Final Project Pharse II

Java SSM Based Online Shopping Website

Team members:
Zeyu Pan (615) Email:zpan8@jhu.edu
Chen Wang (615) Email: cwang129@jhu.edu
Shanshan Yang (615) Email: syang91@jhu.edu

Our project is online shopping website with data acquired from https://www.tmall.com/, which supports all the major functions of any shopping websites, such as user register, login, logout, list products by category, search by keyword, check product in detail and it also implements the entire process of shopping, like add to cart, checkout, generate orders, keep track of orders, review products. Users can browse items according to overall score (asc or desc), post time (asc or desc), popularity (asc or desc), price (asc or desc), sold amount (asc or desc) and price range. Users can also see item details and place order.

How to see result：You can see the result from our demo:
https://drive.google.com/drive/my-drive
or see from part 6 Project demonstration and summerization

Code Link:
https://drive.google.com/file/d/1HqK6k2fDbuRKItx_WtJauhcLAM1SxtB-/view?usp=sharing

Contents

Main Body **3**
1. How the data loaded and related code for processing, extracting or loading: 3
2. User guide 6
Getting Started 6
Prerequisites 6
Installing 6
3. Major areas of specialization 7
4. Project Strengths and selling points 7
Spring framework 8
Spring MVC framework 8
MyBatis framework 8
MyBatis Generator 8
Back-End 9
Front-End 9
Built With 9
5. Project limitations, improvements and extensions 10
6. Project demonstration and summerization 10
Front-End Display 10
Back-End Display 15
7. Full relational table specification of database in SQL DDL 17
8. Hard copy of all SQL Code and additional code used for data acquisition and input 20

Appendix **26**
DataBase Final Project Proposal 26

Main Body

1. How the data loaded and related code for processing, extracting or loading:

We collect the data form Chinese famous online shopping website Tmall: https://www.tmall.com/ and inspect Element using google chrome to parse html and find the link address, header, query conditions, and in some special conditions using cookie files to get Jsonp data, and convert it into Json format and subtract it and save the date into excel. Also subtract the link of images and save the link to excel. After processing read and load the data from excel to my sql database.

Code implementation in python:

```
def getHtml(url,daili='',postdata={}):
    """
    crawler website: support cookie
    the first parameter is website, the seconde parameter is POST data
    """
    # COOKIE file path
    filename = 'cookie.txt'
    # declear a MozillaCookieJar object and save it into file
    cj = http.cookiejar.MozillaCookieJar(filename)
    # read the content of cookie from file
    # ignore_discard means save cookies even if it will be abandoned soon
    # ignore_expires means if the cookies is already exist in the
    # file, override it
```

```
    # if exist, then read main COOKIE
    if os.path.exists(filename):
        cj.load(filename, ignore_discard=True, ignore_expires=True)
    # read other COOKIE
    if os.path.exists('../subcookie.txt'):
        cookie = open('../subcookie.txt', 'r').read()
    else:
        cookie='ddd'
    # build proxy support with cookie handker
    proxy_support = urllib.request.ProxyHandler({'http':'http://'+daili})
    # open proxy support
    if daili:
        print('proxy:'+daili+'open')
        opener = urllib.request.build_opener(proxy_support,
urllib.request.HTTPCookieProcessor(cj), urllib.request.HTTPHandler)
    else:
        opener =
urllib.request.build_opener(urllib.request.HTTPCookieProcessor(cj))


    # open header
    opener.addheaders = [('User-Agent',
```

```
                        'Mozilla/5.0 (iPad; U; CPU OS 4_3_3 like Mac OS
X; en-us) AppleWebKit/533.17.9 (KHTML, like Gecko) Version/5.0.2 Mobile/8J2
Safari/6533.18.5'),

                        ('Referer',

                         'http://s.m.taobao.com'),

                        ('Host', 'h5.m.taobao.com'),

                        ('Cookie',cookie)]



    # dispatch opener

    urllib.request.install_opener(opener)

    # if data need POST

    if postdata:

        # data URL encoding

        postdata = urllib.parse.urlencode(postdata)



        # crawler html

        html_bytes = urllib.request.urlopen(url, postdata.encode()).read()

    else:

        html_bytes = urllib.request.urlopen(url).read()



    # save COOKIE into file

    cj.save(ignore_discard=True, ignore_expires=True)
```

```
    return html_bytes
```

2. User guide

Getting Started

Just like building many other Java projects, before running it, you need to install Java and a suitable IDE on our PC.

Prerequisites

Firstly, you need to install Java JDK, JRE and set some environmental variables on your operating system. Here are some Java install instructions.

Install Java JDK in Windows

Install Java JDK in Ubuntu

The next step is to select an IDE, I recommend you to use Eclipse or IntelliJ IDEA. Here are the links of their download page.

Download Eclipse

Download IntelliJ IDEA

Installing

After copying the project on your local machine, you need to add it into your IDE as a project.

The process is simple. Take IntelliJ IDEA as an example, click File->Open File or Project, select the "pom.xml" file under the main folder, which allows the IDE to open the project with Maven.

To run the project, you need to install Maven at first:

Install maven in Windows

Install maven in Ubunto

Next in the Run/Debug configuration page, you must add a maven builder.

As for the maven builder, you should set the setting file and repository as the folder where you install the maven, like:

```
mavenFolder\.m2\settings.xml
mavenFolder\.m2\repository
```

You should use tomcat as server and set it listening to the 8080 port as default. To start the server, you can set the command line parameter as:

```
tomcat7:run
```

Once you start the server, you can visit the home page by typing the URL below on your browser:

```
http://localhost:8080/tmall_ssm/home
```

3. Major areas of specialization

(1) **Complex extraction of real data from online sources:** We extract the product information from the real shopping website Tmall using the web crawler technique and preprocess the data to present their attributes and relationship with 9 database tables.

(2) **Original and challenging data modeling:** After preprocessing the data, there are 9 tables. For example, the Product table contains the features of each product like its id, name, original price, promote price and stock number and its category id. Another table is the category table which contains its id and name. So there exists a many to one relationship between category and products, that for one category it has multiple products while each product belongs to one category.

(3) **Transaction Control:** Our transaction control allows concurrent read but for writing operation like user writing comments, adding products into the cart and checkout, in order to keep the data consistent, we set the level of consistency as repeatable read that is only the committed records to be read, repeated reads of same record must return same value. However, a transaction may not be serializable, it may find some records inserted by a transaction by a transaction but not find others.

4. Project Strengths and selling points

Technology Stack

SMM framework

In this project we integrate three popular franworks, Spring, Spring MVC and Mybatis, which greatly shorter the time to develop a project.

Spring framework

（1） We use its ability of Inversion of Control to let Spring take over the relation control between objects, which reduce the coherence and shorten the period of development.

（2） Using the ability of Aspect oriented programming to support database transaction, l og and previlege control.

Spring MVC framework

（1） Compatible with Spring framework

（2） It is convenient to deal with data, for example data authentication, data format and data binding mechanism.

MyBatis framework

(1) We write the SQL statement inside the XML which is much easier to control and optimize, and it lowers the degree of coupling between Java code and SQL.
(2) It supports tag mapping and can map attributes in class and features in database.
(3) It supports dynamic SQL editing.

MyBatis Generator

MyBatis Generator is a code generator for MyBatis. It can generate code for all versions of MyBatis. I can generate several classes based on the features of data in database. As a result, we don't need to create data or configuration file when we try to modify or access to data. MyBatis implements most of simple database operations like CRUD(Create, Retrieve, Update, Delete), while for complex operations including join queries and stored procedure, we still need to write SQL code by hand.

MyBatis Generator can generate:

(1) Java POJO classes matching the structure of database tables, which includes:
    (a) a class to match the primary key of the table.
    (b) a class to match the non-primary key fields of the table.
    (c) a class to enable dynamic selects, updates, and deletes

    There exists inheritance relationship between several classes, Using generator with different configuration we can generate different structure of inheritance levels.

(2) MyBatis is compatible for SQL mapping XML files. It can generate SQL of CRUD operations for all the tables, which includes:
    (a) update by primary key

(b) update by example (using a dynamic where clause)
(c) delete by primary key
(d) delete by example (using a dynamic where clause)
(e) select by primary key
(f) select by example (using a dynamic where clause)
(g) count by example

Based on the structure of table, the generated SQL may vary like if the table does not possess primary key, there will not be a method named update by primary key.

Code Structure

Back-End

The backend is implemented with Java and all the codes are under the org.pzy.tmall source directory. According to their function and responsibility, they are distributed into 6 sub-directories.

- controller : all the Spring MVC controllers, which directly handle the HTTP request.
- interceptor: check user login status, being called before requests are handled by controller.
- mapper: a series of interfaces, used to execute SQL command defined in .xml files to CRUD data.
- pojo: all the data types, which are in accord with the data defined in database.
- service: define several APIs called by controller, handle business logic by using the method in mapper.
- util: several utilities to process special kinds of data like uploaded images.

Front-End

All the frontend codes are written in .jsp files with Java, Vue.js, HTML, Bootstrap and jQuery. They are distributed in three main folders ;

- admin: all the codes of data management pages, which are only accessible for administrators.
- fore: the pages visible for all normal customers.
- include: some pages that are integrated into the pages in previous folders, and some shared pages like head and foot.

