

## Приложение А Листинги программы

### FilterRequest.java

```
package com.omendba.shop;
import java.io.*;
import java.util.*;
public class FilterRequest {
public static boolean filterString(String s) {
    if (s == null)
        return false;
    if (s.length() == 0)
        return false;
    java.util.HashMap hm = new java.util.HashMap();
    hm.put(Character.valueOf('"'), "&quot;");
    hm.put(Character.valueOf('\''), "&#0039;");
    hm.put(Character.valueOf('`'), "&#0096;");
    hm.put(Character.valueOf('<'), "&lt;");
    hm.put(Character.valueOf('>'), "&gt;");
    for (int i = 0; i < s.length(); i++) {
        if (hm.containsKey(Character.valueOf(s.charAt(i)))
            || hm.containsValue(Character.valueOf(s.charAt(i))))
            return false;
    }
    return true;
}
public static boolean filterInteger(String i, boolean f) {
    if (i == null)
        return false;
    if (i.length() == 0)
        return false;
    try {
        Integer ii = Integer.valueOf(Integer.parseInt(i));
        if (f) {
            if (ii.intValue() < 0)
                throw new IllegalArgumentException("");
        } else if (ii.intValue() <= 0)
            throw new IllegalArgumentException("");
        } catch (Exception e) {
            return false;
        }
        return true;
    }
}
```

### ShopCipher.java

```
package com.omendba.shop;
import java.io.*;
import java.security.*;
import java.security.spec.*;
import javax.crypto.*;
import javax.crypto.spec.*;
import com.omendba.shop.*;

public class ShopCipher {
    private final byte maxLen=56;
    private final String CipherCode="DESede";
    private final byte size = 24;
    private String curDir = System.getProperty("user.dir")+File.separator+"key";
```

```

private boolean CheckWord(String s) {
    if (s == null){
        return true;
    }
    boolean flag = false;
    for (int i=0; i < s.length(); i++) {
        int p = Integer.valueOf(s.charAt(i));
        if (p < 33 || p > 146) {
            flag = true;
        }
    }
    if (flag) {
        return true;
    }
    return false;
}

public String shopDecrypt(byte [] data) {
    if (data.length==0) return null;
    if (data.length>maxLen) return null;
    try {
        Cipher cipher = Cipher.getInstance(CipherCode);
        cipher.init(Cipher.DECRYPT_MODE, ReadKey(curDir));
        byte [] end = cipher.doFinal(data);
        char [] chend = new char [end.length];
        for (int i=0; i<end.length; i++) {
            chend[i] = (char)end[i];
        }
        return (new String(chend));
    } catch (Exception e) {
        String s = "com.omendba.shop.ShopCipher.shopDecrypt";
        (new LogWrite()).writeLog(e,s);
        return null;
    }
}

public byte [] shopEncrypt(String in) {
    if (in.length() == 0) {
        return null;
    };
    if (in.length() >= maxLen) {
        return null;
    }
    if (CheckWord(in)) {
        return null;
    }
    try {
        Cipher cipher = Cipher.getInstance(CipherCode);
        cipher.init(Cipher.ENCRYPT_MODE, ReadKey(curDir));
        char [] chin = new char [in.length()];
        byte [] data = new byte [in.length()];
        for (int i=0; i<in.length(); i++) {
            chin[i]=in.charAt(i);
        }
        for (int i=0; i<in.length(); i++) {
            data[i]=(byte)chin[i];
        }
        return cipher.doFinal(data);
    } catch (Exception e) {
        String s = "com.omendba.shop.ShopCipher.shopEncrypt";
        (new LogWrite()).writeLog(e,s);
        return null;
    }
}

```

```

    }
    public SecretKey ReadKey(String file) {
        byte [] byteDES = new byte[size];
        try {
            DataInputStream in = new DataInputStream(new FileInputStream(file));
            in.readFully(byteDES);
            in.close();
            DESedeKeySpec keySpec = new DESedeKeySpec(byteDES);
            SecretKeyFactory keyFactory = SecretKeyFactory.getInstance(CipherCode);
            return keyFactory.generateSecret(keySpec);
        } catch (Exception e){
            String s = "com.omendba.shop.ShopCipher.shopEnCrypt";
            (new LogWrite()).writeLog(e,s);
            return null;
        }
    }
    public void CreateKey(String file) {
        try {
            Class clDES = DESedeKeySpec.class;
            KeyGenerator keyGen = KeyGenerator.getInstance(CipherCode);
            SecretKeyFactory keyFactory = SecretKeyFactory.getInstance(CipherCode);
            SecretKey key = keyGen.generateKey();
            DESedeKeySpec keySpec = (DESedeKeySpec)keyFactory.getKeySpec(key,clDES);
            byte[] byteDES = keySpec.getKey();
            FileOutputStream out = new FileOutputStream(file);
            out.write(byteDES);
            out.close();
        } catch (Exception e){
            String s = "com.omendba.shop.ShopCipher.CreateKey";
            (new LogWrite()).writeLog(e,s);
        }
    }
}

```

#### LogWrite.java

```

package com.omendba.shop;
import java.io.*;
import java.util.*;
import org.hibernate.HibernateException;

public class LogWrite {
    private String curDir = System.getProperty("user.dir");
    public void writeLog(Exception e, String method){
        String ls = System.getProperty("line.separator");
        GregorianCalendar ca = new GregorianCalendar();
        Date date = ca.getTime();
        try {
            BufferedWriter out =
                new BufferedWriter(new FileWriter(curDir+File.separator+"shop.log", true));
            out.write("====="+ls);
            out.write(" "+date+": "+method+ls);
            out.write("====="+ls);
            if (e instanceof org.hibernate.HibernateException) {
                out.write("HibernateException: "+ls);
                HibernateException e_ = (HibernateException)e;
                String [] exc = e_.getMessages();
                for (int i=0; i<exc.length; i++) {
                    out.write(" "+i+": "+exc[i]+ls);
                }
            } else
                out.write("Exception: "+e.getMessage()+ls);
            StackTraceElement [] ste = e.getStackTrace();

```

```

    for (int i=0; i<ste.length; i++) {
        out.write(""+i+": "+ste[i]+ls);
    }
    out.close();
} catch (IOException e_){}
}
}

