1.** In Spring MVC, the \_\_\_\_\_\_\_\_\_\_\_\_\_  tag can render field validation errors.

**1)**. <sf:fieldErrors>

**2)**. **<sf:errors>**

**3)**. <sf:error>

**4)**. <sf:errorMessages>

**Q2.** Spring's \_\_\_\_\_\_\_\_\_\_\_ sends out requests to controllers.

**1)**. ActionServlet

**2)**. ControllerServlet

**3)**. **DispatcherServlet**

**4)**. FilterDispatcher

**Q3.**To figure out which controller should handle the request, DispatcherServlet queries \_\_\_\_\_\_\_\_\_\_\_ .

**1)**. **HandlerMappings**

**2)**. ModelAndView

**3)**. ViewResolver

**4)**. RequestHandler

**Q4.** How will you bring in the configuration metadata into a Spring testcase?

**1)**. **@TransactionConfiguration**

**2)**. **@ContextConfiguration**

**3)**. @RunWith

**4)**. @ContextMetadata

**Q5.** The Controller Interface's handleRequest() method returns a?

**1)**. **ModelAndView object**

**2)**. JSP page

**3)**. View object

**4)**. Form object

**Q6.**You want to write a JPA Entity class to model a databank table named COOPERATE\_USER. Each user is  uniquely identified in this table by his or her social security number SSN. Along with SSN the system keeps username, job, address and birth date.  
  
How to declare such JPA Entity class?  
  
Please choose all the answers that apply

**1)**. **Write a CooperateUser public class**

**2)**. **Annotate the CooperateUser class with @Table(name=” COOPERATE\_USER”)**

**3)**. **Annotate the CooperateUser class with @Entity**

**4)**. **Define private attributes ssn, job, name, addess and birthdate**

**5)**. **Annotate the ' name' attibute with @Id**

**6)**. **The CooperateUser class must implement Serializable**

**Q7.** Look at a snippet from web.xml of a Spring MVC application:  
  
<web-app>  
     <servlet>  
          <servlet-name>tradingapp</servlet-name>  
          <servlet-class> org.springframework.web.servlet.DispatcherServlet </servlet-class>  
     </servlet>  
     <servlet-mapping>  
          <servlet-name>tradingapp</servlet-name>  
          <url-pattern>\*.htm</url-pattern>  
     </servlet-mapping>  
  
What should be the name of the Spring configuration file for this application?

**1)**. tradingapp.xml

**2)**. spring-config.xml

**3)**. **tradingapp-servlet.xml**

**4)**. Any name is ok

**Q8.** Given the code snippet below, which of the following statements are true?  
          @PersistenceContext  
    private EntityManager entityManager;  
     public void registerUser() {  
     User usr = new User(); // Line 1  
     usr.setName(“John”); // Line 2  
     entityManager.persist(usr); // Line 3  
   }             
Please choose all the answers that apply:

**1)**. At line 1 the user instance is a managed Entity

**2)**. **At line 1 the user instance is a new Entity**

**3)**. **At line 1 a new record will be inserted in the user table**

**4)**. At line 2 the entityManager will immediately reflect user name change to

**5)**. **At line 3 the entityManager will immediately reflect user name change to**

**Q9.** \_\_\_\_\_\_\_\_\_\_ manages a TestContext.

**1)**. **ContextManager**

**2)**. **TestContextManager**

**3)**. TestExecutionListener

**4)**. TestContextClass

**Q10.** Consider the code snippet below for addUser.jsp in a MVC web-application:  
  
<%@ taglib prefix="sf" uri="http://www.springframework.org/tags/form"%>  
<h2>Create a User</h2>  
<sf:form method="GET" modelAttribute="user" action="ProcessUser.obj" >  
….  
<sf:input  \_\_\_\_\_\_\_\_\_="username" size="15" maxlength="15" />  
.....  
  
Most form elements from spring-form.tld have an attribute that is used to connect form   
input field with the corresponding field in the command object? Which is this element   
that must replace the blank?

**1)**. commandPath

**2)**. command

**3)**. **path**

**4)**. bind