

```
/*
Pro Spring
By Rob Harrop
Jan Machacek
ISBN: 1-59059-461-4
Publisher: Apress
*/
```

```
////////////////////////////////////
////
//File: factory.xml
<!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN//EN" "http://www.springframework.org/dt
d/spring-beans.dtd">
<beans>
  <bean id="shaDigest" class="MessageDigestFactoryBean">
    <property name="algorithmName">
      <value>SHA1</value>
    </property>
  </bean>
  <bean id="defaultDigest" class="MessageDigestFactoryBean"/>
  <bean id="digester" class="MessageDigester">
    <property name="digest1">
      <ref local="shaDigest"/>
    </property>
    <property name="digest2">
      <ref local="defaultDigest"/>
    </property>
  </bean>
</beans>
```

```
////////////////////////////////////
////
import java.security.MessageDigest;

import sun.misc.BASE64Encoder;

public class MessageDigester {

  private MessageDigest digest1 = null;
  private MessageDigest digest2 = null;

  public void setDigest1(MessageDigest digest1) {
    this.digest1 = digest1;
  }

  public void setDigest2(MessageDigest digest2) {
    this.digest2 = digest2;
  }

  public void digest(String msg) {
    System.out.println("Using digest1");
    digest(msg, digest1);

    System.out.println("Using digest2");
```

```

        digest(msg, digest2);
    }

    private void digest(String msg, MessageDigest digest) {
        System.out.println("Using algorithm: " + digest.getAlgorithm());
        digest.reset();
        byte[] bytes = msg.getBytes();
        byte[] out = digest.digest(bytes);
        BASE64Encoder enc = new BASE64Encoder();
        System.out.println(enc.encode(out));
    }
}

////////////////////////////////////
////

import java.security.MessageDigest;

import org.springframework.beans.factory.FactoryBean;
import org.springframework.beans.factory.InitializingBean;

public class MessageDigestFactoryBean implements FactoryBean, InitializingBean {

    private String algorithmName = "MD5";

    private MessageDigest messageDigest = null;

    public Object getObject() throws Exception {
        return messageDigest;
    }

    public Class getObjectType() {
        return MessageDigest.class;
    }

    public boolean isSingleton() {
        return true;
    }

    public void afterPropertiesSet() throws Exception {
        messageDigest = MessageDigest.getInstance(algorithmName);
    }

    public void setAlgorithmName(String algorithmName) {
        this.algorithmName = algorithmName;
    }
}

////////////////////////////////////
////

import org.springframework.beans.factory.BeanFactory;

```

```
import org.springframework.beans.factory.xml.XmlBeanFactory;
import org.springframework.core.io.FileSystemResource;

public class MessageDigestExample {

    public static void main(String[] args) {
        BeanFactory factory = new XmlBeanFactory(new FileSystemResource(
            "build/factory.xml"));

        MessageDigerster digester = (MessageDigerster) factory
            .getBean("digerster");
        digester.digest("Hello World!");
    }
}
```