测试点统计：

# 微信消息界面怎么测？

界面测试：

头像显示，消息显示是否正常

功能测试:

发送功能是否正常，接收功能是否正常，语音，视频聊天是否正常，支持的图片，文字，文件格式

兼容性测试：

不同输入法，不同手机型号，不同手机系统，不同电脑型号与版本

弱网测试：

在网络条件不好或者无网络的情况下各类功能是否正常

场景组合测试：

各种条件组合交叉测试一下

# 微信点赞怎么测：

功能测试

1.给某个好友点赞，点赞数+1，点赞栏显示具体点赞人的名字 ，该用户手动点赞回馈  
2.点完赞后，共同好友在点赞区能看到该人是不是点赞  
了，非共同好友看不到  
3.两个头像一样的人点赞，能否正确显示  
4.点完赞后，在点击点变成点赞取消  
5.取消点赞--不通知用户  
6.点赞后，通知用户，取消，在点赞，此时不通知用户  
7.多个用户同时对其点赞，点赞数正常  
8.最多能点多少个赞--边界值测试  
9.可以从点击点赞区头像，进入相应人的主页查看  
10.点赞是否按照时间顺序排序  
11.点赞后是否能够正常评论

app端测试

1.弱网情况下，点赞能否实时更新  
2.点赞时，有电话进来或者断网，能否显示点赞情况

性能测试

1大量用户并发点赞时，该接口的响应时间，最大承受的qps  
2.大量用户并发点赞时，此时界面进行点赞，点赞功能是否正常

兼容性测试

1. 不同手机型号，点赞功能，显示功能是否正常

# 注册用例测试点：

功能测试用例：

输入符合规定的账号密码进行注册，看注册是否成功，验证注册功能可用；

输入已注册过的账号再次注册，是否提示该账号已经注册；

输入不符规则的账号进行注册，是否提示账号不可用；

输入特殊字符进行注册；

不输入任何内容点击注册；

输入空格点击注册；

密码过于简单是否提示修改

错误的验证码：空，空格，全角字符，半角字符，表情符号；

兼容性测试用例：

（1）不同浏览器下，注册页面的显示以及功能正确性；（2）相同浏览器的不同版本下，注册页面的显示以及功能正确性；

# 登录测试点：

基本功能测试点：

输入正确的用户名和密码是否登录成功，验证登录功能可用；

输入错误的用户名密码登录失败，提示账号密码不正确；

用户名正确，密码错误，是否提示输入密码错误？

如果用户未注册，提示请先注册；

已经注销的用户登录失败；

登录功能是否需要输入验证码？

验证码有效时间？验证码输入错误，输入过期的验证能否登录成功？，验证码是否容易识别？，验证码是否可以点击刷新？

页面测试：

登录页面显示是否正常？，文字和图片能否正常显示，相应的提示信息是否正确，按钮的设置和排列是否正常，页面是否简洁壮观等。

兼容性测试：

不同浏览器

不同操作系统

不同分辨率下界面是否正常

安全测试：

错误登陆的次数限制，考虑是否支持多用户在同一机器上登录；

考虑一用户在多台机器上登录

# 朋友圈测试点：

一、功能测试

朋友圈发送

只发送文本（长按相机图标进入）

字符长度限制，纯数字、纯英文、纯字符、纯中文、纯表情（手机自带表情、微信表情）、数字+英文+字符+中文+表情、包含URL链接、文本是否支持复制粘贴。

发送文本超出字符限制，为空验证

只发送图片（点击相机图标）

图片数量验证，图片格式验证，图片大小限制，超出限制后的处理方式。

为空验证。

只发送视频（点击相机图标）

本地相册选择、拍摄，视频时长验证，视频个数验证，视频大小验证，视频为空验证

交叉验证

发送文本+图片，发送文本+视频，发送图片+视频

位置验证（定位）

不显示位置，显示位置

谁可以看（朋友圈权限设置）

设置公开，所有朋友可见

设置私密（仅自己可见）：自己查看朋友圈可见，好友查看朋友圈不可见。

设置部分可见（部分朋友可见）：选择部分好友可见；不选择的好友不可见；是否有人数限制； 设置不给谁看（选中的朋友不可见）：不选择的朋友可见；选中的朋友不可见；是否有人数限制；