```

#### SendMail.java

```

package com.omendba.shop;
import java.io.*;
import java.text.*;
import java.util.*;
import javax.mail.*;
import javax.mail.internet.*;
import com.omendba.shop.table.News;
public class SendMail implements Runnable {
    private final String FILE1 = "Shop.properties";
    private final String FILE2 = "Shop_ru_RU.properties";
    private final String message1 = "<BR><BR><I><B>С уважением, администрация
магазина!</B></I>";
    private String tempDir = System.getProperty("user.dir");
    private String mail;
    private String body;
    private String site;
    private String fio;
    private String login;
    private String password;
    private String url;
    private String message;
    private DateFormat fmt = DateFormat.getDateInstance(3, Locale.getDefault());;
    public void setFio(String o){
        fio = o;
    }
    public void setPassword(String o){
        password = o;
    }
    public void setLogin(String o){
        login = o;
    }
    public void setMail(String o){
        mail = o;
    }
    public void run() {
        Properties properties = new Properties();
        Properties properties2 = new Properties();
        try {
            properties.load(new FileInputStream(tempDir+File.separator+FILE1));
            properties2.load(new FileInputStream(tempDir+File.separator+FILE2));
        } catch (Exception e){
            return;
        }
        String mailServer=properties.getProperty("sendmail.mailserver");
        String from=properties.getProperty("sendmail.from");
        String to=mail;
        String subject=properties2.getProperty(message);
        String loginSmtп=properties.getProperty("sendmail.loginsmtп");
        String passwordSmtп=properties.getProperty("sendmail.passwordsmtп");
        String messageBody=body;
        Properties props = new Properties();
        props.put("mail.smtp.host", mailServer);
        try {

```

```

Session session = Session.getInstance(props,
    new PopupAuthenticator(loginSMTP, passwordSMTP));
MimeMessage message = new MimeMessage(session);
message.setFrom(new InternetAddress(from));
message.addRecipient(Message.RecipientType.TO,
    new InternetAddress(to));
message.setSubject(subject);
Calendar calendar = Calendar.getInstance();
calendar.setTime(new Date());
message.setSentDate(calendar.getTime());
message.setContent(messageBody, "text/html");
Transport.send(message);
} catch (Exception e){
    (new LogWrite()).writeLog(e,"com.omendba.shop.SendMail.run");
}
}

public SendMail(String mail, String fio, String login, String key) {
    Properties properties = new Properties();
    try {
        properties.load(new FileInputStream(tempDir+File.separator+"Shop.properties"));
    } catch (Exception e){}
    this.mail = mail;
    this.fio = fio;
    this.site = properties.getProperty("sendmail_site");
    this.url = "http://" + site + properties.getProperty("sendmail.active") +
        "active="+key+"&login="+login;
}

public SendMail(){
    Properties properties = new Properties();
    try {
        properties.load(new FileInputStream(tempDir+File.separator+"Shop.properties"));
    } catch (Exception e){}
    this.site = properties.getProperty("sendmail.site");
}

public void setContentBodyRegistration(){
    message="order.message21";
    body = getHead() +
        "Вы получили это письмо, потому что вы зарегистрировались на " +
        getSite() +
        "Активировать вашу учетную запись можно, зайдя по " +
        "<a href='"+url+"'><b>ссылке</b></a>" + "<BR><BR>" +
        "Добро пожаловать в наш магазин!" + message1;
}

public void setContentBodyRemind(){
    message="order.message22";
    body = getHead() +
        "Вы получили это письмо, потому что вы зарегистрированы на " + getSite() +
        "Напоминаем ваш логин и пароль: <BR><BR>" +
        "<B>Логин:</B> " + login + "<BR>" + "<B>Пароль:</B> " + password +
        message1;
}

public void setContentBodyFromManager(String body){
    message="order.message22";
    this.body = getHead() +
        "Вы получили это письмо, потому что вы зарегистрированы на " +
        getSite() + body + message1;
}

public void setContentBodyNews(News news){
    message="news.message2";
    body = getHead() +
        "Вы получили это письмо, потому что вы были подписаны на рассылку " +
        "новостей от нашего магазина " + getSite();
    body+="=====  
";
}

```

```

        body+="<B>№"+news.getId()+" . "+" Дата: "+fmt.format(news.getDates())
+ "</B><BR>" +
        "=====<BR><BR>";
        body+="<B>Тема:</B> "+news.getName()+"<BR><BR>";
        body += news.getBody() + message1;
    }
    public void setContentBodyActive(boolean flag){
        message="order.message22";
        body = getHead();
        if (flag) {
            body += "Ваша учетная запись активирована!";
        } else {
            body += "Ваша учетная запись деактивирована!";
        }
        body += message1;
    }
    private String getHead() {
        return "<B>Здравствуйте "+fio+"!</B><BR><BR>";
    }
    private String getSite() {
        return "<a href='http://'+site+"'><b>"+site+"</b></a>"+ "<BR><BR>";
    }
    private class PopupAuthenticator extends Authenticator {
        private String login;
        private String password;
        public PopupAuthenticator(String login, String password){
            super();
            this.login=login;
            this.password=password;
        }
        protected PasswordAuthentication getPasswordAuthentication(){
            return new PasswordAuthentication(login, password);
        }
    }
}

```

#### **ManipulationManager.java**

```

package com.omendba.shop;
import org.hibernate.*;
import org.hibernate.cfg.*;
import org.hibernate.criterion.Criterion;
import org.hibernate.criterion.Restrictions;
public class ManipulationManager {
    private SessionFactory sf;
    private com.omendba.shop.manager.waybill.Model waybill;
    private com.omendba.shop.manager.news.Model news;
    private com.omendba.shop.manager.users.Model users;
    private com.omendba.shop.manager.goods.Model goods;
    private com.omendba.shop.manager.orders.Model orders;
    private com.omendba.shop.manager.manufacturer.Model manufacturer;
    private com.omendba.shop.manager.supplier.Model supplier;
    private com.omendba.shop.manager.category.Model category;
    private com.omendba.shop.manager.type.Model type;
    private com.omendba.shop.manager.authorization.Model authorization;
    public com.omendba.shop.manager.authorization.Model getAuthorization(){
        if (authorization == null){
            authorization = new com.omendba.shop.manager.authorization.Model(sf);
        }
        return authorization;
    }
    public com.omendba.shop.manager.type.Model getType(){
        if (type == null){