Built With

- Spring, MyBatis, Spring MVC- The back-end web framework used
- Vue.js, jQuery, Bootstrap- The front-end web framework used
- Maven - Dependency Management

5. Project limitations, improvements and extensions

1) The limitations for our project is that because all the data we acquired are from a Chinese website, all the data are written in Chinese, which is hard for foreigners to read and get use it. If we are given additional time, we will also make an all English version website and convert all the files and data into English.
2) For back-end, we can add an administrator module and subdivide the rights to several levels and give rights to administrator and let them maintain the website based on their leaves.
3) We can also add a feature to sort the picture base on different categories like colour, seller etc.
4) Add the function to allow user to use coupon or red package to save money

6. Project demonstration and summerization

DEMO: https://drive.google.com/drive/my-drive

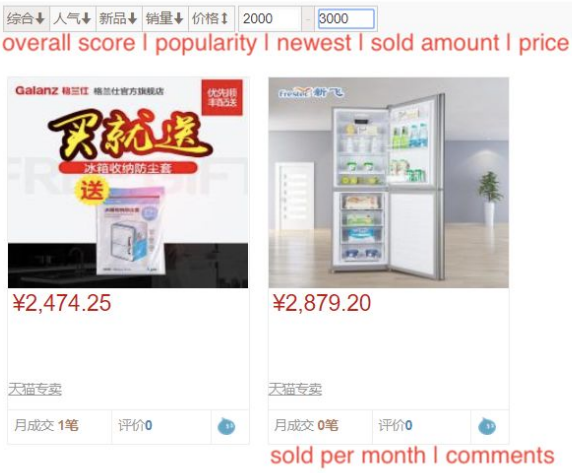
PS: Translation and commont for Chinese are witten in **red color**.

Front-End Display

(1) This is the homepage of our website:

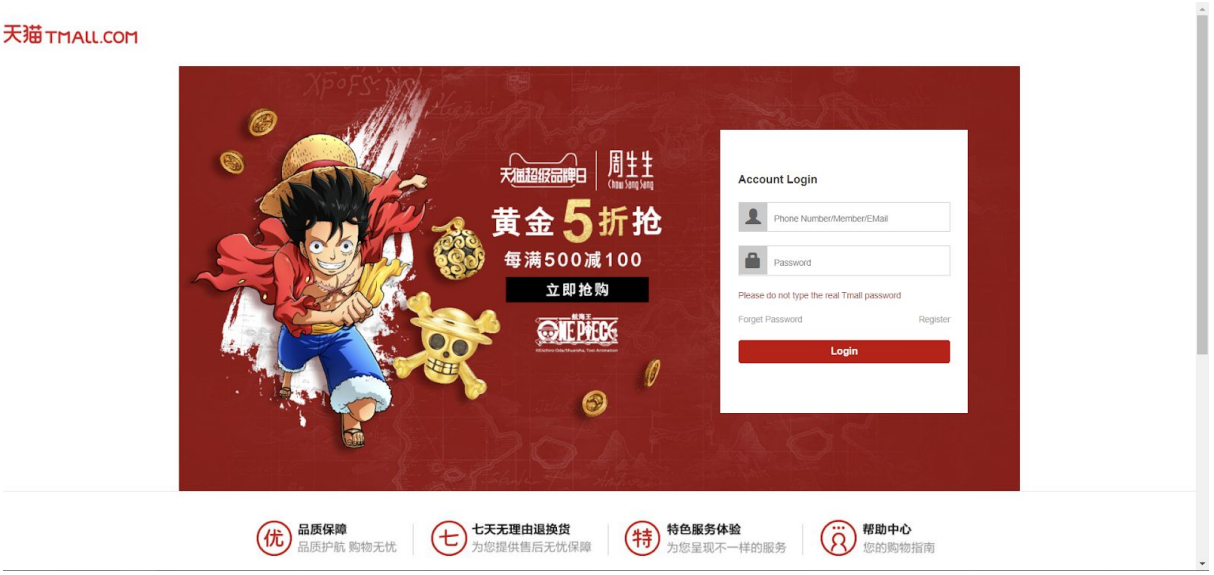
(2) List of all the products in “Refrigerator” category:

(3) List of all the refrigerators with price between 1000 and 2000:

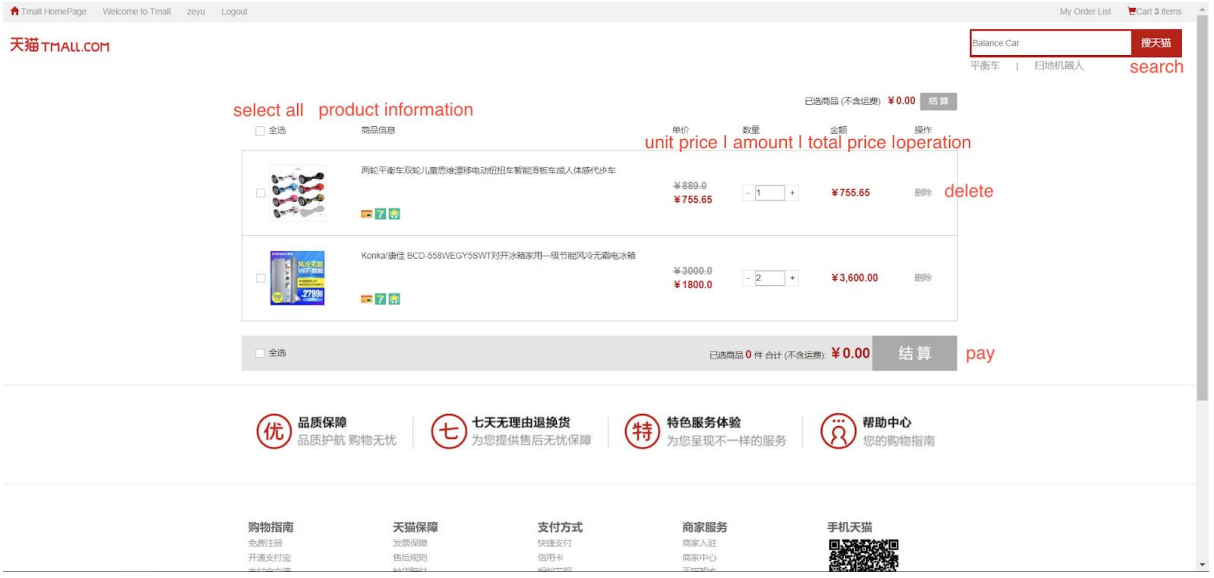


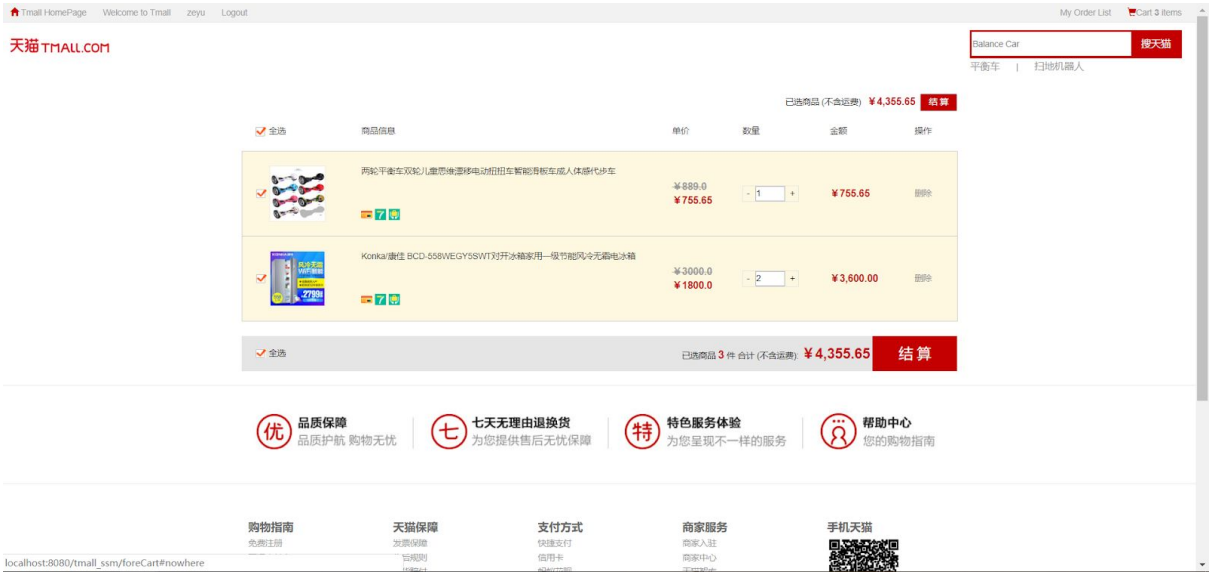
(4) Information of a specific frige in detail:

(5) Login Page:

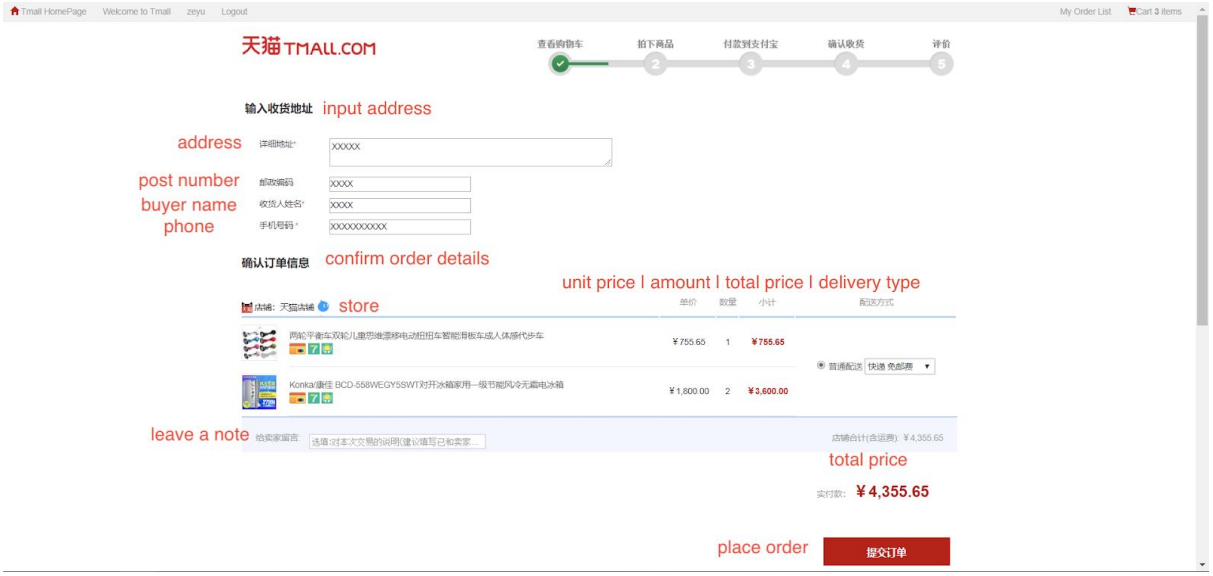


(6) All the items inside the cart:

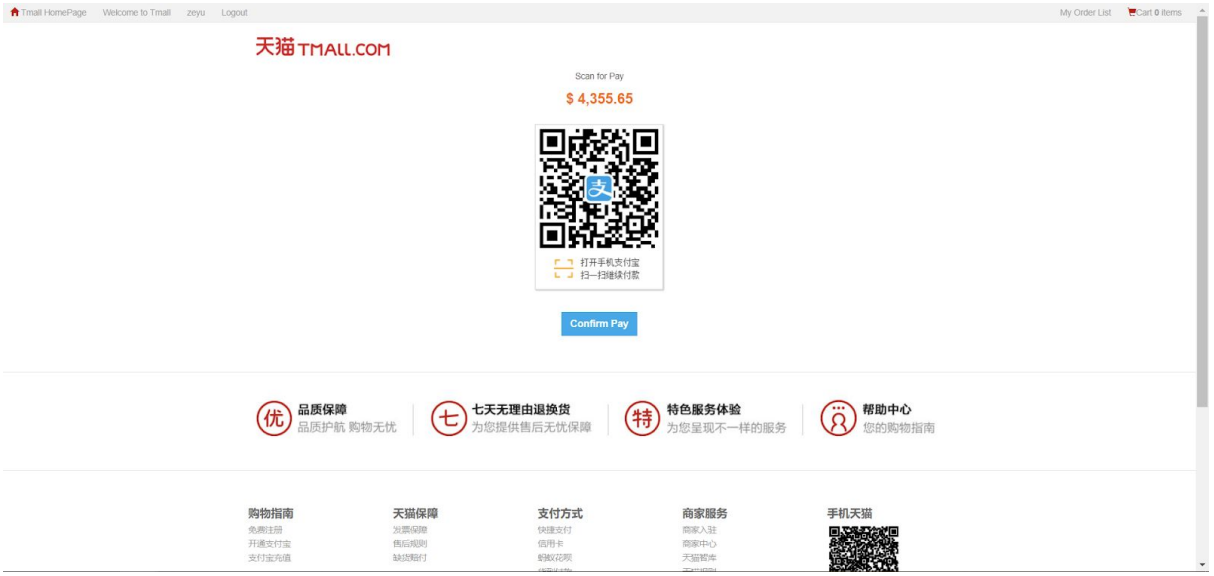


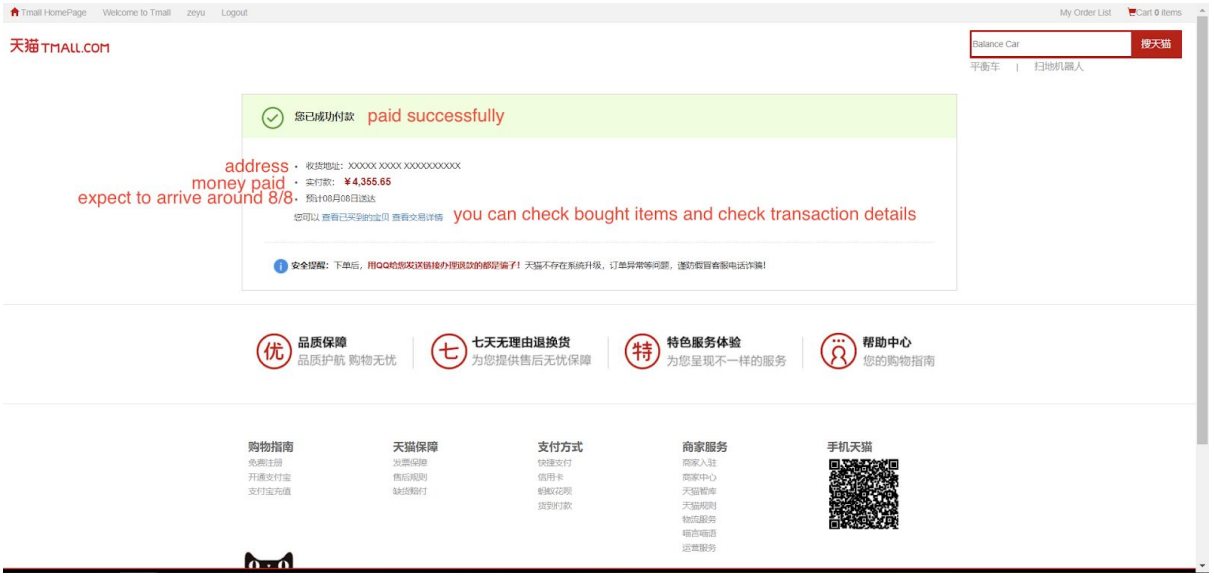


(7) Set delivery address:

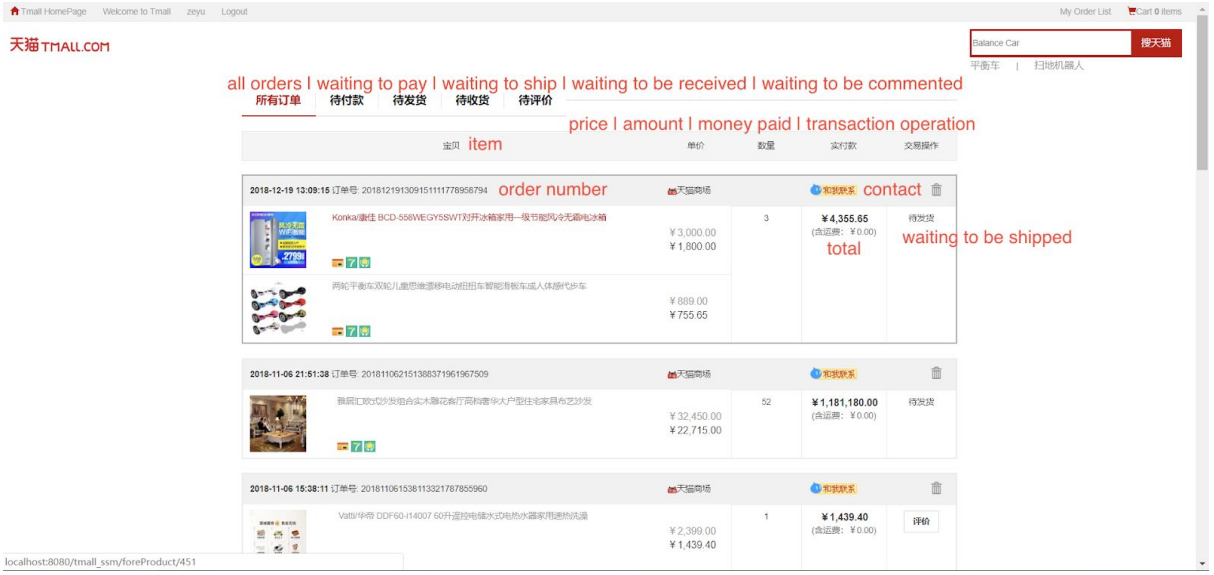


(8) Payment page

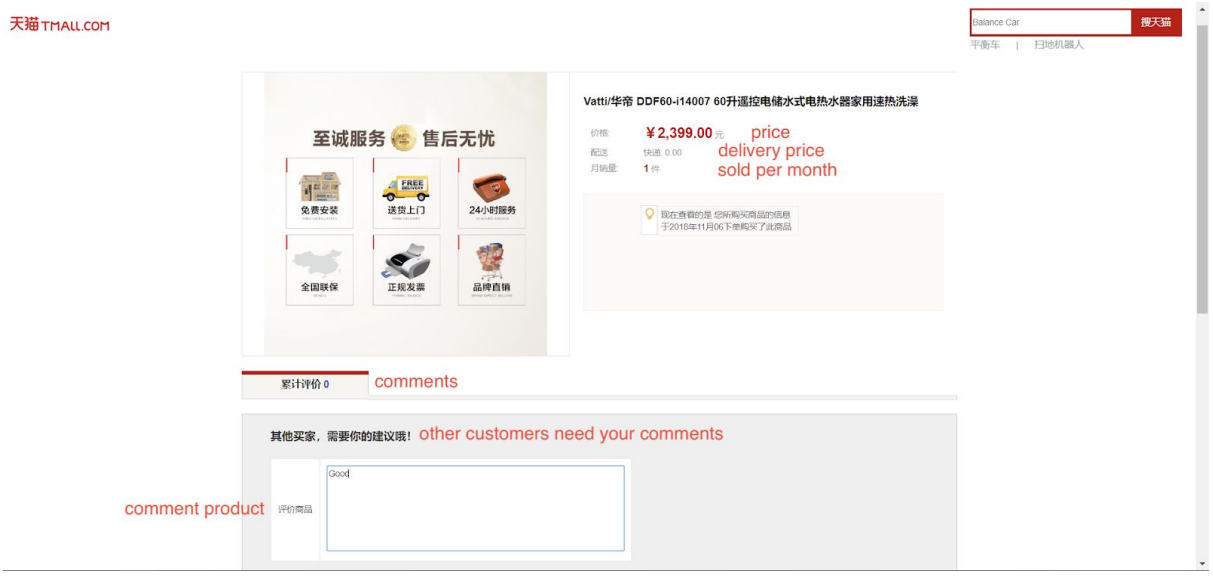




(9) List of all order lists the user have finished, some of which are waiting for delivery:

