

**Министерство образования и науки Российской Федерации**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО**  
**ОБРАЗОВАНИЯ**  
**“НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**  
**УНИВЕРСИТЕТ ИТМО”**

**Факультет программной инженерии и компьютерной техники**

**ЛАБОРАТОРНАЯ РАБОТА № 7**  
**ПО ДИСЦИПЛИНЕ «ТЕХНОЛОГИИ ВЕБ-СЕРВИСОВ»**

Студент: Калинин Даниил Леонидович

Группа: Р41141

Преподаватель: Дергачев Андрей Михайлович

\

Санкт-Петербург

2021

## Задание:

Требуется разработать приложение, осуществляющее регистрацию сервиса в реестре jUDDI, а также поиск сервиса в реестре и обращение к нему. Рекомендуется реализовать консольное приложение, которое обрабатывает 2 команды. Итог работы первой команды – регистрация сервиса в реестре; вторая команда должна осуществлять поиск сервиса, а также обращение к нему.

### Этапы выполнения:

1. На первом шаге был развернут реестр jUDDI v. 3.2.0. Затем была реализована регистрация разработанного ранее web-сервиса в реестре jUDDI. Ниже приведен код классов регистрации сервиса

```
public class Client {  
  
    private final static Logger log =  
Logger.getLogger(Client.class.getName());  
  
    public static void main(String[] args) {  
        try {  
            UddiService uddiService =  
                new UddiService("uddi.xml");  
            String URL = "http://localhost:8081/PersonService?wsdl";  
            String serviceKey = uddiService.publish("Persons",  
"PersonService", URL);  
            System.in.read();  
            uddiService.removeBusiness(serviceKey);  
            uddiService.logout();  
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
    }  
}
```

```
public class UddiService {  
    private UDDIClerk clerk;  
  
    private UDDISecurityPortType security;  
    private UDDIInquiryPortType inquiry;  
    private BusinessService businessService;  
  
    public UddiService(String pathToUddiSettings) {  
        try {  
            UDDIClient uddiClient = new UDDIClient(pathToUddiSettings);  
            this.clerk = uddiClient.getClerk("default");  
            security =  
uddiClient.getTransport("default").getUDDISecurityService();  
            inquiry =  
uddiClient.getTransport("default").getUDDIInquiryService();  
            if (clerk == null)  
                throw new Exception("the clerk wasn't found, check the config
```

```

file!");
    } catch (Exception e) {
        e.printStackTrace();
    }
}

private Name getName(String name,
                     String lang) {
    Name busName = new Name();
    busName.setValue(name);
    busName.setLang(lang);
    return busName;
}

private String ListToString(List<Name> name) {
    StringBuilder sb = new StringBuilder();
    for (int i = 0; i < name.size(); i++) {
        sb.append(name.get(i).getValue()).append(" ");
    }
    return sb.toString();
}

private Description getDescription(String descr) {
    Description description = new Description();
    description.setValue(descr);
    description.setLang("ru-ru");
    return description;
}

private String ListToDescString(List<Description> name) {
    StringBuilder sb = new StringBuilder();
    for (Description description : name) {
        sb.append(description.getValue()).append(" ");
    }
    return sb.toString();
}

private BusinessEntity registerBusiness(String businessName) {
    BusinessEntity businessEntity = new BusinessEntity();

    businessEntity.getName().add(getName(businessName, "ru-ru"));
    businessEntity.getDescription().add(getDescription(businessName));
    BusinessEntity register = clerk.register(businessEntity);
    if (register == null) {
        System.out.println("Ошибка регистрации...");
        return null;
    }
    String businessKey = register.getBusinessKey();
    System.out.println("Business key: " + businessKey);
    return register;
}

private BusinessService registerService(String businessKey,
                                       String servName,
                                       String URL) {
    BindingTemplate bindingTemplate = new BindingTemplate();
    AccessPoint accessPoint = new AccessPoint();
    accessPoint.setUseType(AccessPointType.WSDL_DEPLOYMENT.toString());
    accessPoint.setValue(URL);
    bindingTemplate.setAccessPoint(accessPoint);

    BindingTemplates bindingTemplates = new BindingTemplates();
    bindingTemplates.getBindingTemplate().add(bindingTemplate);
}