提醒谁看

提醒单人、提醒多人，是否有人数上限；被提醒的朋友，收到提醒消息；未被提醒，未有消息提醒。

朋友圈当天发送次数是否有上限限制

朋友圈显示设置

发表评论（评论字数限制）

纯数字，纯英文，纯字符，纯中文，纯表情（手机自带表情、微信表情），数字+英文+字符+中文+表情，文本是否支持复制粘贴

评论超过字数限制，为空验证，发表评论后删除，评论回复操作，删除朋友圈动态，上滑是否加载更多

界面/易用性测试

页面布局排版，文字排序，操作是否简易

网络测试

不同的运营商，网络切换测试：WIFI/4G/3G/2G，无网测试：对于缓存在本地的数据，部分朋友圈信息是否支持浏览

弱网测试：

延时：页面响应时间是否可接受、不同网络制式是否区分超时时长、出现请求超时，是否给予相应的提示

 兼容性测试

不同的操作系统，不同的手机子型号（Android手机端、苹果手机端、pad版（主流）功能界面显示是否正常）

性能测试

服务器性能测试：通过loadrunner工具，测试TPS、响应时间、吞吐量、CPU、内存等

app压力稳定性测试：通过monkey工具实现，（adb shell monkey -p 应用包名 1000 ）频繁发送朋友圈，浏览朋友圈请求，是否容易发生崩溃

# 复制测试：

1.路径测试

复制成功，记录下传输时间

PC->移动硬盘，移动硬盘-> PC

2.复制类型测试

单文件，多文件，,文件内容，快捷键：Ctrl + C，鼠标，快捷键鼠标交互

1. 异常测试

断电，断网，路径错误，路径被删除，或找不到，复制过程中文件被删除或文件被移动

3.功能测试

将同一内容复制多次到不同的路径下，进行复制操作后，对应文件的属性是否保持不变

4.兼容性测试

是否可以在不同的平台下进行复制(例如将将本机的内容复制到虚拟机中等)。

5.压力测试

同时进行多个大容量文件夹的复制，查看系统是否能正常复制的次数和持续时间

6.容量测试

将一个大文件夹同时拷贝到多个不同的路径下，持续不断地进行，一直到系统资源消耗殆尽，检查复制的最大容量

# 文件上传功能测试的测试点

1.选择符合要求的文件，上传，看是否上传成功

2.查看下载上传成功的文件，看上传的文件是否可查看或下载

3.删除上传成功的文件，看文件是否可删除

5.路径是否可以手工输入如不可以,给出提示

7.输入正确的文件路径，上传，在上传过程中，在本地修改上传文件名或文件类型

8.输入正确的文件路径，上传，在上传过程中，在本地修改上传文件夹名或文件类型

9.输入正确的文件路径，上传，在上传过程中，在本地删除上传文件

10.输入正确的文件路径，上传，在上传过程中，在本地删除上传文件夹

11.手动输入错误的文件路径，上传，是否提示不能上传

12.上传文件最大值，小于等于最大值,上传成功;大于最大值,上传失败.

13.上传文件格式是否全部支持，图片：gif/jpg/bmp...文档：doc/ xls...压缩包：zip/rar...安装文件：exe/msi

14.上传文件是否支持中文名称

15.文件名称的最大值、最小值

16.文件名称是否支持特殊字符(包括空格)