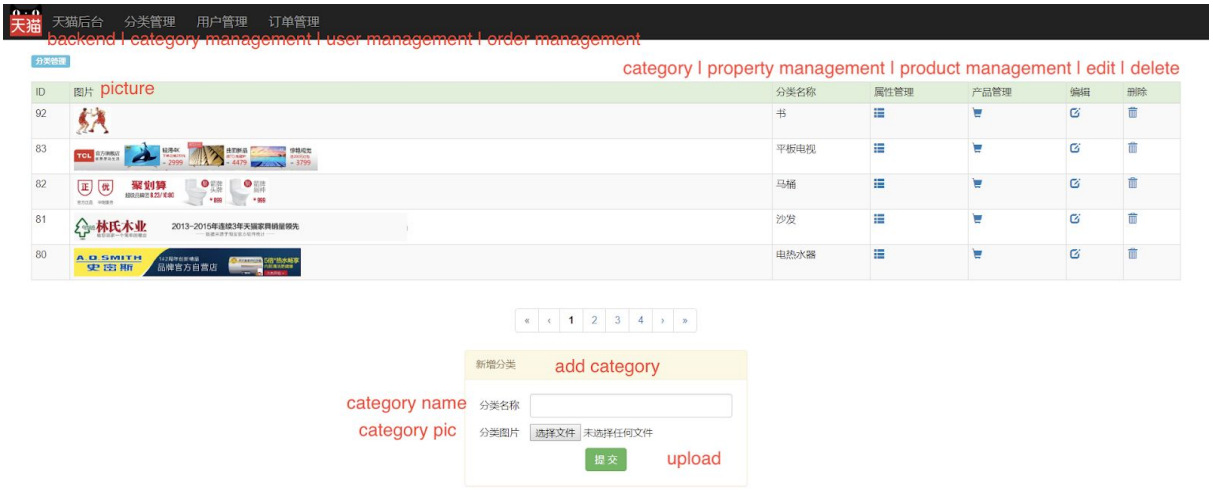


(10) Add Comments after receiving the product:

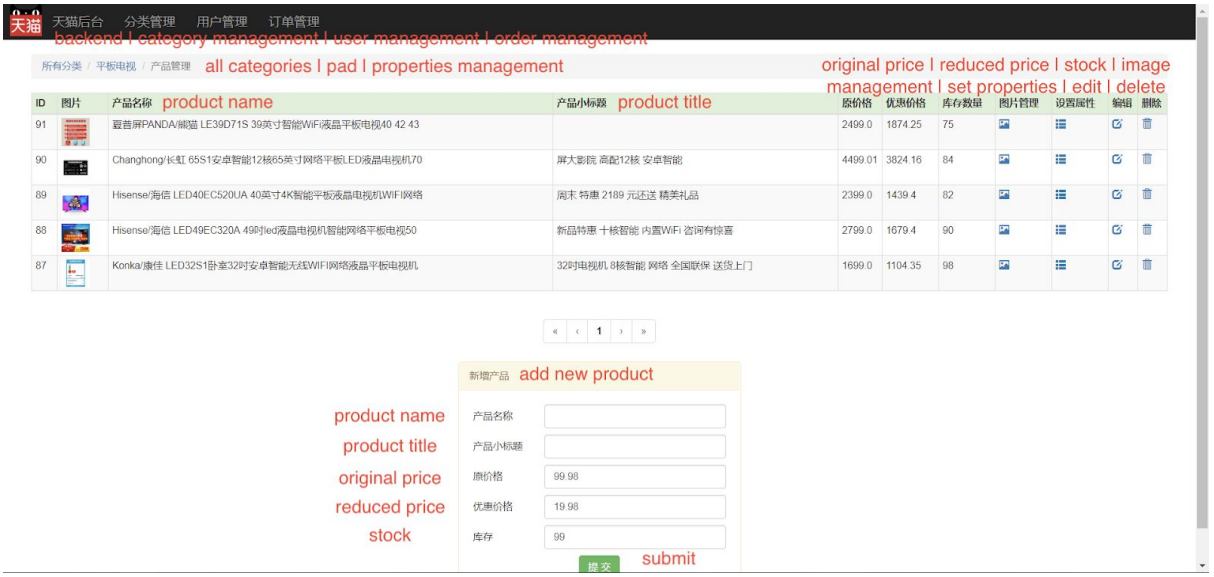


Back-End Display

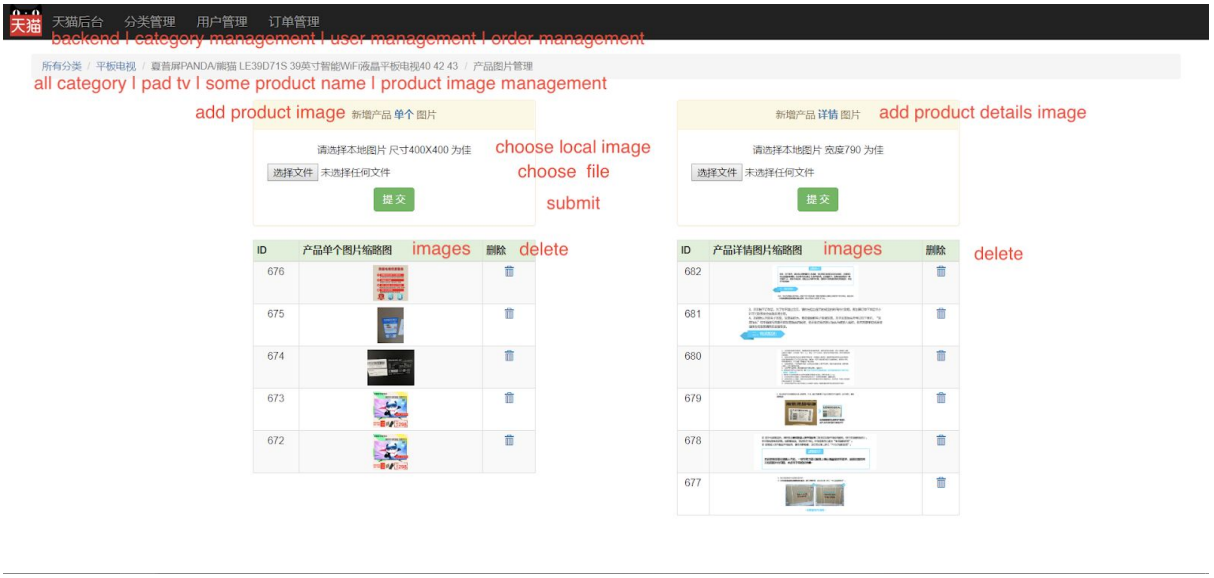
(1) Product Management:



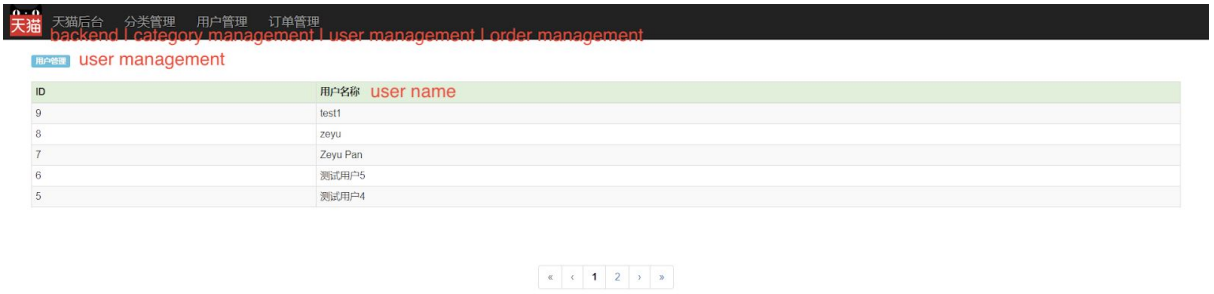
(2) Attributes Management:



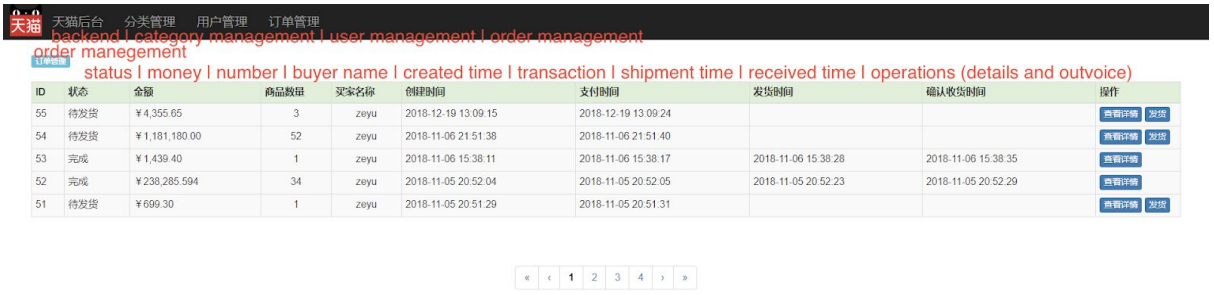
(3) Product Picture Management:



(4) User Management：



（5）Order Management:



7. Full relational table specification of database in SQL DDL

```
(1) CREATE TABLE category (
  id int(11) NOT NULL AUTO_INCREMENT,
  name varchar(255) DEFAULT NULL,
  PRIMARY KEY (id)
) ENGINE=InnoDB AUTO_INCREMENT=84 DEFAULT CHARSET=utf8;

(2) CREATE TABLE user (
```

```
  id int(11) NOT NULL AUTO_INCREMENT,
  name varchar(255) DEFAULT NULL,
  password varchar(255) DEFAULT NULL,
  PRIMARY KEY (id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

(3) CREATE TABLE order_ (
  id int(11) NOT NULL AUTO_INCREMENT,
  orderCode varchar(255) DEFAULT NULL,
  address varchar(255) DEFAULT NULL,
  post varchar(255) DEFAULT NULL,
  receiver varchar(255) DEFAULT NULL,
  mobile varchar(255) DEFAULT NULL,
  userMessage varchar(255) DEFAULT NULL,
  createDate datetime DEFAULT NULL,
  payDate datetime DEFAULT NULL,
  deliveryDate datetime DEFAULT NULL,
  confirmDate datetime DEFAULT NULL,
  uid int(11) DEFAULT NULL,
  status varchar(255) DEFAULT NULL,
  PRIMARY KEY (id),
  CONSTRAINT fk_order_user FOREIGN KEY (uid) REFERENCES user (id)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

(4) CREATE TABLE product (
  id int(11) NOT NULL AUTO_INCREMENT,
  name varchar(255) DEFAULT NULL,
  subTitle varchar(255) DEFAULT NULL,
  originalPrice float DEFAULT NULL,
  promotePrice float DEFAULT NULL,
  stock int(11) DEFAULT NULL,
  cid int(11) DEFAULT NULL,
  createDate datetime DEFAULT NULL,
  PRIMARY KEY (id),
   CONSTRAINT fk_product_category FOREIGN KEY (cid) REFERENCES category (id)
) ENGINE=InnoDB AUTO_INCREMENT=963 DEFAULT CHARSET=utf8;

(5) CREATE TABLE productimage (
  id int(11) NOT NULL AUTO_INCREMENT,
  pid int(11) DEFAULT NULL,
  type varchar(255) DEFAULT NULL,
  PRIMARY KEY (id),
  CONSTRAINT fk_productimage_product FOREIGN KEY (pid) REFERENCES product (id)

) ENGINE=InnoDB AUTO_INCREMENT=10211 DEFAULT CHARSET=utf8;
```

```
(6) CREATE TABLE property (
  id int(11) NOT NULL AUTO_INCREMENT,
  cid int(11) DEFAULT NULL,
  name varchar(255) DEFAULT NULL,
  PRIMARY KEY (id),
  CONSTRAINT fk_property_category FOREIGN KEY (cid) REFERENCES category (id)
) ENGINE=InnoDB AUTO_INCREMENT=258 DEFAULT CHARSET=utf8;

(7) CREATE TABLE propertyvalue (
  id int(11) NOT NULL AUTO_INCREMENT,
  pid int(11) DEFAULT NULL,
  ptid int(11) DEFAULT NULL,
  value varchar(255) DEFAULT NULL,
  PRIMARY KEY (id),
  CONSTRAINT fk_propertyvalue_property FOREIGN KEY (ptid) REFERENCES property
(id)) ENGINE=InnoDB AUTO_INCREMENT=14092 DEFAULT CHARSET=utf8;

(8) CREATE TABLE review (
  id int(11) NOT NULL AUTO_INCREMENT,
  content varchar(4000) DEFAULT NULL,
  uid int(11) DEFAULT NULL,
  pid int(11) DEFAULT NULL,
  createDate datetime DEFAULT NULL,
  PRIMARY KEY (id),
  CONSTRAINT fk_review_product FOREIGN KEY (pid) REFERENCES product (id),
  CONSTRAINT fk_review_user FOREIGN KEY (uid) REFERENCES user (id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

(9) CREATE TABLE orderitem (
  id int(11) NOT NULL AUTO_INCREMENT,
  pid int(11) DEFAULT NULL,
  oid int(11) DEFAULT NULL,
  uid int(11) DEFAULT NULL,
  number int(11) DEFAULT NULL,
  PRIMARY KEY (id),
  CONSTRAINT fk_orderitem_user FOREIGN KEY (uid) REFERENCES user (id),
  CONSTRAINT fk_orderitem_product FOREIGN KEY (pid) REFERENCES product (id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

8. Hard copy of all SQL Code and additional code used for data acquisition and input

See link for full code:

https://drive.google.com/file/d/1HqK6k2fDbuRKItx_WtJauhcLAM1SxtB-/view?usp=sharing

Part of SQL Code:

```
DROP DATABASE IF EXISTS tmall_ssm;
CREATE DATABASE tmall_ssm DEFAULT CHARACTER SET utf8;
USE tmall_ssm;
CREATE TABLE category (
  id int(11) NOT NULL AUTO_INCREMENT,
  name varchar(255) DEFAULT NULL,
  PRIMARY KEY (id)
) ENGINE=InnoDB AUTO_INCREMENT=84 DEFAULT CHARSET=utf8;
INSERT INTO category VALUES (60,'安全座椅');
INSERT INTO category VALUES (64,'太阳镜');
INSERT INTO category VALUES (68,'品牌女装');
INSERT INTO category VALUES (69,'时尚男鞋');
INSERT INTO category VALUES (71,'男士西服');
INSERT INTO category VALUES (72,'男士手拿包 ');
INSERT INTO category VALUES (73,'男表');
INSERT INTO category VALUES (74,'女表');
INSERT INTO category VALUES (75,'空调');
INSERT INTO category VALUES (76,'冰箱');
INSERT INTO category VALUES (77,'原汁机');
INSERT INTO category VALUES (78,'扫地机器人 ');
INSERT INTO category VALUES (79,'平衡车');
INSERT INTO category VALUES (80,'电热水器');
INSERT INTO category VALUES (81,'沙发');
INSERT INTO category VALUES (82,'马桶');
INSERT INTO category VALUES (83,'平板电视');
CREATE TABLE user (
  id int(11) NOT NULL AUTO_INCREMENT,
  name varchar(255) DEFAULT NULL,
  password varchar(255) DEFAULT NULL,
  PRIMARY KEY (id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
CREATE TABLE order_ (
  id int(11) NOT NULL AUTO_INCREMENT,
  orderCode varchar(255) DEFAULT NULL,
  address varchar(255) DEFAULT NULL,
  post varchar(255) DEFAULT NULL,
  receiver varchar(255) DEFAULT NULL,
  mobile varchar(255) DEFAULT NULL,
```