```

```

        type = new com.omendba.shop.manager.type.Model(sf);
    }
    return type;
}
public com.omendba.shop.manager.category.Model getCategory(){
    if (category == null){
        category = new com.omendba.shop.manager.category.Model(sf);
    }
    return category;
}
public com.omendba.shop.manager.supplier.Model getSupplier(){
    if (supplier == null){
        supplier = new com.omendba.shop.manager.supplier.Model(sf);
    }
    return supplier;
}
public com.omendba.shop.manager.manufacturer.Model getManufacturer(){
    if (manufacturer == null){
        manufacturer = new com.omendba.shop.manager.manufacturer.Model(sf);
    }
    return manufacturer;
}
public com.omendba.shop.manager.orders.Model getOrders(){
    if (orders == null){
        orders = new com.omendba.shop.manager.orders.Model(sf);
    }
    return orders;
}
public com.omendba.shop.manager.goods.Model getGoods(){
    if (goods == null){
        goods = new com.omendba.shop.manager.goods.Model(sf);
    }
    return goods;
}
public com.omendba.shop.manager.users.Model getUsers(){
    if (users == null){
        users = new com.omendba.shop.manager.users.Model(sf);
    }
    return users;
}
public com.omendba.shop.manager.waybill.Model getWaybill(){
    if (waybill == null){
        waybill = new com.omendba.shop.manager.waybill.Model(sf);
    }
    return waybill;
}
public com.omendba.shop.manager.news.Model getNews(){
    if (news == null){
        news = new com.omendba.shop.manager.news.Model(sf);
    }
    return news;
}
public ManipulationManager(SessionFactory sf){
    this.sf=sf;
}
}

```

#### **ShowImage.java**

```

package com.omendba.shop.web;
import java.applet.*;
import java.awt.*;
import java.net.*;

```

```

import java.io.*;
import java.util.*;

public class ShowImage extends Applet {
private boolean flag;
private String key="";
private ArrayList<Code> al = new ArrayList<Code>();
public void init() {
    for (int i=0;i<6;i++){
        int num = (int)(Math.random()*10)%3+1;
        char a=' ';
        switch (num) {
            case 1: a = (char)((int)(Math.random()*100)%25+97); break;
            case 2: a = (char)((int)(Math.random()*100)%25+65); break;
            case 3: a = (char)((int)(Math.random()*100)%9+48); ;
        }
        key+=a;
    }
    URL url=null;
    String AppletHost=String.valueOf(this.getCodeBase());
    String jsid=this.getParameter("jsid");
    try {
        url = new URL(AppletHost+"IndexAction.do;" +jsid+"?key="+key);
        URLConnection shop = url.openConnection();
        InputStreamReader isr = new InputStreamReader(shop.getInputStream());
        isr.close();
    } catch(Exception e){}
    flag=true;
    int x=10;
    for (int i=0;i<6;i++){
        Font f = null;
        Color col=null;
        int style = (int)(Math.random()*10)%2; //1..2
        switch (style) {
            case 1: f = new Font("Serif", Font.ITALIC, 23); break;
            case 2: f = new Font("Serif", Font.BOLD, 23);
        }
        int color = (int)(Math.random()*10)%5+1; //1..5
        switch (color) {
            case 1: col=Color.BLUE; break;
            case 2: col=Color.RED; break;
            case 3: col=Color.BLACK; break;
            case 4: col=Color.PINK; break;
            case 5: col=Color.GREEN;
        }
        int y=0;
        if (((i+2)%2)==0) y=16;
        else y=23;
        al.add(new Code(f,col,key.charAt(i),x,y));
        x+=9;
    }
}

public void paint(Graphics g) {
    if (flag){
        for (Code c : al) {
            g.setFont(c.getF());
            g.setColor(c.getCol());
            g.drawString(String.valueOf(c.getC()),c.getX(),c.getY());
        }
    }
}

class Code{

```



```

private Font f;
private Color col;
private char c;
private int x;
private int y;
public Font getF() {return f;}
public char getC() {return c;}
public int getX() {return x;}
public int getY() {return y;}
public Color getCol() {return col;}
public Code(Font f, Color col, char c, int x, int y){
    this.f=f;
    this.c=c;
    this.x=x;
    this.y=y;
    this.col=col;
}
}

```

#### o **schemaMysql.sql**

```

drop table if exists `Goodsorder`;
drop table if exists `Goodsbasket`;
drop table if exists `Orders`;
drop table if exists `Address`;
drop table if exists `Messageuser`;
drop table if exists `Users`;
drop table if exists `Manager`;
drop table if exists `News`;
drop table if exists `Waybill`;
drop table if exists `Goods`;
drop table if exists `Types`;
drop table if exists `Category`;
drop table if exists `Manufacturer`;
drop table if exists `Supplier`;
create table `Goods` (
  `id` int(11) not null auto_increment,
  `name` varchar(100),
  `characteristics` varchar(300),
  `guarantee` int(11),
  `supplier_id` int(11) not null,
  `manufacturer_id` int(11) not null,
  `wholesaleprice` float,
  `procurementprice` float,
  `discount` int(11),
  `image` varchar(200),
  `existss` smallint(6),
  `amount` int(11),
  `types_id` int(11) not null,
  `beginsales` date,
  primary key (`id`)
) engine=innodb default charset=utf8 row_format=dynamic;
create table `Types` (
  `id` int(11) not null auto_increment,
  `name` varchar(100),
  `existss` smallint(6),
  `category_id` int(11) not null,
  primary key (`id`)
) engine=innodb default charset=utf8 row_format=dynamic;
create table `Supplier` (
  `id` int(11) not null auto_increment,
  `name` varchar(100),
  `existss` smallint(6),

