**Javaee Assignment2测试设计与测试报告**

1. 测试目标及测试范围

本次测试主要是针对Javaee第二次作业的各项功能进行单元测试，主要是测试代码中service层额外添加的功能。

1. 测试案例与测试数据

我们一共测试了20轮单元测试，下面是其中一些测试用例和预期结果

OrderService:

delOrderByOrderId()

输入数据：OrderId

预期结果：数据库删除OrderId对应的数据

getOrderByOrderId()

输入数据：OrderId

预期结果：返回OrderId对应的Order数据

ProductService

addProduct()

输入数据：product实例

预期结果： 数据库成功添加对应数据

delProductById()

输入数据：productId

预期结果：删除productId对应Product

SiteService:

addSite()

输入数据：site实例

预期结果：数据库成功添加对应数据

delSiteById()

输入数据：siteId

预期结果：删除siteId对应site

1. 测试代码

OrderServiceTest.java:

import org.junit.Test;

import com.hny.bean.Order;

import com.hny.service.OrderService;

import org.junit.Test;

import org.junit.runner.RunWith;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.test.context.ContextConfiguration;

import org.springframework.test.context.junit4.SpringJUnit4ClassRunner;

import java.util.Arrays;

import java.util.List;

import static org.junit.Assert.\*;

@RunWith(SpringJUnit4ClassRunner.class)

@ContextConfiguration(locations = {"classpath\*:/\*.xml"})

public class OrderServiceTest {

@Autowired

public OrderService orderService;

@Test

public void delOrderByOrderId() {

int i=orderService.delOrderByOrderId(1);

System.out.println(i);