```
  userMessage varchar(255) DEFAULT NULL,
  createDate datetime DEFAULT NULL,
  payDate datetime DEFAULT NULL,
  deliveryDate datetime DEFAULT NULL,
  confirmDate datetime DEFAULT NULL,
  uid int(11) DEFAULT NULL,
  status varchar(255) DEFAULT NULL,
  PRIMARY KEY (id),
  CONSTRAINT fk_order_user FOREIGN KEY (uid) REFERENCES user (id)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE product (
  id int(11) NOT NULL AUTO_INCREMENT,
  name varchar(255) DEFAULT NULL,
  subTitle varchar(255) DEFAULT NULL,
  originalPrice float DEFAULT NULL,
  promotePrice float DEFAULT NULL,
  stock int(11) DEFAULT NULL,
  cid int(11) DEFAULT NULL,
  createDate datetime DEFAULT NULL,
  PRIMARY KEY (id),
   CONSTRAINT fk_product_category FOREIGN KEY (cid) REFERENCES category (id)
) ENGINE=InnoDB AUTO_INCREMENT=963 DEFAULT CHARSET=utf8;
INSERT INTO product VALUES (87,'Konka/康佳 LED32S1卧室32吋安卓智能无线WIFI网络液晶平板电视机','32吋电视机 8核智能 网络 全国联保 送货上门',1699,1104.35,98,83,'2016-08-13 16:43:32');
INSERT INTO product VALUES (88,'Hisense/海信 LED49EC320A 49吋led液晶电视机智能网络平板电视50','新品特惠 十核智能 内置WiFi 咨询有惊喜',2799,1679.4,90,83,'2016-07-28 16:43:38');
INSERT INTO product VALUES (89,'Hisense/海信 LED40EC520UA 40英寸4K智能平板液晶电视机WIFI网络','周末 特惠 2189 元还送 精美礼品',2399,1439.4,82,83,'2016-07-28 16:44:18');
INSERT INTO product VALUES (90,'Changhong/长虹 65S1安卓智能12核65英寸网络平板LED液晶电视机70','屏大影院 高配12核 安卓智能',4499.01,3824.16,84,83,'2016-08-13 16:44:39');
INSERT INTO product VALUES (91,'夏普屏PANDA/熊猫 LE39D71S 39英寸智能WiFi液晶平板电视40 42 43','',2499,1874.25,75,83,'2016-07-18 16:45:05');
INSERT INTO product VALUES (147,'乐洁士智能马桶高品质全自动遥控感应一体式智能坐便器座便器','全国联保 清洗烘干 气泡按摩 座圈加热 断电冲水',8960,8512,62,82,'2016-08-15 00:37:21');
INSERT INTO product VALUES (148,'箭牌马桶/ARROW AB1118脲醛盖板连体喷射虹吸式坐便器座便器正品','领券更优惠 品质脲醛盖 千城送装 送安装配件',1976,1482,65,82,'2016-07-29 00:37:23');
INSERT INTO product VALUES (149,'纳蒂兰卡1066卫浴漩冲虹吸式马桶350坑距坐便器可配智能盖座便器','盖板易拆 送全套配件 500区县 送货入户安装',1670,1503,93,82,'2016-07-20 00:37:25');
INSERT INTO product VALUES (150,'德国DGPOSY超漩式彩色马桶缓降坐便器静音节水抽水马桶连体座便器','纳米自洁釉 漩涡排污 静音节水 净重50kg',2399,1799.25,79,82,'2016-07-31 00:37:27');
INSERT INTO product VALUES (151,'纳蒂兰卡 1082卫浴洁具 钻石切边 坐便器 虹吸式抽水马桶 座便器','高性价比 造型新颖 钻石品质 时尚尊贵 好评如潮',1900,1710,64,82,'2016-08-03 00:37:29');
INSERT INTO product VALUES (202,'新款布艺沙发 可拆洗大小户型客厅简约家居贵妃组合L型转角布沙发','',5198,3638.6,94,81,'2016-08-20 00:45:28');
INSERT INTO product VALUES (203,'凯米蒂亚欧式真皮沙发组合客厅奢华转角皮艺蓝色沙发美式实木家具','匠心独运 欧式美式设计 头层真皮 三包到家',18398,11038.8,27,81,'2016-08-19 00:45:32');
INSERT INTO product VALUES (204,'南台布艺沙发组合可拆洗简约现代客厅整装家具布艺沙发组合大户型','16CM坐垫 16CM坐垫 重要的 事情 说三遍',6576,4274.4,63,81,'2016-08-16 00:45:36');
```

INSERT INTO product VALUES (205,'依然美佳欧式布艺沙发组合可拆洗新款实木雕花大户型奢华别墅家具','高档户型客厅 精湛雕花工艺 实木框架 经典款式',10012,7008.4,51,81,'2016-08-08 00:45:39');
INSERT INTO product VALUES (206,'雅居汇欧式沙发组合实木雕花客厅高档奢华大户型住宅家具布艺沙发','热卖推荐 端庄大气 四色可选 五年质保 品质服务',32450,22715,51,81,'2016-08-02 00:45:44');
INSERT INTO product VALUES (262,'Vatti/华帝 DDF60-i14007 60升遥控电储水式电热水器家用速热洗澡','热水器狂欢 专享特价 送货入户 免费安装',2399,1439.4,80,80,'2016-08-01 00:52:47');
INSERT INTO product VALUES (263,'Haier/海尔 EC6005-T+ 60升电热水器 洗澡淋浴 防电墙 送装同步','3D速热 即用即洗 遥控智能控制 预约洗',1999,1299.35,37,80,'2016-08-10 00:52:49');
INSERT INTO product VALUES (264,'哆啦A梦weber I2-80 威博电热水器80升储水式智能速热洗澡淋浴60','阿里智能 每天限送 20份大礼包 全国联保 包邮',1799,1079.4,79,80,'2016-08-10 00:52:52');
INSERT INTO product VALUES (265,'Vanward/万和 DSCF50-C32双盾电热水器速热联保安装 洗澡50L省电','搪瓷超强内胆 高效节能 迅电速热 双盾保护',999,699.3,17,80,'2016-07-31 00:52:54');
INSERT INTO product VALUES (266,'Haier/海尔 EC8003-I/80升洗澡淋浴/储热电热水器防电墙/送货入户','机控遥控 一级节能 自动关机 触摸智能控制',1799,1169.35,71,80,'2016-07-14 00:52:56');
INSERT INTO product VALUES (317,'乐行天下R1N智能电动平衡车双轮成人思维体感车两轮儿童代步车','12期免息 智能显示屏 APP遥控 双轮平衡',15960,11970,15,79,'2016-07-09 00:59:24');
INSERT INTO product VALUES (318,'乐途智能电动悬浮滑板车太空轮平行车蓝牙自平衡独轮代步车','电动平衡 独轮滑板 蓝牙播放 新潮代步车',3885,3690.75,70,79,'2016-07-08 00:59:25');
INSERT INTO product VALUES (319,'瑞士米高micro g-bike+ chopper儿童平衡车二轮自行车','',899,809.1,13,79,'2016-07-12 00:59:27');
INSERT INTO product VALUES (320,'力达康成人电动滑板车折叠锂电池迷你型可折叠两轮代步车','可折叠 退货赔运费 续航50 公里更轻更远',4160,2912,82,79,'2016-08-19 00:59:28');
INSERT INTO product VALUES (321,'两轮平衡车双轮儿童思维漂移电动扭扭车智能滑板车成人体感代步车','',889,755.65,64,79,'2016-08-09 00:59:29');
INSERT INTO product VALUES (373,'洁霸BF522多功能洗地机刷地机酒店家用地毯清洗机 擦地机抛光机','纯铜电机 厂家直销 售后无忧',1900,1235,39,78,'2016-07-21 01:06:32');
INSERT INTO product VALUES (374,'Proscenic Swan天鹅扫地机器人 智能家用擦地机拖地全自动吸尘器','非常夏日 直降300元 WIFI智控 卷吸互换',6580,4935,65,78,'2016-07-08 01:06:34');
INSERT INTO product VALUES (375,'海尔拖地机器人家用全自动擦地机湿拖扫洗地电动智能懒人无线超薄','仿人跪式 干湿两拖 自动充电 精准导航 规划清洁',3999,2999.25,30,78,'2016-07-21 01:06:37');
INSERT INTO product VALUES (376,'莱克扫地吸尘器 家用超薄拖地机器人WR10智能静音全自动','新品上市 3000pa 大吸力 无惧毛发缠绕',1299,974.25,11,78,'2016-07-18 01:06:39');
INSERT INTO product VALUES (377,'公众智能扫地机器人家用全自动电动清洁地毯擦拖地一体机吸尘器','扫拖一体 纤薄高效锂电 限时疯抢 不缠毛发',2499,2124.15,88,78,'2016-07-19 01:06:41');
INSERT INTO product VALUES (429,'Philips/飞利浦榨汁机原汁家用HR1884多功能水果冰淇淋机慢汁机','',2899,1884.35,40,77,'2016-07-14 01:14:03');
INSERT INTO product VALUES (430,'Whirlpool/惠而浦WJU-VP152B原汁机多功能家用大口径料理榨果汁机','整投整榨 原汁无氧化 纯铜电机 三挡压榨',1099,824.25,22,77,'2016-08-24 01:14:05');
INSERT INTO product VALUES (431,'Ranbem/瑞本611原汁机大口径低慢速榨汁家用水果汁迷你豆浆机电动','新款上市 96mm口径 45慢磨转速 出汁率高',2680,2278,35,77,'2016-07-21 01:14:08');
INSERT INTO product VALUES (432,'AUX/奥克斯 AUX-Y3802原汁机低速 慢磨水果榨汁机果汁机','',999,799.2,50,77,'2016-07-15 01:14:12');
INSERT INTO product VALUES (433,'惠而浦WJU-VS208J原汁机家用慢榨多功能榨汁机果汁机豆浆机炸汁机','大功率慢榨 浓度调节 噪音小 易清洗 快捷清洗',1499,1124.25,35,77,'2016-08-12 01:14:14');
INSERT INTO product VALUES (449,'MeiLing/美菱 BCD-560WUCX对开门电冰箱家用冰箱WIFI智能风冷无霜','家用冰箱 WIFI智能 风冷无霜',3399,3059.1,54,76,'2016-08-14 01:19:59');
INSERT INTO product VALUES (450,'FRESTECH/新飞 BCD-286WDKSM阿里云智能wifi风冷无霜智控两门冰箱','阿里云智能 远程WIFI 智控 风冷无霜',3599,2879.2,68,76,'2016-08-05 01:20:00');
INSERT INTO product VALUES (451,'Konka/康佳 BCD-558WEGY5SWT对开冰箱家用一级节能风冷无霜电冰箱','阿里云智能 风冷无霜 大容量 一级能效节能',3000,1800,74,76,'2016-07-24 01:20:03');

INSERT INTO product VALUES (452,'【阿里智能】Galanz/格兰仕 BCD-251WTHG(T) 三门电冰箱251升风冷','风冷无霜 一级能效 智能控温 菱格面板 立体风',3299,2474.25,63,76,'2016-07-27 01:20:05');
INSERT INTO product VALUES (453,'Haier/海尔 BCD-618WDVGU1海尔变频对开门风冷智能控温双门电冰箱','手机智能控制 玫瑰金外观 杀菌净味系统 变频',6999,5949.15,22,76,'2016-08-11 01:20:07');
INSERT INTO product VALUES (509,'AUX/奥克斯 KFR-26GW/BpTYC19+1大1匹高端智能云变频冷暖挂机空调','30秒速冷 一级能效 ECO节能 阿里智能 除湿',3799,2659.3,15,75,'2016-08-01 01:27:18');
INSERT INTO product VALUES (510,'Kelon/科龙 KFR-26GW/EFQRA2(1N20) 大1匹二级变频智能冷暖挂机','二级变频 阿里智能 高效除甲醛 快速制冷暖',2599,1559.4,78,75,'2016-08-08 01:27:20');
INSERT INTO product VALUES (511,'TCL KFRd-35GW/EP13 大1.5匹冷暖电辅空调1.5P挂机定频壁挂式包邮','快速冷暖 四重静音 每日前5名 好礼2选1',2399,1679.3,64,75,'2016-07-21 01:27:22');
INSERT INTO product VALUES (512,'Panasonic/松下 KFR-36GW/BpSJ1S大1.5匹直流变频冷暖挂式空调','原装变频压缩 机静音除雾霾 送货入户 0元分期',3999,3599.1,59,75,'2016-07-16 01:27:25');
INSERT INTO product VALUES (513,'Panasonic/松下 KFR-36GW/SH2-1大1.5匹二级能效静音空调挂机','原装压缩机 品质保证 二级能效',3399,3059.1,71,75,'2016-07-12 01:27:27');
INSERT INTO product VALUES (565,'正品雷克斯韩版简约时尚潮流真皮石英情侣表女士手表女表皮带防水','时尚之选 做精致女人 精美包装 支持货到付款',1280,896,82,74,'2016-08-22 01:34:54');
INSERT INTO product VALUES (566,'正港手表女四叶草防水手链表简约潮流学生女表韩版时尚石英表正品','热销万只 4.9分好评 代写贺卡 质保一年',143.71,114.968,23,74,'2016-08-25 01:34:57');
INSERT INTO product VALUES (567,'罗宾 钢带女表 女士腕表 水钻夜光防水石英表 潮流时尚复古手表女','正品 水钻刻度 钢带女表 买一送三 赠送运费险',990,792,54,74,'2016-07-11 01:34:59');
INSERT INTO product VALUES (568,'浪铂正品长方形手表女表皮带时尚潮流石英表防水真皮学生休闲腕表','今日拍下 自动减30 送真皮表带 石榴石手链',998,598.8,81,74,'2016-07-26 01:35:02');
INSERT INTO product VALUES (569,'宾格手表正品女式精钢腕表大表盘全自动机械表防水情侣表女表真皮','陈小春代言 宾格正品 情侣手表 同款还有男表',488,341.6,16,74,'2016-07-19 01:35:05');
INSERT INTO product VALUES (620,'依伦新款超薄手表 男表真皮带休闲时尚男士腕表潮流防水石英表','简约两针 深度防水 进口机芯 送真皮表带 运费险',1280,1024,64,73,'2016-08-11 01:42:48');
INSERT INTO product VALUES (621,'【入门】西铁城CITIZEN新品超值体验机械手表男表NH8350-59AB/EB','全国联保 3期0利息 大气简约 新品',1700,1445,60,73,'2016-08-13 01:42:51');
INSERT INTO product VALUES (622,'正品手表男精钢商务休闲男士手表石英表时尚潮流夜光超薄男表防水','机构质检 防水夜光 赠送皮表带 特价抢购',699,489.3,34,73,'2016-08-25 01:42:54');
INSERT INTO product VALUES (623,'正品名仕爵手表男机械表夜光精钢防水大表盘 全自动时尚潮流男表','终身保修 正品 全自动 机械表 镂空机械表',1900,1140,85,73,'2016-08-25 01:42:57');
INSERT INTO product VALUES (624,'飞克手表 男表皮带手表 防水商务休闲男士手表 腕表 石英表手表','正品进口机芯 蓝宝石镜面 指针星期显示',1999,1199.4,82,73,'2016-08-05 01:43:00');
INSERT INTO product VALUES (673,'劳迪莱斯男士手包休闲手拿包牛皮大容量钱包男包软面小包包手抓包','牛皮手包 耐磨耐用 多卡位设计',185,157.25,24,72,'2016-07-21 01:50:46');
INSERT INTO product VALUES (674,'英伦邦纹男士手包牛皮大容量真皮手拿包手抓包双拉链商务正品软皮','意大利牛皮 纹路自然漂亮 更耐用更尊贵 手感棒',588,411.6,42,72,'2016-08-23 01:50:48');
INSERT INTO product VALUES (675,'唯美诺新款男士手包男包真皮大容量小羊皮手拿包信封包软皮夹包潮','优质羊皮 大容量 品牌做工 超软手感 终身包保修',498,448.2,84,72,'2016-07-30 01:50:52');
INSERT INTO product VALUES (676,'宾度 男士手包真皮大容量手拿包牛皮个性潮男包手抓包软皮信封包','专注品质领卷 顺丰包邮 头层牛皮 大容量 软皮',568,511.2,79,72,'2016-08-04 01:50:54');
INSERT INTO product VALUES (677,'MAXFEEL休闲男士手包真皮手拿包大容量信封包手抓包夹包软韩版潮','无理由退换货 赠退货运费险',888,799.2,98,72,'2016-07-21 01:50:58');
INSERT INTO product VALUES (729,'【新】罗蒙羊毛西服套装男修身商务职业装2016年秋季新款#1S51126','',2199,1759.2,96,71,'2016-08-04 01:57:55');
INSERT INTO product VALUES (730,'男士西服套装修身英伦竖条纹休闲西装男套装三件套新郎结婚礼服春','今赠马甲衬衫 精致竖条纹 奢华品质 精细做工',729,437.4,30,71,'2016-07-20 01:57:57');
INSERT INTO product VALUES (731,'西服套装男夏季新郎结婚礼服三件套伴郎服薄款西服大码西装男套装','亮眼帅气 优雅休闲 格调气质 时尚格子 结婚首选',1988,1491,10,71,'2016-08-10 01:57:59');