```

```

    primary key (`id`)
) engine=innodb default charset=utf8 row_format=dynamic;
create table `Manufacturer` (
`id` int(11) not null auto_increment,
`name` varchar (100),
`existss` smallint(6),
    primary key (`id`)
) engine=innodb default charset=utf8 row_format=dynamic;
create table `Waybill` (
`id` int(11) not null auto_increment,
`dates` date default null,
`amount` int(11),
`goods_id` int(11),
    primary key (`id`)
) engine=innodb default charset=utf8 row_format=dynamic;
create table `Category` (
`id` int(11) not null auto_increment,
`name` varchar (100),
`existss` smallint(6),
    primary key (`id`)
) engine=innodb default charset=utf8 row_format=dynamic;
create table `News` (
`id` int(11) not null auto_increment,
`name` varchar (100),
`body` varchar (2000),
`dates` date default null,
    primary key (`id`)
) engine=innodb default charset=utf8 row_format=dynamic;
create table `Messageuser` (
`id` int(11) not null auto_increment,
`body` varchar (400),
`users_id` int(11),
`type_mes` int(11),
`status` int(11),
    primary key (`id`)
) engine=innodb default charset=utf8 row_format=dynamic;
create table `Address` (
`id` int(11) not null auto_increment,
`country` varchar (200),
`area` varchar (200),
`city` varchar (200),
`street` varchar (200),
`house` int(11),
`users_id` int(11),
    primary key (`id`)
) engine=innodb default charset=utf8 row_format=dynamic;
create table `Goodsorder` (
`goodsbasket_id` int(11),
`orders_id` int(11)
) engine=innodb default charset=utf8 row_format=dynamic;
create table `Goodsbasket` (
`id` int(11) not null auto_increment,
`amount` int(11),
`goods_id` int(11),
    primary key (`id`)
) engine=innodb default charset=utf8 row_format=dynamic;
create table `Orders` (
`id` int(11) not null auto_increment,
`dates` date default null,
`status` int(11),
`cost` float,
`users_id` int(11),
`manager_id` int(11) not null,

```

```

`address_id` int(11) not null,
`comment` varchar (400),
primary key (`id`)
) engine=innodb default charset=utf8 row_format=dynamic;
create table `Users` (
`id` int(11) not null auto_increment,
`fio` varchar (200),
`login` varchar (50),
`passwords` blob,
`phone` int(11),
`email` varchar (100),
`lastinput` date default null,
`dispatchmail` int(11),
`keyss` varchar (32),
`actives` int(11),
primary key (`id`)
) engine=innodb default charset=utf8 row_format=dynamic;
create table Manager (
`id` int(11) not null auto_increment,
`login` varchar (50),
`passwords` blob,
`lastinput` date default null,
`existss` smallint(6),
`amountorder` int(11),
primary key (`id`)
) engine=innodb default charset=utf8 row_format=dynamic;
alter table `Goods` add constraint fk_8f0fc0730c foreign key (`supplier_id`)
references `Supplier`(`id`);
alter table `Goods` add constraint fk_79c28d3c0c foreign key (`manufacturer_id`)
references `Manufacturer`(`id`);
alter table `Goods` add constraint fk_4d8b11cf31 foreign key (`types_id`)
references `Types`(`id`);
alter table `Types` add constraint fk_5762f3dd31 foreign key (`category_id`)
references `Category`(`id`);
alter table `Waybill` add constraint fk_8fae50f40c foreign key (`goods_id`)
references `Goods`(`id`);
alter table `Orders` add constraint fk_45f0cf6d49 foreign key (`users_id`)
references `Users`(`id`);
alter table `Orders` add constraint fk_89ferf6d4a foreign key (`manager_id`)
references `Manager`(`id`);
alter table `Orders` add constraint fk_15f1c86d41 foreign key (`address_id`)
references `Address`(`id`);
alter table `Goodsorder` add constraint fk_c9d0807325 foreign key (`orders_id`)
references `Orders`(`id`);
alter table `Goodsorder` add constraint fk_2ae0817b25 foreign key
(`goodsbasket_id`) references `Goodsbasket`(`id`);
alter table `Goodsbasket` add constraint fk_65e0517525 foreign key (`goods_id`)
references `Goods`(`id`);
alter table `Address` add constraint fk_93e9e45049 foreign key (`users_id`)
references `Users`(`id`);
alter table `Messageuser` add constraint fk_2a5e1cf54b foreign key (`users_id`)
references `Users`(`id`);
alter table `Users` add constraint pk_Users1 unique (`login`);
alter table `Manager` add constraint pk_Users2 unique (`login`);

```

#### **schemaFirebird.sql**

```

drop table Goodsorder;
drop table Goodsbasket;
drop table Orders;
drop table Address;
drop table Messageuser;

```

```

drop table Users;
drop table Manager;
drop table News;
drop table Waybill;
drop table Goods;
drop table Types;
drop table Category;
drop table Manufacturer;
drop table Supplier;
create table Goods (
  id integer not null,
  name varchar (100) character set unicode_fss collate unicode_fss,
  characteristics varchar (300) character set unicode_fss collate unicode_fss,
  guarantee integer,
  supplier_id integer not null,
  manufacturer_id integer not null,
  wholesaleprice float,
  procurementprice float,
  discount integer,
  image varchar (200) character set unicode_fss collate unicode_fss,
  existss smallint,
  amount integer,
  types_id integer not null,
  beginsales date
);
create table Category (
  id integer not null,
  name varchar (100) character set unicode_fss collate unicode_fss,
  existss smallint
);
create table News (
  id integer not null,
  name varchar (100) character set unicode_fss collate unicode_fss,
  body varchar (2000) character set unicode_fss collate unicode_fss,
  dates date default null
);
create table Supplier (
  id integer not null,
  name varchar (100) character set unicode_fss collate unicode_fss,
  existss smallint
);
create table Types (
  id integer not null,
  name varchar (100) character set unicode_fss collate unicode_fss,
  existss smallint,
  category_id integer not null
);
create table Manufacturer (
  id integer not null,
  name varchar (100) character set unicode_fss collate unicode_fss,
  existss smallint
);
create table Waybill (
  id integer not null,
  dates date default null,
  amount integer,
  goods_id integer
);
create table Messageuser (
  id integer not null,
  body varchar (400) character set unicode_fss collate unicode_fss,
  users_id integer,
  type_mes integer,

