

TESTPASSPORT

練習問題

● ● ● ● ●



高品質な学習リソース

<https://www.testpassport.jp>

1年間無料アップデート

Exam : CoreSpringV3.2

**Title : Core-Spring (based on
Spring 3.2)**

Version : DEMO

1. Select which statement(s) is/are true with respect to programming to interfaces with Spring
- A. The use of interfaces allows for reduced coupling between collaborating objects
 - B. Spring requires all beans to implement interfaces
 - C. Spring requires that parameters in constructors and setters are defined using interface types
 - D. Spring requires all beans to have an empty constructor (either default or declared)

Answer: A

2. How can a bean of type LegacySingleton be created (using XML configuration)? (select one)

Consider the following class:

```
public class LegacySingleton {  
    private LegacySingleton(){}  
    public static LegacySingleton getAServiceInstance() {  
        return new LegacySingleton();  
    }  
}
```

- A. It is not possible, the constructor must be public
- B. Use the factory-method attribute on the <bean> tag
- C. Use the init-method attribute on the <bean> tag
- D. Use autowiring

Answer: B

3. Which of the following scenarios requires you to instantiate an ApplicationContext using the 'new' keyword? (Select one)

- A. Running your Spring application inside a JUnit test (using SpringJUnit4ClassRunner)
- B. Bootstrapping your Spring application within a Java main() method
- C. Deploying your Spring application in an application server, packaged in a WAR file
- D. Both a and b

Answer: B

4. Select which statement is true with respect to constructor injection with Spring (select one)

- A. Multiple parameters can be dependency injected into a constructor
- B. Using XML configuration, the constructor-arg element may be omitted if the constructor requires a single parameter
- C. One single bean cannot mix constructor injection with setter injection
- D. All of the above

Answer: A

5. Which of the following statements is true? (Select one)

Consider the following complete configuration sample:

```
<bean class="rewards.internal.RewardNetworkImpl">  
    <property name="accountRepository" ref="accountRepository"/>  
</bean>  
<bean class="rewards.internal.account.JdbcAccountRepository"/>
```

- A. This configuration is correct

- B. This configuration is not valid because the first bean should have an id. Its value should be "rewardNetwork".
- C. This configuration is not valid because the second bean should have an id. Its value should be "accountRepository".
- D. Both (b) and (c)

Answer: C