```
INSERT INTO product VALUES (732,'三件套Actionmates新款韩版修身休闲格子西服套装男士小西装礼服','韩国进口面料 韩版立体裁剪 版型好 送领结领带',799,479.4,56,71,'2016-07-16 01:58:02');
INSERT INTO product VALUES (733,'Action mates韩版修身男士西服套装发型师休闲小西装结婚礼服潮','包邮申通 韩版修身 版型好 不容错过',699,419.4,79,71,'2016-07-07 01:58:04');
INSERT INTO product VALUES (784,'DOGEXI透气真皮男鞋子雕花男士休闲鞋反绒皮英伦低帮板鞋男潮鞋秋','牛皮鞋面 猪皮内里 标准运动鞋码',289,202.3,21,69,'2016-08-13 02:06:10');
INSERT INTO product VALUES (785,'花花公子男鞋夏季小白鞋男士休闲鞋白鞋板鞋白色平底百搭鞋子男潮','天猫正品 百搭小白鞋 顺丰包邮',330,313.5,44,69,'2016-08-20 02:06:14');
INSERT INTO product VALUES (786,'花花公子皮鞋男士商务休闲男鞋夏季英伦真皮系带休闲鞋男板鞋子','高端大气 商务休闲 时尚潮流 全国包邮 赠运费险',368,294.4,23,69,'2016-08-11 02:06:17');
INSERT INTO product VALUES (787,'酷队秋季运动鞋情侣鞋跑步鞋潮男鞋子透气休闲鞋飞织低帮潮鞋系带','',328,229.6,73,69,'2016-08-04 02:06:20');
INSERT INTO product VALUES (788,'木林森夏季男鞋休闲鞋男英伦真皮男士休闲皮鞋圆头系带青年潮鞋子','头层牛皮 时尚休闲 韩版潮流 型男必备',326,293.4,96,69,'2016-08-17 02:06:24');
INSERT INTO product VALUES (844,'拉夏贝尔puella普埃拉2016休闲修身五分袖一字领收腰套装20007986','',369,350.55,71,68,'2016-08-14 02:13:28');
INSERT INTO product VALUES (845,'拉夏贝尔 7m莫丽菲尔2016秋新款条纹五分袖连衣裙套装女70006522','8月3日 新品',399,299.25,79,68,'2016-07-10 02:13:30');
INSERT INTO product VALUES (846,'女巫2016欧洲站夏季新品欧美时尚绣花衬衣上衣九分裤两件套装高端','女巫高端定制 欧美时尚 两件套',928,835.2,29,68,'2016-08-10 02:13:32');
INSERT INTO product VALUES (847,'妖精的口袋P360度漂浮秋装欧美半身裙短款外套时尚套装女两件套','口袋特殊材质 拼接 品质扣子 水洗色 套装',399,319.2,73,68,'2016-08-18 02:13:35');
INSERT INTO product VALUES (848,'妖精的口袋P几枚礼物秋装甜美连帽短外套印花休闲裤时尚套装女','帽里撞色 植绒抽绳装饰 质感吊钟 图案印花 套装',339,322.05,87,68,'2016-08-15 02:13:37');
INSERT INTO product VALUES (899,'新款男士偏光太阳镜日夜两用墨镜潮运动开车专用驾驶偏光镜夜视镜','时尚新款 铝镁纤维镜架 科技动感',580,551,91,64,'2016-08-06 02:21:07');
INSERT INTO product VALUES (900,'变色眼镜男女款半框太阳镜大框潮流防辐射防紫外防蓝光近视镜护目','',200,170,77,64,'2016-07-27 02:21:10');
INSERT INTO product VALUES (901,'帕莎Prsr太阳镜女偏光镜潮范冰冰同款女士眼镜可配近视墨镜T60017','镜片 防紫外线 超轻材质 记忆材质',780,624,86,64,'2016-08-16 02:21:13');
INSERT INTO product VALUES (902,'陌森太阳眼镜男女2016偏光定制驾驶近视墨镜有带度数的莫森MS8006','加拍镜片可配 近视： 原装无度数！ 顺丰包邮',798,518.7,45,64,'2016-08-03 02:21:17');
INSERT INTO product VALUES (903,'好先生同款墨镜孙红雷偏光男士太阳镜韩明星女款圆脸方框金属近视','偏光 圆脸 方框',150,97.5,84,64,'2016-08-12 02:21:20');
INSERT INTO product VALUES (958,'惠尔顿儿童安全座椅isofix硬接口汽车用安全坐椅9个月-12岁酷睿宝','双接口固定 夏克立推荐 好评如潮',2848,1993.6,52,60,'2016-07-25 02:29:07');
INSERT INTO product VALUES (959,'好孩子汽车儿童安全座椅goodbaby9个月－12岁宝宝座椅cs668侧碰王','开团3秒半价 475元 提前购买 咨询客服',1999,1199.4,89,60,'2016-07-17 02:29:10');
INSERT INTO product VALUES (960,'REEBABY儿童安全座椅9个月-12岁宝宝婴儿汽车用坐椅车载3C认证','睿睿熊定制款 合金钢骨架 全国包邮',1280,1216,71,60,'2016-07-26 02:29:13');
INSERT INTO product VALUES (961,'REEBABY汽车儿童安全座椅ISOFIX 0-4-6-12岁婴儿宝宝新生儿可躺','165度超大 躺角 0-12岁 正反双向 安装',1680,1344,59,60,'2016-08-21 02:29:16');
INSERT INTO product VALUES (962,'新生儿婴儿提篮式安全座椅汽车用车载儿童安全坐椅宝宝摇篮便携式','法国畅销品牌 ITW卡扣 环保针织面料',980,882,16,60,'2016-08-09 02:29:19');
CREATE TABLE productimage (
  id int(11) NOT NULL AUTO_INCREMENT,
  pid int(11) DEFAULT NULL,
  type varchar(255) DEFAULT NULL,
  PRIMARY KEY (id),
  CONSTRAINT fk_productimage_product FOREIGN KEY (pid) REFERENCES product (id)
```