```

```

status integer
);
create table Address (
id integer not null,
country varchar (200) character set unicode_fss collate unicode_fss,
area varchar (200) character set unicode_fss collate unicode_fss,
city varchar (200) character set unicode_fss collate unicode_fss,
street varchar (200) character set unicode_fss collate unicode_fss,
house integer,
users_id integer
);
o create table Goodsorder (
goodsbasket_id integer,
orders_id integer
);
create table Goodsbasket (
id integer not null,
amount integer,
goods_id integer
);
create table Orders (
id integer not null,
dates date default null,
status integer,
cost float,
users_id integer,
manager_id integer not null,
address_id integer not null,
comment varchar (400) character set unicode_fss collate unicode_fss
);
o create table Users (
id integer not null,
fio varchar (200) character set unicode_fss collate unicode_fss,
login varchar (50) character set unicode_fss collate unicode_fss,
passwords blob sub_type 0 segment size 80,
phone integer,
email varchar (100) character set unicode_fss collate unicode_fss,
lastinput date default null,
dispatchmail integer,
keyss varchar (32) character set unicode_fss collate unicode_fss,
actives integer
);
create table Manager (
id integer not null,
login varchar (50) character set unicode_fss collate unicode_fss,
passwords blob sub_type 0 segment size 80,
lastinput date default null,
existss smallint,
amountorder integer
);
o alter table Goods add constraint pk_313ae5960b primary key (id);
alter table Goodsbasket add constraint pk_2535e59601 primary key (id);
alter table Types add constraint pk_83f818010c primary key (id);
alter table Supplier add constraint pk_ba0c2c630c primary key (id);
alter table Manufacturer add constraint pk_1482f2860c primary key (id);
alter table Waybill add constraint pk_99620dcc0c primary key (id);
alter table category add constraint pk_8ce60dc531 primary key (id);
alter table News add constraint pk_eadlfa3f4a primary key (id);
alter table Manager add constraint pk_b765d70d4d primary key (id);
alter table Users add constraint pk_1a1f89e748 primary key (id);
alter table Orders add constraint pk_ee70c81725 primary key (id);
alter table Address add constraint pk_1212192749 primary key (id);
alter table Messageuser add constraint pk_b74d3ff04b primary key (id);

```

```

alter table Goods add constraint fk_8f0fa0730c foreign key (supplier_id)
references Supplier(id);
alter table Goods add constraint fk_7ac28d8c0c foreign key (manufacturer_id)
references Manufacturer(id);
alter table Goods add constraint fk_4d8b11cf31 foreign key (types_id) references
Types(id);
alter table Types add constraint fk_5762f3dd31 foreign key (category_id)
references category(id);
alter table Waybill add constraint fk_8fae50f40c foreign key (goods_id)
references Goods(id);
alter table Orders add constraint fk_45f0cf6d49 foreign key (users_id)
references Users(id);
alter table Orders add constraint fk_89ferf6d4a foreign key (manager_id)
references Manager(id);
alter table Goodsorder add constraint fk_c9d0807325 foreign key (orders_id)
references Orders(id);
alter table Goodsorder add constraint fk_2ae0817b25 foreign key (goodsbasket_id)
references Goodsbasket(id);
alter table Goodsbasket add constraint fk_65e0517525 foreign key (goods_id)
references Goods(id);
alter table Address add constraint fk_93e9e45049 foreign key (users_id)
references Users(id);
alter table Messageuser add constraint fk_2a5e1cf54b foreign key (users_id)
references Users(id);
alter table Orders add constraint fk_15f1c86d41 foreign key (address_id)
references Address(id);
alter table Users add constraint pk_Users1 unique (login);
alter table Manager add constraint pk_Users2 unique (login);
drop generator hibernate_sequence1;
drop generator hibernate_sequence2;
drop generator hibernate_sequence3;
drop generator hibernate_sequence4;
drop generator hibernate_sequence5;
drop generator hibernate_sequence6;
drop generator hibernate_sequence7;
drop generator hibernate_sequence8;
drop generator hibernate_sequence9;
drop generator hibernate_sequence10;
drop generator hibernate_sequence11;
drop generator hibernate_sequence12;
drop generator hibernate_sequence13;
create generator hibernate_sequence1;
create generator hibernate_sequence2;
create generator hibernate_sequence3;
create generator hibernate_sequence4;
create generator hibernate_sequence5;
create generator hibernate_sequence6;
create generator hibernate_sequence7;
create generator hibernate_sequence8;
create generator hibernate_sequence9;
create generator hibernate_sequence10;
create generator hibernate_sequence11;
create generator hibernate_sequence12;
create generator hibernate_sequence13;

```

#### **struts-config.xml**

```

<?xml version="1.0"?>
<!DOCTYPE struts-config PUBLIC
"-//Apache Software Foundation//DTD Struts Configuration 1.2//EN"
"http://jakarta.apache.org/struts/dtds/struts-config_1_2.dtd">
<struts-config>
<global-forwards>

```

```

    <forward name="logon" path="/manager/logon.jsp"/>
    <forward name="report" path="/manager/report.jsp"/>
</global-forwards>
<message-resources parameter="Message"/>
<form-beans>
    <form-bean name="indexForm" type="com.omendba.shop.form.IndexForm"/>
    <form-bean name="managerForm" type="com.omendba.shop.form.ManagerForm"/>
    <form-bean name="loginForm" type="com.omendba.shop.form.LoginForm"/>
    <form-bean name="reportForm" type="com.omendba.shop.form.ReportForm"/>
</form-beans>
<action-mappings>
    <action path="/IndexAction"
        type="com.omendba.shop.action.IndexAction"
        input="/client/index.jsp"
        validate="true"
        name="indexForm">
        <forward name="success" path="/client/index.jsp"/>
    </action>
    <action path="/ManagerAction"
        type="com.omendba.shop.action.ManagerAction"
        name="managerForm"
        validate="true"
        input="/manager/manager.jsp">
        <forward name="success" path="/manager/manager.jsp"/>
        <forward name="report" path="/manager/report.jsp"/>
        <forward name="logon" path="/manager/logon.jsp"/>
    </action>
    <action path="/ReportAction"
        type="com.omendba.shop.action.ReportAction"
        name="reportForm">
        <forward name="success" path="/manager/manager.jsp"/>
        <forward name="logon" path="/manager/logon.jsp"/>
    </action>
    <action path="/LoginAction"
        type="com.omendba.shop.action.LoginAction"
        validate="true"
        name="loginForm"
        input="/manager/logon.jsp">
        <forward name="success" path="/manager/manager.jsp"/>
        <forward name="logon" path="/manager/logon.jsp"/>
    </action>
</action-mappings>
</struts-config>