}

@Test

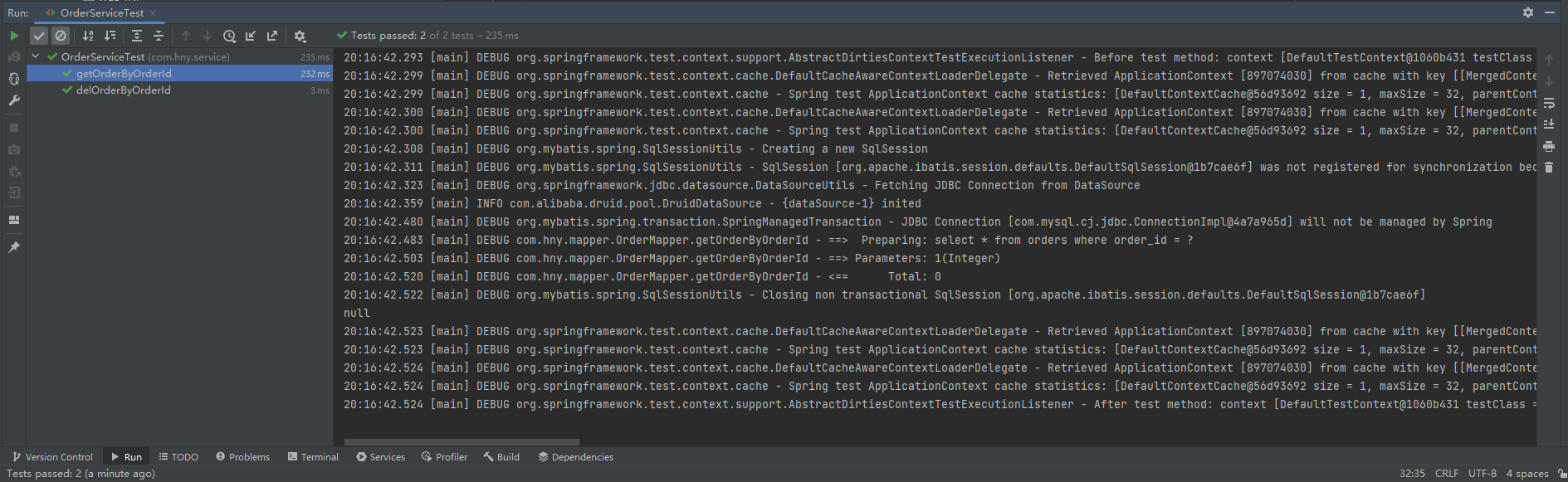
public void getOrderByOrderId() {

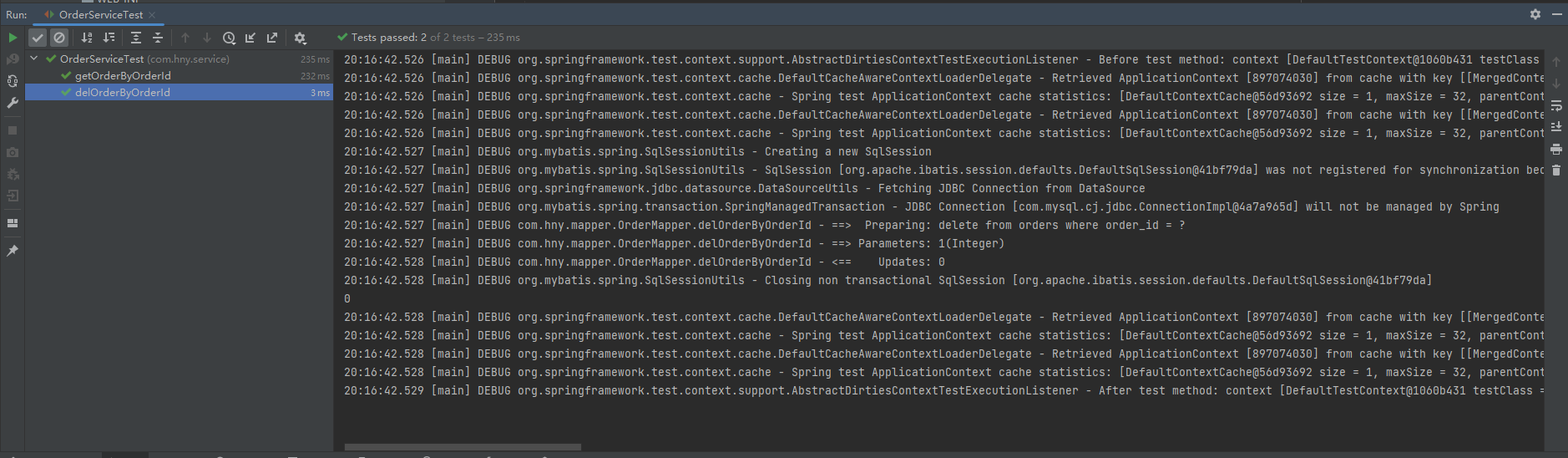
Order order=orderService.getOrderByOrderId(1);

System.out.println(order);

}

}





ProductServiceTest.java:

import org.junit.Test;

import static org.junit.Assert.\*;

import org.junit.Test;

import static org.junit.Assert.\*;

import org.junit.Test;

import com.hny.service.ProductService;

import com.hny.bean.Product;

import org.junit.Test;

import org.junit.runner.RunWith;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.test.context.ContextConfiguration;

import org.springframework.test.context.junit4.SpringJUnit4ClassRunner;

import java.util.Arrays;

import java.util.List;

@RunWith(SpringJUnit4ClassRunner.class)

@ContextConfiguration(locations = {"classpath\*:/\*.xml"})

public class ProductServiceTest {

@Autowired

public ProductService productService;

@Test

public void addProduct() {

Product product=new Product();

product.setProductId(1);

int i= productService.addProduct(product);

System.out.println(i);