```
) ENGINE=InnoDB AUTO_INCREMENT=10211 DEFAULT CHARSET=utf8;
```

Appendix

DataBase Final Project Proposal

1. Shanshan Yang, Zeyu Pan, Chen Wang (With prior approval teams of 3)
2. Our database contains the data of a shopping website. It consists of the data of products, properties, categories, users, orders and order items so on and so forth.
3. (1) Compute the number of products for each category.
   (2) List all the products in television category with prices less than 1000$.
   (3) Compute the average prices of products in each product categories.
   (4) Get the most viewed products in the past six months.
   (5) Calculate the number of clothes in each different size.
   (6) Return all the products with name containing a specific search word.
   (7) Sort a list of products in the same category based on their prices.
   (8) Sort a list of products in the same category based on their viewed number.
   (9) List all the orders the user have submitted.
   (10) Calculate the sum of saving money for a specific user in the past one year.
   (11) Get the most products for one user having bought.
   (12) From all the product the user having bought, get the product with highest price.
   (13) Return the number of products in each of orders.
   (14) List the total number of products of one category.
   (15) List the user with sailing address in the same state.

4. Design and show a relational data model

```
# 1
CREATE DATABASE tmall_ssm DEFAULT CHARACTER SET utf8;
USE tmall_ssm;

# 2
CREATE TABLE category (
 id int(11) NOT NULL AUTO_INCREMENT,
 name varchar(255) DEFAULT NULL,
 PRIMARY KEY (id)
) ENGINE=InnoDB AUTO_INCREMENT=84 DEFAULT CHARSET=utf8;
INSERT INTO category VALUES (60,'安全座椅');
INSERT INTO category VALUES (64,'太阳镜');

# 3
```

```
CREATE TABLE user (
 id int(11) NOT NULL AUTO_INCREMENT,
 name varchar(255) DEFAULT NULL,
 password varchar(255) DEFAULT NULL,
 PRIMARY KEY (id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

# 4
CREATE TABLE order_ (
 id int(11) NOT NULL AUTO_INCREMENT,
 orderCode varchar(255) DEFAULT NULL,
 address varchar(255) DEFAULT NULL,
 post varchar(255) DEFAULT NULL,
 receiver varchar(255) DEFAULT NULL,
 mobile varchar(255) DEFAULT NULL,
 userMessage varchar(255) DEFAULT NULL,
 createDate datetime DEFAULT NULL,
 payDate datetime DEFAULT NULL,
 deliveryDate datetime DEFAULT NULL,
 confirmDate datetime DEFAULT NULL,
 uid int(11) DEFAULT NULL,
 status varchar(255) DEFAULT NULL,
 PRIMARY KEY (id),
 CONSTRAINT fk_order_user FOREIGN KEY (uid) REFERENCES user (id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

# 5
CREATE TABLE product (
 id int(11) NOT NULL AUTO_INCREMENT,
 name varchar(255) DEFAULT NULL,
 subTitle varchar(255) DEFAULT NULL,
 originalPrice float DEFAULT NULL,
 promotePrice float DEFAULT NULL,
 stock int(11) DEFAULT NULL,
 cid int(11) DEFAULT NULL,
 createDate datetime DEFAULT NULL,
 PRIMARY KEY (id),
  CONSTRAINT fk_product_category FOREIGN KEY (cid) REFERENCES category
(id)
) ENGINE=InnoDB AUTO_INCREMENT=963 DEFAULT CHARSET=utf8;
INSERT INTO product VALUES (87,'Konka/康佳 LED32S1卧室32吋安卓智能无线
WIFI网络液晶平板电视机','32吋电视机 8核智能 网络 全国联保 送货上门
',1699,1104.35,98,83,'2016-08-13 16:43:32');
INSERT INTO product VALUES (88,'Hisense/海信 LED49EC320A 49吋led液晶电视
机智能网络平板电视50','新品特惠 十核智能 内置WiFi 咨询有惊喜
',2799,1679.4,90,83,'2016-07-28 16:43:38');
```

```
# 6
CREATE TABLE productimage (
 id int(11) NOT NULL AUTO_INCREMENT,
 pid int(11) DEFAULT NULL,
 type varchar(255) DEFAULT NULL,
 PRIMARY KEY (id),
 CONSTRAINT fk_productimage_product FOREIGN KEY (pid) REFERENCES
product (id)

) ENGINE=InnoDB AUTO_INCREMENT=10211 DEFAULT CHARSET=utf8;

INSERT INTO productimage VALUES (629,87,'type_single');
INSERT INTO productimage VALUES (630,87,'type_single');

# 7
CREATE TABLE property (
 id int(11) NOT NULL AUTO_INCREMENT,
 cid int(11) DEFAULT NULL,
 name varchar(255) DEFAULT NULL,
 PRIMARY KEY (id),
 CONSTRAINT fk_property_category FOREIGN KEY (cid) REFERENCES category
(id)
) ENGINE=InnoDB AUTO_INCREMENT=258 DEFAULT CHARSET=utf8;
INSERT INTO property VALUES (1,83,'品牌');
INSERT INTO property VALUES (2,83,'证书状态');

# 8
CREATE TABLE propertyvalue (
 id int(11) NOT NULL AUTO_INCREMENT,
 pid int(11) DEFAULT NULL,
 ptid int(11) DEFAULT NULL,
 value varchar(255) DEFAULT NULL,
 PRIMARY KEY (id),
 CONSTRAINT fk_propertyvalue_property FOREIGN KEY (ptid) REFERENCES
property (id)

) ENGINE=InnoDB AUTO_INCREMENT=14092 DEFAULT CHARSET=utf8;

INSERT INTO propertyvalue VALUES (716,87,13,'VIDAA');
INSERT INTO propertyvalue VALUES (717,87,12,'无');

# 9
CREATE TABLE review (
 id int(11) NOT NULL AUTO_INCREMENT,
 content varchar(4000) DEFAULT NULL,
```

```
 uid int(11) DEFAULT NULL,
 pid int(11) DEFAULT NULL,
 createDate datetime DEFAULT NULL,
 PRIMARY KEY (id),
 CONSTRAINT fk_review_product FOREIGN KEY (pid) REFERENCES product (id),
 CONSTRAINT fk_review_user FOREIGN KEY (uid) REFERENCES user (id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
# 10
CREATE TABLE orderitem (
 id int(11) NOT NULL AUTO_INCREMENT,
 pid int(11) DEFAULT NULL,
 oid int(11) DEFAULT NULL,
 uid int(11) DEFAULT NULL,
 number int(11) DEFAULT NULL,
 PRIMARY KEY (id),
 CONSTRAINT fk_orderitem_user FOREIGN KEY (uid) REFERENCES user (id),
 CONSTRAINT fk_orderitem_product FOREIGN KEY (pid) REFERENCES product
(id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

5.

```
#1
SELECT Products.name, avg(price) AS price_avg
FROM Products NATURAL JOIN OrderItems NATURAL JOIN Orders
          NATURAL JOIN
          (SELECT DISTINCT Users.userId
           FROM  Users NATURAL JOIN Orders INNER JOIN OrderItems ON
OrderItems.oid = Orders.id
           GROUP BY Users.UserId )
GROUP BY Product.name;


#2
SELECT Prodcut.name, Category.cid, Orders.oid
FROM (SELECT Product.name,count(*) as num, avg(prices) AS avg
     FROM Product NATURAL JOIN OrderItems NATURAL JOIN Orders NATURAL
JOIN Users
     GROUP BY Product.name) AS L
    NATURAL JOIN
    (SELECT Product.name, Category.cid, Orders.oid
     FROM Product NATURAL JOIN Orders NATURAL JOIN Categories);



#3
SELECT Users,name, Users.address
```

```
FROM (SELECT Users.name, Users.address, Orders.totalPrice
      FROM Users NATURAL JOIN Orders
      WHERE Orders.totalPrice > 1000)
     NATURAL JOIN
     (SELECT User.name, Users.address, Orders.totalPrice
      FROM Users NATURAL JOIN Orders
      WHERE Orders.totalPrice < 2000);

SELECT Product.name, Users.Name
FROM Product NATURAL JOIN OrderItems NATURAL JOIN Orders INNER JOIN
Users on users.uid = orders.uid
WHERE sum(Orders.prices) / sum(Orders.count) >= ALL (SELECT
sum(Orders.prices) / sum(Orders.count) AS avg
                                    FROM Orders INNER JOIN Users ON users.uid =
orders.uid
                                    GROUP BY orders.oid)
```

6. Plan to load the databases:

   We go through the website of Tmall online shopping platform and crawler data form this website https://www.tmall.com/ by parsing the html tags and store them into xml format files. Then we design relations and tables according to these data. The issues that we expected to encounter is that the raw data that we crawled need to reformat and reduce the redundancy to better fit our database.

7. The reslut that we plan to generate:

   For our databases final project, we want to build an online shopping website, which contains some similar functions as Amazon or eBay. More specific, it is a Java web project which imitates the famous Chinese online shopping website Tmall. It will support all the major functions of any shopping websites, such as user register, login, logout, list products by category, search by keyword, check product in detail and it also implements the entire process of shopping, like add to cart, checkout, generate orders, keep track of orders, review products so on and so forth. For implementation, we want to use Vue.js, JQuery and AJAX.

8. The specialized / advanced topics that we plan to focus on in the database is:
   - Security - Shiro (for payment)
   - Object-oriented database design / implementation
   - Advanced SQL topics - myBatis, JDBC
   - data mining