```

#### hibernate.cfg.xml

```

<?xml version='1.0' encoding='utf-8'?>
<!DOCTYPE hibernate-configuration PUBLIC
"-//Hibernate/Hibernate Configuration DTD 3.0//EN"
"http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">
<hibernate-configuration>
<session-factory>
    <property name="connection.driver_class">com.mysql.jdbc.Driver</property>
    <property name="connection.url">jdbc:mysql://localhost:3307/poterenkoag?
autoReconnect=true</property>
    <property name="connection.username">poterenkoag</property>
    <property name="connection.password">HG#78EG#L@jy4t364mjg) ( ) ? !</property>
    <property name="connection.pool_size">10</property>
    <property name="connection.isolation">4</property>
    <property name="connection.autocommit">>false</property>
    <property name="dialect">org.hibernate.dialect.MySQL5InnoDBDialect</property>
    <property name="current_session_context_class">thread</property>
</session-factory>
</hibernate-configuration>
</mapping resource="shop.hbm.xml"/>

```

```
</session-factory>
</hibernate-configuration>
```

### shop.hbm.xml

```
<?xml version="1.0"?>
<!DOCTYPE hibernate-mapping PUBLIC
"-//Hibernate/Hibernate Mapping DTD 3.0//EN"
"http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd">
<hibernate-mapping>
<class name="com.omendba.shop.table.Waybill" table="Waybill">
<id name="id" type="int" column="id">
<generator class="native"/>
</id>
<property name="dates" type="date"/>
<property type="int" name="amount"/>
<many-to-one
name="goodsId"
column="goods_Id"
class="com.omendba.shop.table.Goods"
not-null="true"/>
</class>
<class name="com.omendba.shop.table.News" table="News">
<id name="id" type="int" column="id">
<generator class="native"/>
</id>
<property type="string" name="name"/>
<property type="string" name="body"/>
<property type="date" name="dates"/>
</class>
<class name="com.omendba.shop.table.Messageuser" table="Messageuser">
<id name="id" type="int" column="id">
<generator class="native"/>
</id>
<property type="string" name="body"/>
<property type="int" name="status"/>
<property type="int" name="typeMessage" column="type_mes"/>
<many-to-one
name="userId"
column="users_Id"
class="com.omendba.shop.table.User"
not-null="true"
/>
</class>
<class name="com.omendba.shop.table.Goods" table="Goods">
<id name="id" type="int" column="id">
<generator class="native"/>
</id>
<property type="string" name="name"/>
<property type="string" name="characteristics"/>
<property type="string" name="image"/>
<property type="short" name="exists" column="existss"/>
<property type="int" name="guarantee"/>
<property type="int" name="discount"/>
<property type="int" name="amount"/>
<property type="float" name="wholesaleprice"/>
<property type="float" name="procurementprice"/>
<property type="date" name="beginsales"/>
<many-to-one
name="supplierId"
column="supplier_Id"
class="com.omendba.shop.table.Supplier"
not-null="true"
```



```

/>
<many-to-one
name="manufacturerId"
column="manufacturer_Id"
class="com.omendba.shop.table.Manufacturer"
not-null="true"
/>
<many-to-one
name="typeId"
column="types_Id"
class="com.omendba.shop.table.Type"
not-null="true"
/>
<list name="waybills" >
<key column="goods_Id" />
<index column="id" />
<one-to-many class="com.omendba.shop.table.Waybill"/>
</list>
</class>
<class name="com.omendba.shop.table.Supplier" table="Supplier">
<id name="id" type="int" column="id">
<generator class="native"/>
</id>
<property type="string" name="name"/>
<property type="short" name="exists" column="existss"/>
<list name="goodses" >
<key column="supplier_Id"/>
<index column="id"/>
<one-to-many class="com.omendba.shop.table.Goods" />
</list>
</class>
<class name="com.omendba.shop.table.Manufacturer" table="Manufacturer">
<id name="id" type="int" column="id">
<generator class="native"/>
</id>
<property type="string" name="name"/>
<property type="short" name="exists" column="existss"/>
<list name="goodses" >
<key column="manufacturer_Id"/>
<index column="id"/>
<one-to-many class="com.omendba.shop.table.Goods" />
</list>
</class>
<class name="com.omendba.shop.table.Category" table="Category">
<id name="id" type="int" column="id">
<generator class="native"/></id>
<property type="string" name="name"/>
<property type="short" name="exists" column="existss"/>
<list name="types" >
<key column="category_Id"/>
<index column="id"/>
<one-to-many class="com.omendba.shop.table.Type" />
</list>
</class>
<class name="com.omendba.shop.table.Type" table="Types">
<id name="id" type="int" column="id">
<generator class="native"/>
</id>
<property type="string" name="name"/>
<property type="short" name="exists" column="existss"/>
<many-to-one
name="categoryId"
column="category_Id"

```

```

class="com.omendba.shop.table.Category"
not-null="true"/>
<list name="goodses" >
<key column="types_Id"/>
<index column="id" />
<one-to-many class="com.omendba.shop.table.Goods" />
</list>
</class>
<class name="com.omendba.shop.table.Goodsbasket" table="Goodsbasket">
<id name="id" type="int" column="id">
<generator class="native"/>
</id>
<property type="int" name="amount"/>
<many-to-one
name="goodsId"
column="goods_Id"
class="com.omendba.shop.table.Goods"
not-null="true"/>
</class>
<class name="com.omendba.shop.table.User" table="Users">
<id name="id" type="int" column="id">
<generator class="native"/>
</id>
<property type="string" name="fio"/>
<property type="string" name="login"/>
<property type="byte[]" name="passwords"/>
<property type="int" name="phone"/>
<property type="string" name="email"/>
<property type="date" name="lastinput"/>
<property type="int" name="dispatchmail"/>
<property type="string" name="keys" column="keyss"/>
<property type="int" name="actives"/>
<list name="orders" >
<key column="users_Id"/>
<index column="id"/>
<one-to-many class="com.omendba.shop.table.Order" />
</list>
</class>
<class name="com.omendba.shop.table.Address" table="Address">
<id name="id" type="int" column="id">
<generator class="native"/>
</id>
<property type="string" name="country"/>
<property type="string" name="area"/>
<property type="string" name="city"/>
<property type="string" name="street"/>
<property type="int" name="house"/>
<many-to-one
name="userId"
column="users_Id"
class="com.omendba.shop.table.User"
not-null="true"/>
</class>
<class name="com.omendba.shop.table.Order" table="Orders">
<id name="id" type="int" column="id">
<generator class="native"/>
</id>
<property type="date" name="dates"/>
<property type="int" name="status"/>
<property type="float" name="cost"/>
<property type="string" name="comment"/>
<many-to-one
name="userId"