```

```

        BusinessService businessService = new BusinessService();
        businessService.getName().add(getName(servName, "ru-ru"));
        businessService.getDescription().add(getDescription(servName));
        businessService.setBusinessKey(businessKey);
        businessService.setBindingTemplates(bindingTemplates);
        BusinessService svc = clerk.register(businessService);

        return svc;
    }

    public String publish(String businessName,
                        String serviceName,
                        String URL) {
        try {
            BusinessEntity businessEntity =
this.registerBusiness(businessName);
            if (businessEntity == null) {
                System.out.println("Ошибка регистрации бизнеса...");
                exit(1);
            }

            BusinessService svc =
registerService(businessEntity.getBusinessKey(),
                serviceName,
                URL);
            if (svc == null) {
                System.out.println("Непредвиденная ошибка регистрации веб-
сервиса.");
                exit(1);
            }

            businessService = svc;
            String serviceKey = svc.getServiceKey();
            System.out.println(String.format("ServiceKey %s для сервиса %s",
serviceKey, serviceName));

            System.out.println("Регистрация сервиса произведена успешно.");
            return serviceKey;
        } catch (Exception e) {
            e.printStackTrace();
        }
        return null;
    }

    public void printServiceWSDLINFOToConsole(String WSDL_url) throws
Exception {
        URL url = new URL(WSDL_url);
        URLConnection conn = url.openConnection();

        DocumentBuilderFactory factory =
DocumentBuilderFactory.newInstance();
        DocumentBuilder builder = factory.newDocumentBuilder();

        Document document = builder.parse(conn.getInputStream());

        TransformerFactory factory1 = TransformerFactory.newInstance();
        Transformer xform = factory1.newTransformer();

        File file = new File("write.xml");
        if (!file.exists()) {

```

```

        file.createNewFile();
    }

    StreamResult streamResult = new StreamResult();
    streamResult.setSystemId(file);

    xform.transform(new DOMSource(document), streamResult);
    xform.transform(new DOMSource(document), new
StreamResult(System.out));
    }

    public void printServiceDetail(BusinessService businessService) throws
Exception {
        if (businessService == null) {
            return;
        }
        System.out.println("Name: " +
ListToString(businessService.getName()));
        System.out.println("Desc: " +
ListToDescString(businessService.getDescription()));
        System.out.println("Key: " + (businessService.getServiceKey()));

        JAXB.marshal(businessService, System.out);

printServiceWSDLINFOToConsole(businessService.getBindingTemplates().getBindin
gTemplate().get(0).getAccessPoint().getValue());

    }

    public void removeBusiness(String businessKey) {
        clerk.unregisterBusiness(businessKey);
    }

    public void logOut() {
        clerk.discardAuthToken();
    }

    public BusinessService getBusinessService() {
        return businessService;
    }
}

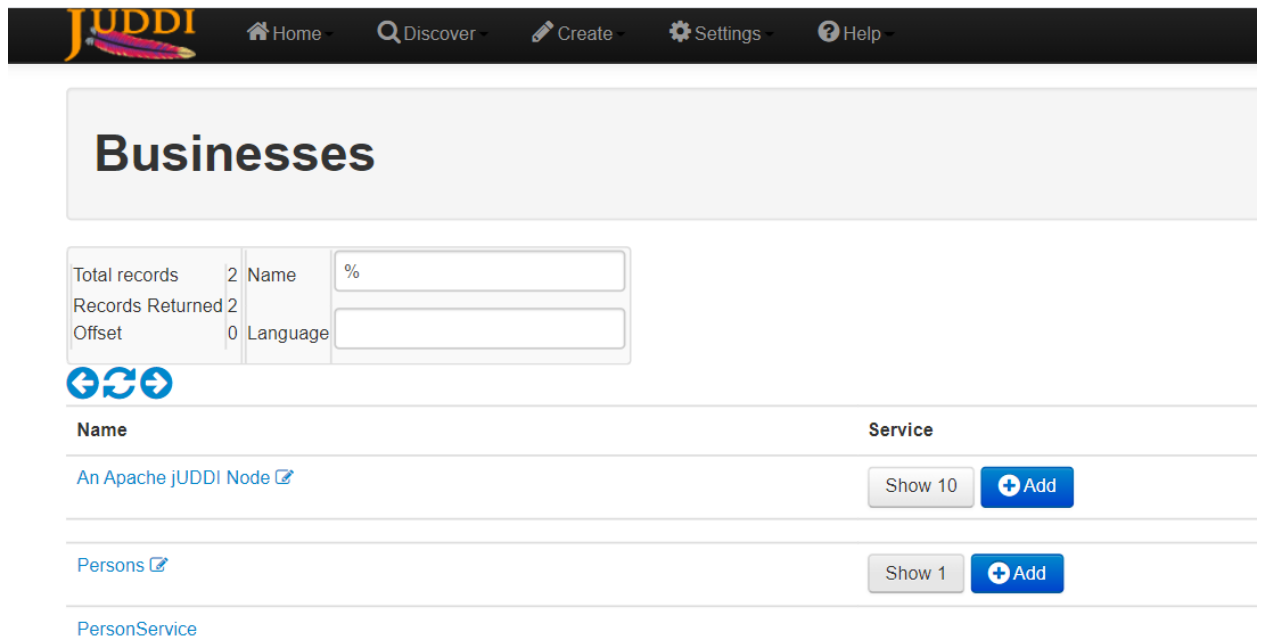
```

