**Q1.** The DispatcherServlet is an implementation of what design pattern?

**1)**. Controller

**2)**. Template

**3)**. **Front Controller**

**4)**. Interceptor

**Solution** :  
option [3] is correct

**Attempted** :  
option [3] is attempted

**Q2.** Which statements about DispatcherServlet are true?

**1)**. It's a Model part of the MVC pattern implemented in Spring MVC.

**2)**. It's a View part of the MVC pattern implemented in Spring MVC.

**3)**. **It's a Controller part of the MVC pattern implemented in Spring MVC.**

**4)**. It's an example of the Intercepting Filter pattern implementation

**Q3.** JpaRepository is a child inteface of following?

**1)**. CrudRepository<T,ID>,

**2)**. PagingAndSortingRepository<T,ID>,

**3)**. QueryByExampleExecutor<T>,

**4)**. **Repository<T,ID>**

**5)**. all of the above.

**Q4.**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)**. **The CooperateUser class must implement Serializable**

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

**Q5.** Which of the following correctly describe the modules of the Spring architecture?

**1)**. The Context module provides a Web-based framework for integrating Spring’s IoC and DI containers through the J2EE Servlet API

**2)**. **The Web module provide transaction management for data access business logic, utilizing AOP to inject transaction logic into domain functionality**

**3)**. **The inversion of Control(IOC) and dependency Injection (DI) contains , as well as fundamental parts of the Spring framework, are provided by the core and Beans modules.**

**4)**. **The transaction modules provides and AOP framework to utilize proxies to transparently inject any desired business logic into domain functionality.**

**5)**. . The Aspect-Oriented Programming(AOP) modules builds on the Core and Business modules to provide the ApplicationContext infrastructure, allowing access to object in a framework style

**Q6.** @Controller is a specialization of the @Component annotation.

**1)**. **TRUE**

**2)**. FALSE

**Q7.**The request has already reached the DispatcherServlet. Given the snippet from the xml file,   
which component will handle the request next?   
  
<beans>   
    <!-- basicspringController implements Controller -->  
    <bean id="basicspringController" class="training.web.spring.BasicSpringController" >  
           <!-- some properties -->  
    </bean>  
    <bean id="beanMapping" class="org.springframework.web.servlet.handler.BeanUrlHandlerMapping">  
          <!-- some properties -->  
    </bean>  
    <bean id="viewResolver" class="org.springframework.web.servlet.view.InternalResourceViewResolver">  
          <!-- some properties -->  
    </bean>  
    <bean id="messageSource" class="org.springframework.context.support.ResourceBundleMessageSource">  
           <!-- some properties -->  
     </bean>  
</beans>

**1)**. basicspringController

**2)**. **beanMapping**

**3)**. **viewResolver**

**4)**. messageSource

**Q8.** Where do you expect to find the servlet specific Spring config to the "example"   
servlet in the web.xml below?  
  
<servlet>  
    <servlet-name>example</servlet-name>  
    <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>  
    <load-on-startup>1</load-on-startup>  
</servlet>

**1)**. **/WEB-INF/example-servlet.xml**

**2)**. **/WEB-INF/servlet-config.xml**

**3)**. /WEB-INF/default-config.xml

**4)**. /WEB-INF/example-servlet-config.xml

**Q9.** When Spring (MVC) is used in a web-aware way 3 additional scopes become available to bind beans to, these are?

**1)**. **request**

**2)**. **singleton**

**3)**. **session**

**4)**. **prototype**

**5)**. **global-session**

**Q10.** Given the code snippet below:  
@Controller  
public class MyClass {  
    public ModelAndView \_\_\_\_\_\_\_\_\_\_\_\_\_(HttpServletRequest request, HttpServletResponse response)   
    throws ServletException, IOException  {  
           String now = new java.util.Date().toString();  
            return new ModelAndView("hello","now",now);  
       }    
}  
  
What should be the method name?

**1)**. **controlRequest**

**2)**. handle

**3)**. only handleRequest

**4)**. **any user defined method**