```

```

column="users_Id"
class="com.omendba.shop.table.User"
not-null="true"/>
<many-to-one
name="addressId"
column="address_Id"
class="com.omendba.shop.table.Address"
not-null="true"/>
<many-to-one
name="managerId"
column="manager_Id"
class="com.omendba.shop.table.Manager"
not-null="true"/>
<set name="goodsorders" table="Goodsorder">
<key column="orders_id"/>
<many-to-many column="goodsbasket_Id"
class="com.omendba.shop.table.Goodsbasket"/>
</set>
</class>
<class name="com.omendba.shop.table.Manager" table="Manager">
<id name="id" type="int" column="id">
<generator class="native"/>
</id>
<property type="string" name="login"/>
<property type="byte[]" name="passwords"/>
<property type="date" name="lastinput"/>
<property type="short" name="exists" column="existss"/>
<property type="int" name="amountorder"/>
<list name="orders" >
<key column="manager_Id"/>
<index column="id"/>
<one-to-many class="com.omendba.shop.table.Order" />
</list>
</class>
</hibernate-mapping>

```

### index.jsp

```

<%@page contentType="text/html"%>
<%@page page%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://struts.apache.org/tags-bean" prefix="bean" %>
<%@ taglib uri="http://struts.apache.org/tags-html" prefix="html" %>
<%@ taglib uri="http://struts.apache.org/tags-logic" prefix="logic" %>
<logic:notPresent name="index" scope="session">
<logic:redirect action="IndexAction"/>
</logic:notPresent>
<c:set var="L00" value="440" scope="request"/>
<c:set var="Y1" value="205" scope="request"/>
<c:set var="Y40" value="647" scope="request"/>
<c:set var="Y4" value="{Y40}" scope="request"/>
<c:set var="Z0" value="20" scope="request"/>
<c:set var="Z1" value="20" scope="request"/>
<c:set var="M" value="0" scope="request"/>
<c:set var="N" value="0" scope="request"/>
<c:set var="D2" value="2" scope="request"/>
<c:set var="L0" value="0" scope="request"/>
<logic:present scope="request" name="listGoods">
<c:set var="K0" value="34" scope="request"/>
<c:set var="K1" value="40" scope="request"/>
<c:set var="P0" value="180" scope="request"/>
<c:set var="D0" value="139" scope="request"/>
<c:set var="G0" value="2" scope="request"/>

```

```

</logic:present>
<logic:present scope="request" name="listBasket">
<c:set var="Y2" value="239" scope="request"/>
<c:set var="P0" value="200" scope="request"/>
<c:set var="D0" value="34" scope="request"/>
<c:set var="K0" value="54" scope="request"/>
</logic:present>
<html>
<head>
<meta http-equiv="imagetoolbar" content="no">
<link rel="stylesheet" type="text/css" href="image/style.css" id="sitestyles">
</head>
<jsp:include page="/client/prompt.jsp"/>
<jsp:include page="/client/about.jsp"/>
<jsp:include page="/client/delivery.jsp"/>
<jsp:include page="/client/heading.jsp"/>
<jsp:include page="/client/filter.jsp"/>
<jsp:include page="/client/registration.jsp"/>
<jsp:include page="/client/authorization.jsp"/>
<jsp:include page="/client/message.jsp"/>
<jsp:include page="/client/messageuser.jsp"/>
<jsp:include page="/client/cabinet.jsp"/>
<jsp:include page="/client/profile.jsp"/>
<jsp:include page="/client/address.jsp"/>
<jsp:include page="/client/historyorder.jsp"/>
<jsp:include page="/client/contentsorder.jsp"/>
<jsp:include page="/client/find.jsp"/>
<jsp:include page="/client/orderother.jsp"/>
<jsp:include page="/client/basket.jsp"/>
<jsp:include page="/client/private.jsp"/>
<jsp:include page="/client/goods.jsp"/>
<jsp:include page="/client/catalogue.jsp"/>
<jsp:include page="/client/minibasket.jsp"/>
<jsp:include page="/client/repair.jsp"/>
<jsp:include page="/client/news.jsp"/>
<div>
<div class="default_style" style="position:absolute;
top:135; left:1; width:1003; height:34; border-width:1; border-style:solid;
border-color:#0E6D99; background-color:#9CBEDB;"></div>
<logic:present scope="session" name="currentChoose">
<html:link page="/IndexAction.do?noaction=es"
styleClass="link_style"
style="top:142; left:12; width:80; height:30;"
transaction="true">
<bean:message key="index.main"/> /
</html:link>
<div class="link_style" style="top:142; left:71; width:480; height:30;
color:#1E90FF">
<bean:write scope="session" name="currentChoose"/>
</div>
</logic:present>
<logic:notPresent scope="session" name="currentChoose">
<div class="link_style" style="top:142; left:12; width:80; height:30;">
<bean:message key="index.main"/>
</div>
</logic:notPresent>
</div>
<div class="default_style" style="position:absolute;
top:523; left:5; width:170; height:122; background-color:#1180AB;"></div>
<div class="subheading" style="position:absolute; top:520; left:6; width:110;
height:30; color:#CADEE9; font-size:15px;">
<bean:message key="index.mail"/>
</div>