}

@Test

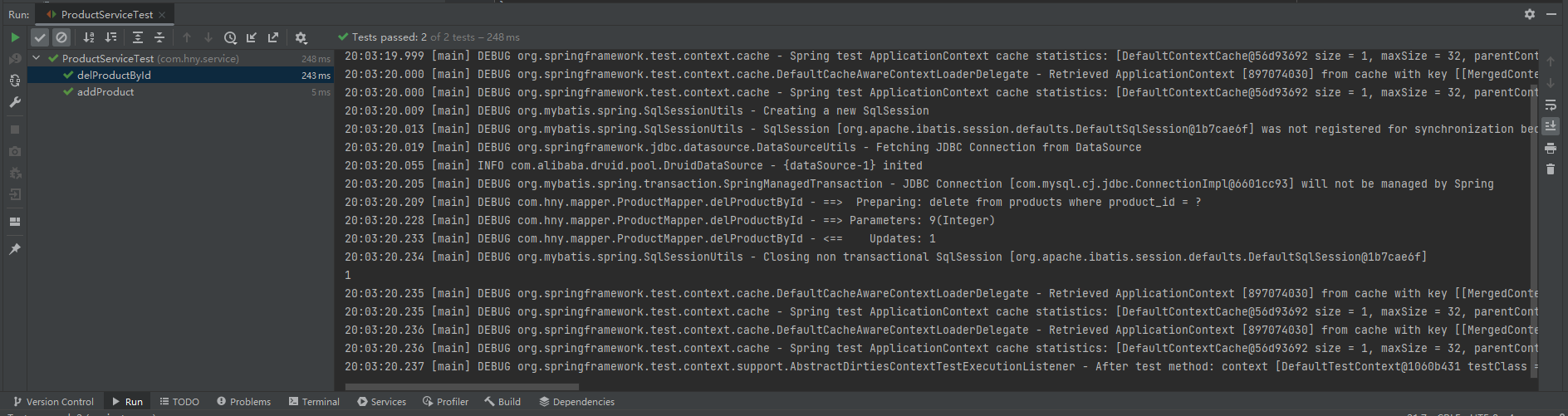
public void delProductById() {

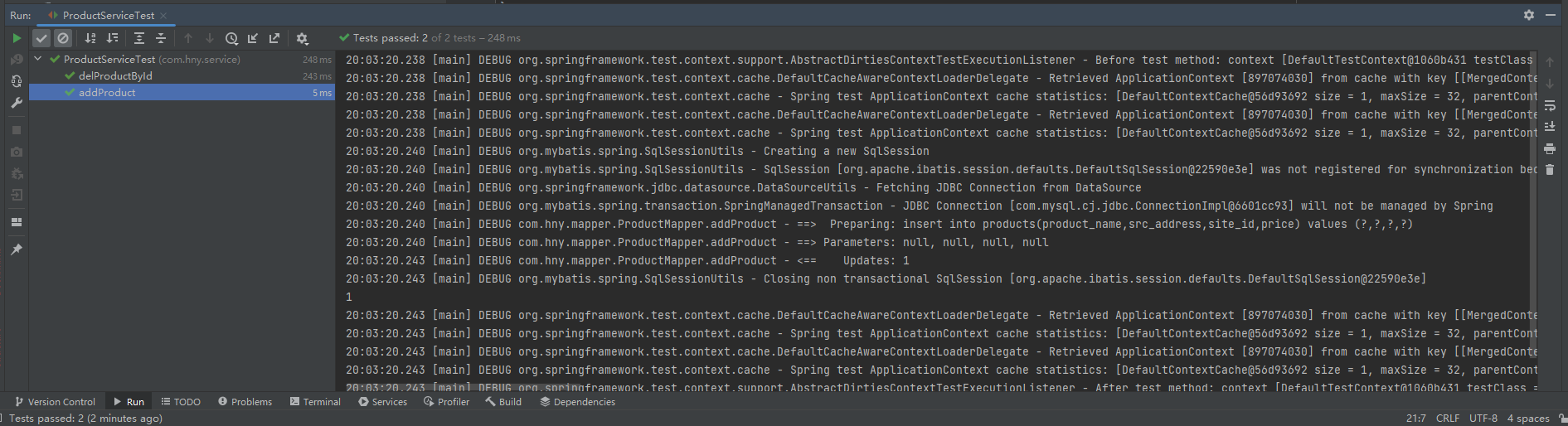
int i= productService.delProductById(9);

System.out.println(i);

}

}





SiteServiceTest.java:

import org.junit.Test;

import static org.junit.Assert.\*;

import com.hny.service.SiteService;

import com.hny.bean.Site;

import org.junit.Test;

import org.junit.runner.RunWith;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.test.context.ContextConfiguration;

import org.springframework.test.context.junit4.SpringJUnit4ClassRunner;

import java.util.Arrays;

import java.util.List;

@RunWith(SpringJUnit4ClassRunner.class)

@ContextConfiguration(locations = {"classpath\*:/\*.xml"})

public class SiteServiceTest {

@Autowired

public SiteService siteService;

@Test

public void addSite() {

Site site=new Site();

site.setSiteId(1);

site.setSiteAddress("aabb");

siteService.addSite(site);

}

@Test

public void delSiteById() {

int i=siteService.delSiteById(1);

System.out.println(i);

}

}