17.上传过程断网

18.上传时网速很慢，超过一定时间,提示

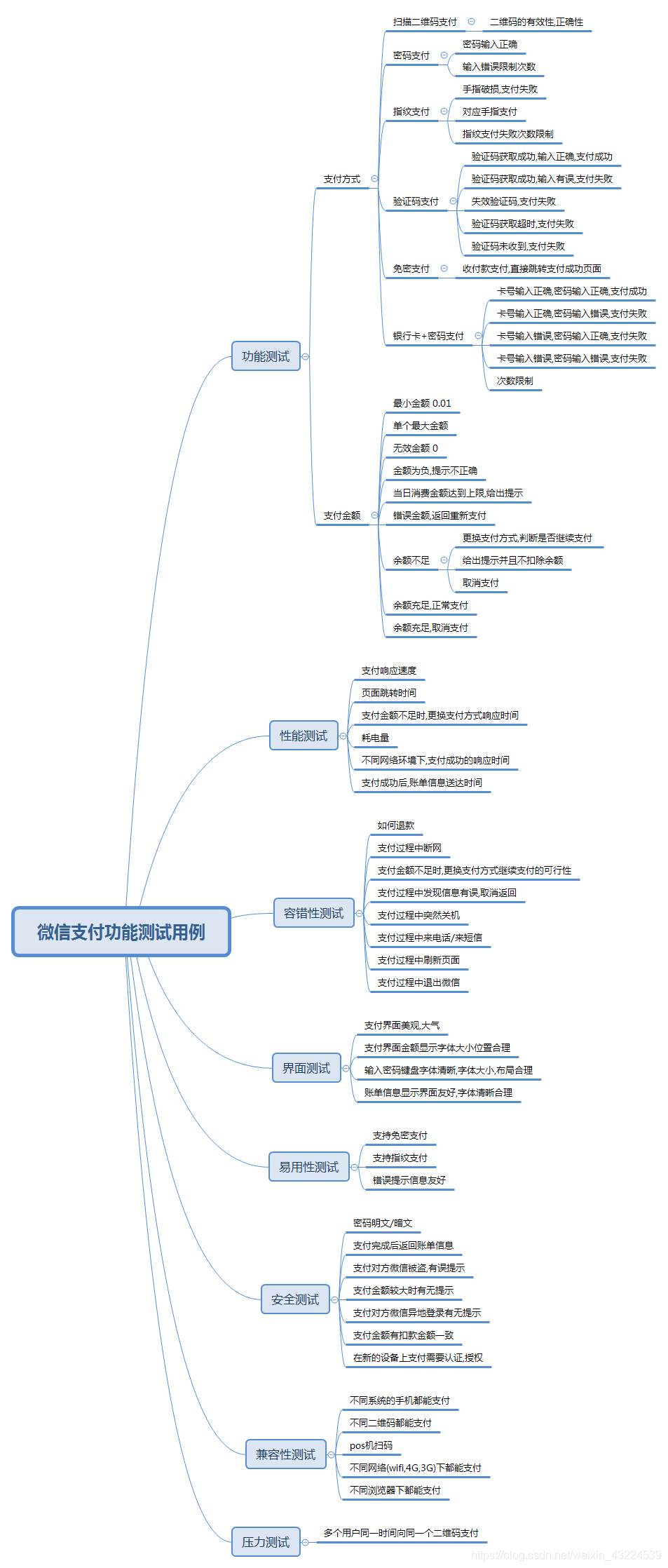
19.界面测试-说明性文字是否正确

20.界面测试-提示性文字是否正确

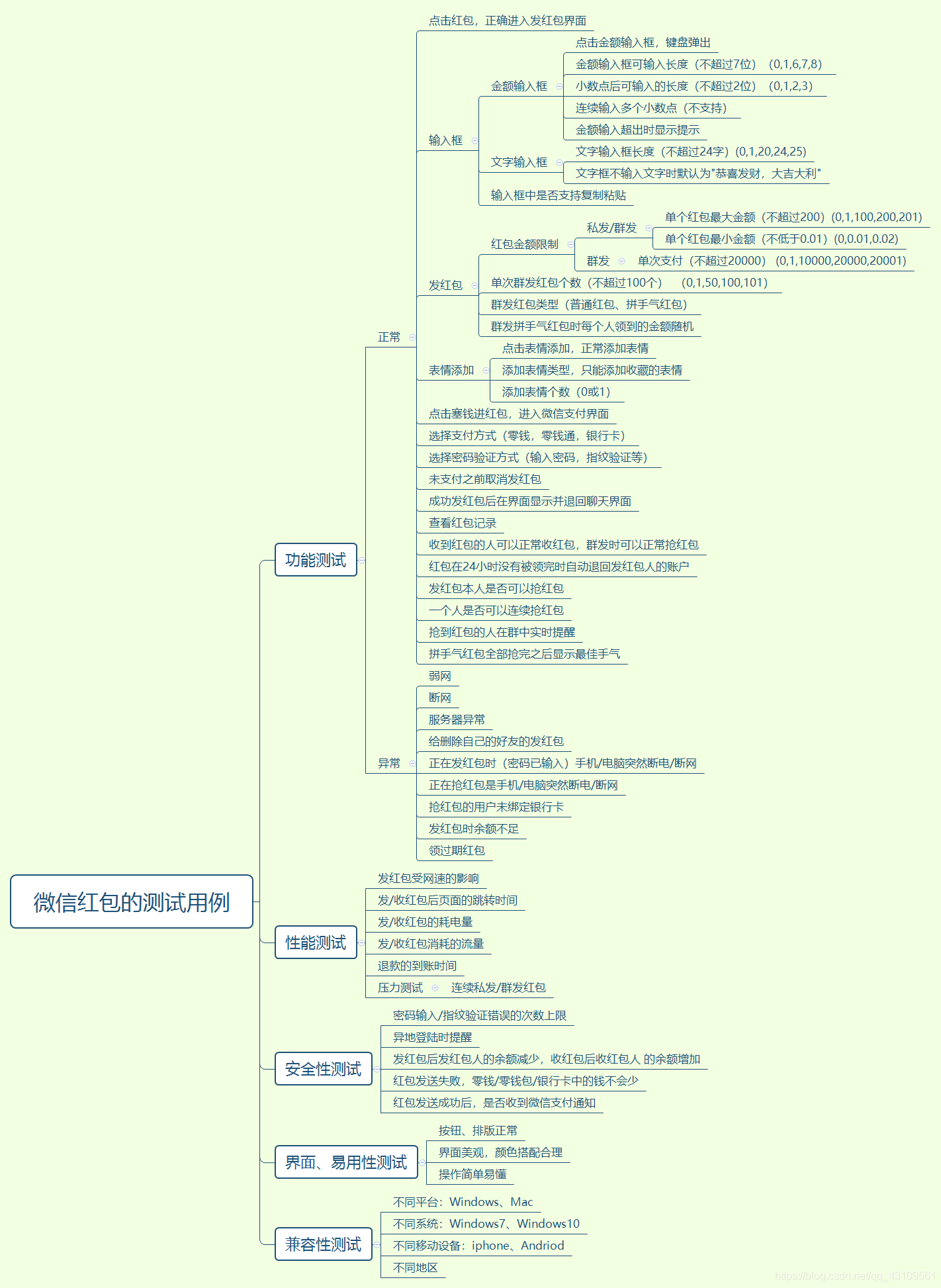
21.选择好但是未上传的文件是否可以取消选择

22.是否可以同时上传多个文件

# 微信支付功能：



# 微信红包用例（拼手气红包）：



# 问：给你一个网站，你如何测试？

（1）首先，查找需求说明、网站设计等相关文档，**分析测试需求**。

（2）**制定测试计划，确定测试范围和测试策略**，一般包括以下几个部分：**功能性测试；界面测试；性能测试；数据库测试；安全性测试；兼容性测试。**

**（3）设计测试用例：**

**功能性测试**可以包括，但不限于以下几个方面：

* 链接测试。链接是否正确跳转，是否存在空页面和无效页面，是否有不正确的出错信息返回。
* 提交功能的测试。
* 多媒体元素是否可以正确加载和显示。
* 多语言支持是否能够正确显示选择的语言等。

**界面测试**可以包括但不限于一下几个方面：

* 页面风格统一，美观
* 页面布局合理，重点内容突出
* 控件是否正常使用
* 对于必须但未安装的控件，是否提供自动下载并安装的功能
* 文字检查

**性能测试**一般从以下两个方面考虑：

压力测试；负载测试；强度测试

**数据库测试**要具体决定是否需要开展。数据库一般需要考虑连结性，对数据的存取操作，数据内容验证等方面。

**安全性测试**：

* 基本的登录功能的检查
* 是否存在溢出错误，导致系统崩溃或者权限泄露
* 相关开发语言的常见安全性问题检查，例如SQL注入等
* 如果需要高级的安全性测试，确定获得专业安全公司的帮助，外包测试，或者获取支持