```

```

<div class="subheading" style="position:absolute; top:540; left:6; width:200;
height:20; color:#000000; font-size:11px;">
  <a href="mailto:<bean:message key="index.email"/>">
    <bean:message key="index.email"/>
  </a>
</div>
<div class="default_style" style="position:absolute;
top:432; left:5; width:170; height:90; background-color:#1180AB;">
</div>
<div class="subheading" style="position:absolute;
top:430; left:6; width:110; height:30; color:#CADEE9; font-size:15px;">
  <bean:message key="index.adviser"/>
</div>
<div class="subheading" style="position:absolute;
top:450; left:6; width:160; height:70; color:#FFFFFF; font-size:11px;">
  <bean:message key="index.adviser.info"/>
</div>
<div class="subheading" style="position:absolute;
top:496; left:6; width:200; height:20; color:#000000; font-size:11px;">
  <bean:message key="index.adviser.cellular"/>
</div>
<logic:present scope="request" name="listGoods">
  <c:set var="Y3" value="\${Y1+D0+G0+K0+K1+P0*N}"/>
  <c:if test="\${(Y40>=Y3+D2) && (Y40>=Y1+L0+D2)}">
    <c:set var="Y4" value="\${Y40}" scope="request"/>
  </c:if>
  <c:if test="\${(Y40<=Y3+D2) && (Y1+L0+D2<=Y3+D2)}">
    <c:set var="Y4" value="\${Y3+D2}" scope="request"/>
  </c:if>
  <c:if test="\${(Y3+D2<=Y1+L0+D2) && (Y40<=Y1+L0+D2)}">
    <c:set var="Y4" value="\${Y1+L0+D2}" scope="request"/>
  </c:if>
</logic:present>
<logic:present scope="request" name="listBasket">
  <c:set var="Y3" value="\${Y2+K0+P0*N}"/>
  <c:if test="\${(Y40>=L0+Y1) && (Y40>=Y3+D2)}">
    <c:set var="Y4" value="\${Y40}" scope="request"/>
  </c:if>
  <c:if test="\${(Y3+D2>=Y40) && (Y3+D2>=L0+Y1)}">
    <c:set var="Y4" value="\${Y3+D2}" scope="request"/>
  </c:if>
  <c:if test="\${(L0+Y1>=Y40) && (L0+Y1>=Y3+D2)}">
    <c:set var="Y4" value="\${L0+Y1+D2}" scope="request"/>
  </c:if>
</logic:present>
<jsp:include page="/client/bottom.jsp"/>
</body>
</html>

```

#### orderother.jsp

```

<%@page contentType="text/html"%>
<%@page page%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://struts.apache.org/tags-bean" prefix="bean" %>
<%@ taglib uri="http://struts.apache.org/tags-html" prefix="html" %>
<%@ taglib uri="http://struts.apache.org/tags-logic" prefix="logic" %>
<html>
<head>
  <meta http-equiv="imagetoolbar" content="no">
  <link rel="stylesheet" type="text/css" href="image/style.css" id="sitestyles">
</head>
<body>

```

```

<logic:notPresent name="privateform" scope="session">
  <logic:redirect action="IndexAction"/>
</logic:notPresent>
<logic:present name="orderother" scope="request">
<div>
  <div class="default_style" style="position:absolute;
top:205; left:176; width:591; height:20; border-width:1; border-style:solid;
border-color:#0D6D98; background-color:#9CBEDC;"></div>
  <div class="default_style" style="position:absolute;
top:224; left:176; width:591; height:421; border-width:1; border-style:solid;
border-color:#0E6D99; background-color:#F3F8FA;"></div>
  <div class="line_style" style="top:460; left:184; width:570;"></div>
  <div class="text_style_large"
style="top:435; left:183; width:200; height:30;">
    <bean:message key="orderother.chooseaddress"/>
  </div>
  <div class="text_style_large_head"
style="top:206; left:180; width:560;">
    <bean:message key="orderother.head"/>
  </div>
  <div class="text_style_large"
style="top:236; left:183; width:570; height:40;">
    <bean:message key="orderother.message1"/>
  </div>
  <div class="text_style_large"
style="top:466; left:183; width:570; height:30;">
    <bean:message key="orderother.message2"/>
  </div>
  <logic:present name="currentAddress">
    <input type="text"
class="edit_style_readonly_large"
readonly="true"
style="top:435; left:383; width:371; height:23;"
value="<bean:write scope="request" name="currentAddress"/>">
  </logic:present>
  <logic:notPresent name="currentAddress">
    <input type="text"
class="edit_style_readonly_large"
readonly="true"
style="top:435; left:383; width:371; height:23;"
value="<bean:message key="orderother.noneaddress"/>">
  </logic:notPresent>
  <html:form action="IndexAction.do">
    <html:submit styleClass="button_style"
property="submit_back_basket"
style="top:206; left:733; width:29; height:18;
background-image:url('image/back.png') value=" "/>
  </html:form>

  <html:form action="IndexAction.do">
    <html:submit styleClass="button_style"
property="submit_order_commit"
style="top:600; left:649; width:110; height:24;">
      <bean:message key="orderother.order"/>
    </html:submit>
    <html:textarea property="comment"
styleClass="area_style_write"
rows="5"
cols="20"
style="top:490; left:184; width:574; height:100;"
readonly="false"/>
    <input type="hidden"
name="currentAddressId"

```

```

        value="<c:out value="\${requestScope.currentAddressId}"/>">
    </html:form>
</div>
<logic:present scope="request" name="listAddress">
    <div class="table_style"
style="top:270; left:180; width:580; height:160;">
        <table cellSpacing="2"
            cellPadding="2"
            width="100%"
            border="1">
            <tr align="middle" bgColor=#0D6D98>
o    <c:set var="Head"
value="address.num,address.country,address.area,address.city,address.street,addr
ess.house"/>
        <c:forEach var="i" items="\${Head}">
        <td><div class="table_style_td_head">
            <bean:message key="\${i}"/>
        </div></td>
        </c:forEach>
        </tr>
        <c:set var="_id" value="1"/>
        <logic:iterate id="list" name="listAddress" scope="request">
        <tr align="middle" bgColor=#0D6D98>
        <c:set var="Head" value="addressId,country,area,city,street,house"/>
        <c:forEach var="i" items="\${Head}">
        <td><div class="table_style_td">
            <c:if test="\${i==&apos;addressId&apos;}">
            <html:link page="/IndexAction.do"
                paramId="currentAddressId"
                paramName="list"
                paramProperty="addressId"
                transaction="true">
o            <c:out value="\${_id}"/>
            </html:link>
            </c:if>
            <c:if test="\${i!=&apos;addressId&apos;}">
            <bean:write name="list" property="\${i}"/>
            </c:if>
            </div></td>
        </c:forEach>
        </tr>
        <c:set var="_id" value="\${_id+1}"/>
        </logic:iterate>
        </table>
        </div>
        </logic:present>
        <logic:notPresent scope="request" name="listAddress">
            <div class="text_style_large"
style="top:273; left:183; width:500; height:20;">
                <bean:message key="address.none"/>
            </div>
o        </logic:notPresent>
        </logic:present>
    </body>
</html>

```