

# W1D3: Quiz

## CS544 Enterprise Architecture

What is the output of the code shown below, and on the back:

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns=http://www.springframework.org/schema/beans ...>

    <bean id="classA" class="wld1.ClassA" init-method="init"
        destroy-method="destroy" scope="prototype">
        <constructor-arg value="Hello" />
        <property name="classB" ref="classB" />
    </bean>

    <bean id="classB" class="wld1.ClassB" init-method="init"
        destroy-method="destroy" lazy-init="true">
        <property name="text" value="World" />
    </bean>

    <bean id="classC" class="wld1.ClassC" />
</beans>
```

```
public class Application {
    public static void main(String[] args) {
        ConfigurableApplicationContext context = new
            ClassPathXmlApplicationContext("wld1/springconfig.xml");
        System.out.println("Testing Spring Startup");
        ClassA classA = context.getBean("classA", ClassA.class);
        classA.showText();
        context.close();
    }
}
```

```
public class ClassA {
    private String text;
    private ClassB classB;

    public ClassA(String text) {
        this.text = text;
        System.out.println("Class A constructor - text: " + text);
    }

    public void setClassB(ClassB classB) { this.classB = classB; }

    public void showText() {
        System.out.println("Class A text: " + text);
        classB.showText();
    }

    public void init() { System.out.println("Class A init - text: " + text); }

    public void destroy() { System.out.println("Class A destroy"); }
}
```

Name: \_\_\_\_\_ StudentID: \_\_\_\_\_

```
public class ClassB {  
    private String text;  
  
    public ClassB() { System.out.println("Class B constructor - text: " + text); }  
  
    public void setText(String text) { this.text = text; }  
  
    → public void showText() { System.out.println("Class B text: " + text); }  
  
    public void init() { System.out.println("Class B init - text: " + text); }  
  
    public void destroy() { System.out.println("Class B destroy"); }  
}
```

```
public class ClassC {  
    public ClassC() { System.out.println("Class C constructor"); }  
}
```

Output:

class C constructor  
Testing Spring Startup  
Class A constructor - text: Hello  
Class B constructor - text:  
class B init - text: World  
Class A init - text: Hello  
Class A text: Hello  
class B text: World  
Class B Destroy

A [ Constructor  
setter  
init ] B  
const.  
setter  
init