**兼容性测试**，根据需求说明的内容，确定支持的平台组合：

* 浏览器的兼容性；
* 操作系统的兼容性；
* 软件平台的兼容性；
* 数据库的兼容性

（4）开展测试，并记录缺陷。合理的安排调整测试进度，提前获取测试所需的资源，建立管理体系（例如，需求变更、风险、配置、测试文档、缺陷报告、人力资源等内容）。

（5）定期评审，对测试进行评估和总结，调整测试的内容。

# 添加购物车用例：



# Adb，monkey命令

Adb devices连接手机

Adb install 安装软件

abd uninstall 卸载软件

adb push 文件路径 /sdcard/ ；将电脑文件传输到手机里面

adb pull /sdcard/文件路径 电脑路径 ；将手机资料导入电脑里面

Adb logcat 查看日志 \*：E错误日志 \*：F致命日志

延时参数：--throttle 1000，指定事件之间的间隔为1000毫秒；

种子参数：-s 100，指定随机生成树的seed值为100，使得两次的随机操作事件完全一样，可以用来复现bug；

触摸事件参数：--pct-touch 50，设置触摸事件的百分比为50%，剩余的事件会随机；

动作事件参数：--pct-motion 30，设置动作事件的百分比为30%，剩余的时间会随机；

轨迹球事件：--pct-trackball <percent>，设置轨迹球事件的百分比；

基本导航事件：--pct-nav <percent>，设置基本导航事件的百分比，输入设备的上主要导航事件：--pct-majornav <percent>，设置主要导航事件的百分比，兼容中间键，返回键，菜单按键；

# python函数跟脚本：

list=[1,2,3,4,5]列表

def sunof（）：定义函数

import time 导入参数time

list.append（5）在列表后面加5

list.insert（3,7）在序列3的位置插入一个7

list.extend(list2)把list2添加到list后面

list.count(3)查询列表中3的个数

list.index(n)查询n在列表中的序位

list.insert(1,100)在列表list中序位1的地方加上数字100

list.pop(3)删除列表中序位3上的数字

list.remove(2)删除列表中值为2，且排在最前面的数

list.reverse()将列表中所有的值翻转过来

list.sort()将列表中的值从小到大排序

d={}字典

d=['a']='one'

d=['b']='tow'

d=['c']='three'

print d 查询字典d中所有的键值

print d ['b']查询字典d中键为b的对应值

for ker，value in d.items()

print "%s:%s"%(key,value)

用for循环查询字典d中的所有键值（key键，value值）

d.clear()删除字典d种所有得键值

d1=d.copy(）将字典d中的所有键值复制到字典d1中

d.get(a)查询字典d种键为a的对应值

d.has\_key(a)查询字典中是否有键a

for items in d.items()：得到字典d中的键值组成的列表

for key in d.keys（）：得到字典d中所有的键

for value in d.values（）：得出字典d中所有的值

d.pop(a)删除字典d中的键a，但保留键a的值

d.popitem（）依次删除字典中的键值

d.setdefault(a)查询字典中键a的对应值

d.undate（d1）将字典d1中的键值加到字典d中

d[1]查询字典d中序位为1的值

d[-1]查询字典中序列为最后一个的值

首行➕#coding=utf-8 识别中文

Ture真

False假

raw原生态，未处理的

input输入

ouput输出

if如果

elif否则，如果

else否则

integer整数

print运行，输出

％s字符串占位符

%d整数占位符

％.2小数占位符

while循环 当……的时候（不知道循环范围，需要自己制定）

while Ture 循环正确的（后面加条件）

for循环（事先知道范围）

range范围（0，5）例如范围0到5，左边界为0，右边界却为4，不包括最右边的数

# 冒泡排序

#coding=utf-8

list1 =[1,4,5,6,3,8,9,0,7,2,10,13,11] #冒泡排序

print"排序前："