## Результат работы программы

```
"C:\Program Files\Java\jdk1.8.0_281\bin\java.exe" ...
июн 24, 2021 6:42:46 PM org.apache.juddi.v3.client.config.UDDIClient <init>
INFO: jUDDI Client version - 3.3.5
июн 24, 2021 6:42:47 PM org.apache.juddi.v3.client.config.ClientConfig loadConfiguration
INFO: Reading UDDI Client properties file file:///C:/Users/User/IdeaProjects/MTP4/src/Lib/Lab1/Lab_3/SOAP_client/target/classes/uddi.xml use -Du
июн 24, 2021 6:42:48 PM org.apache.juddi.v3.client.transport.JAXWSTransport getUDDISecurityService
WARNING: You should consider using a secure protocol (https) when sending your password!
июн 24, 2021 6:42:48 PM org.apache.juddi.v3.client.config.UDDIClerk register
INFO: Registering business Persons with key null
июн 24, 2021 6:42:48 PM org.apache.juddi.v3.client.config.UDDIClerk getAuthToken
WARNING: Hey, I couldn't help but notice that your credentials aren't encrypted. Please consider doing so
июн 24, 2021 6:42:49 PM org.apache.juddi.v3.client.transport.JAXWSTransport getUDDISecurityService
WARNING: You should consider using a secure protocol (https) when sending your password!
Business key: uddi:juddi.apache.org:897aaead-2e25-4ef7-acce-94157f95b621
июн 24, 2021 6:42:50 PM org.apache.juddi.v3.client.config.UDDIClerk register
INFO: Registering service PersonService with key null
ServiceKey uddi:juddi.apache.org:c1124c2f-8aad-4c72-a74d-22f0229b3acf для сервиса PersonService
Регистрация сервиса произведена успешно.
июн 24, 2021 6:42:51 PM org.apache.juddi.v3.client.transport.JAXWSTransport getUDDISecurityService
WARNING: You should consider using a secure protocol (https) when sending your password!

Process finished with exit code 0
```

## Сервис успешно зарегистрирован в реестре Juddi



**JUDDI** Home Discover Create Settings Help

### Businesses

Total records: 2 Name: %  
Records Returned: 2  
Offset: 0 Language:

← ↺ →

Name	Service
<a href="#">An Apache jUDDI Node</a>	Show 10 <a href="#">+ Add</a>
<a href="#">Persons</a>	Show 1 <a href="#">+ Add</a>
<a href="#">PersonService</a>	

## Вывод:

В ходе выполнения лабораторной работы было произведено развертывание реестра сервисов jUDDI, после чего в развернутый реестр был добавлен PersonWebService, созданный в ходе выполнения первых трех лабораторных работ