print list1

for i in range(1,len(list1)):

for j in range(i,len(list1)):

if list1[i - 1] > list1[j]:

temp = list1[i-1]

list1[i-1]=list1[j]

list1[j]=temp

print"排序后："print list1

# 乘法口诀表

# coding=utf-8for i in range(1,10):

for j in range(1,i+1):

print"%d x %d = %d" %(j, i, j\*i),

print

#乘法口诀表

# 接乘

# coding=utf-8

n=input("输入一个数字：")

jc=1

for i in range(l, n +l):

jc=jc i

print jc

#接乘

# 拿鸡蛋

n=1

while True:

if n %2==1 and n %3==0\

andn%4==1 and n%

5==4 and n%6==3

and n%7==0 and n%

8==1 and n%9==0

print n

break

n=n+1

#拿鸡蛋

# 三角形

#coding=utf\_8

#等腰三角形打印

import randomfor i in range(1, 11):

for n in range(i,11):

print" ",

for m in range(1,2\*i):

print"\*",

Print

# 三角形打印，手动输入层数

def sjx(n):

for i in range(1, n + 1):

for k in range(i, n):

print (" "),

for j in range(1, 2 \* i):

print ("\*"),

printn = input("输入需打印的三角形层数：")sjx(n)

# 数据库：

表格命令：

1进入数据库

use test；进入数据库test

2查询数据库所有的表

show tables；查询数据库所有的表

3创建表格

create table 表名字 （字段名 字段类型，）；创建表格

（字段类型：纯整数数字用int 或bigint或float，带字符符号数字汉子的用varchar（）或char（），带小数的用double，年月用date）

4删除表格

drop table 表名；删除表格

5查看表的结构

desc 表格名；查询表格的结构

6修改表名

alter table 表名 rename 新名字；修改表名

7修改表的字符集

alter table 表名 character set utf8；只修改表格的字符集为utf8

Alter table 表名convert to character set latin1修改表格的字符集以及表里面所有的字段的字符集为latin1

8往表格添加数据

insert into 表格名 values ('a','b,','c',null）;添加表格数据，为空不填的用null表示

insert into 表名 values （1,"张三"，‘22’，‘1.75’，‘2020-01-09’），(2,‘李四’，‘22’，‘1.77’）；一次添加多个数据到表格

先用Excel创建一个数据表格，日期斜杠修改，右键全部日期，单元格设置，自定义修改格式保存，在复制粘贴到记事本，粘贴后注意删掉所有数据后面的空格，保存到E盘根目录，文件名\*\*.txt：

load data local infile 'e:/\*\*.txt' into table 表名;批量插入数据

Insert：插入数据

Insert into表名values（数据）；插入数据

表格中有中文导致错位的话在后面加上lines terminated by '/r/n'

9修改表格中的数据

update 表名 set 字段名 = '修改后的数据 'where name = '要修改数据所在的位置'；

10删除表格数据

delete from 表格名;删除整个表格

delete from 表格名 where name ='\*\*';根据名字删除整行内容

11 往表格添加字段

alter table s add sex char(3); 用add添加字段，默认添加到最后面

alter table s add (age int, sex char(3)); 用add依次添加多个字段，字段写在括号内（）

alter table s add age int after name; 用（after 字段名）指定添加到那个字段后面

12删除字段

alter table s drop age; 用drop删除表s中的字段age

13修改字段名

alter table s change id sno char(10);

用change修改字段名id为 字段sno 字段类型char（10）；

alter table s modify id int(10);

用modify修改字段id的字段类型为int（10）（不能修改字段名）

14修改字段位置

alter table 表名 modify 字段1 定义 after 字段2;

修改字段位置，用modify将字段1添加到字段字段2后面

15查看表的构造

show create table 表名;

16设置字段的默认值

alter table 表名 modify 字段 定义 default'不详'; 设置字段没有数据就显示默认值 不详

Delete：删除数据

Delete from表名 where字段名=数据

Drop删除：

Drop use用户名；删除用户

Drop table 表名：删除表

Drop database 数据库名：删除数据库

Drop view视图名：删除视图

Update：修改数据

Update 表名set字段名=更改后的数据where字段名=表中有的数据；修改数据

Set：更改密码

Set password for 用户名=password（‘新密码’）：修改密码

数据库库命令：

Create：创建

Create database 数据库名; 创建数据库

select查询语法：

\*代表所有的字段

select \* from 表名； 查询表里面所有的数据

不用\*直接用字段查询

select 字段1，字段2 from表名 ;直接查询表格中的指定字段

模糊查询：

%代表任意字符的任意个数 \*所有字段（a%表示a开头、%a表示a结尾、%a%包含a，“\_”一个下划线代表一个字符，一个汉字三个字符）

     select \* from 表名where 字段 like 'b%'; 查找字段中以'b'开头的数据

计数行  count计数( )：

     select count(\*) from 表格;  查询表格里面有多少行

分组 group by

select 字段 from 表名group by 字段;   分组查询，查询表里面该字段数据分别出现的次数

排序 order by：

  select 字段 from 表名 order by 字段; 对该字段数据进行排序(asc升序，默认升序)

    加desc变倒序

分页显示：

select \* from 表格 limit 5;

区间查询：

select \* from 表名 where 字段>= 60 and 字段<70;查询表格中60-70分之间的分数

between 60 and 69;同上，用between表示60-70之间,

交叉连接查询：

select \* from 表1，表2; 将表格表1跟表2交叉连接，所有的数据都交叉一遍

显示内连接：

inner join：内部连接

select \* from 表1 inner join 表2 on 表1.字段=表2.字段;显示内连接，用inner join连接另一个表，用on 连接条件两个表的相同字段筛选没意义重复的组合，不显示筛选条件为空的组合

隐式内连接：

select \* from 表1，表2 where 表1.字段=表2.字段; 隐式内连接，将一些没意义的重复的组合剔除，不显示筛选条件为空的组合

子查询：子查询也叫嵌套查询，是指在select子句或者where子句中又嵌入select查询语句

select \* from 表2where 字段1 =( select sno from 表2 where子句);

先用查询到表1的数据，然后查询表格表2的数据，用where子句相同字段连接表1表2

alter修改语法：

字段约束：Not null不能为空

alter table 表名 modify 字段 定义 not null;

修改表格的字段名不能为空null

## 对数据的增删改查：

1. 新增：insert into 表名 values（数据 ）
2. 删除：delete from 表名 where 子句 ；
3. 修改：update 表名 set 字段名 = 新的数据 where子句；
4. 查询：select \* from 表名；

## 对数据库的增删改查：

1. 新增：create database 新增数据库
2. 删除：drop database 数据库名
3. 修改：alter showatabase 数据库名 character set 新的字符集
4. 查询：show databases；查看服务器有哪些数据库

Show create database 数据库名；查看数据库的创建

## 对表的增删改查：

1. 新增：create table 表名 （字段定义 ）
2. 删除：drop table 表名 ；
3. 修改：alter table 表名 修改内容
4. 查询：show tables；查看数据库里有那些表

Desc 表名；查看表的结构

Show create table 表名；查看表创建

# linux系统：

Linux[nei nik si]系统各个文件夹作用：

◎cd /加文件名/加文件夹子名（ls打开）

◎bin[bing]:binary[bai nei rui]二进制，用于存放linux常用命令

◎dev[dai fu]:device[dei fai si]设备，用于存放计算机设备文件

◎home:家，用于存放普通用户的家目录

◎root：管理员，用于存放管理员文件资料

◎lib：library[lai bu rui]仓库，用于存放linux资源

◎lib64：64位资源仓库

◎media[mi der]：媒体，存放媒体资源

◎opt：option[a pu sheng]选项，用于安装软件

◎selinux:linux系统内核

◎sys：system[se si te]系统，存放linux系统资源

◎usr：Unix[u naik si] System Resource[rei so si]，Unix系统资源

◎boot[bu te]：引导，启动操作系统

◎etc：杂项，存放linux系统重要的配置文件，子文件夹（passwd：linux账号文件夹、shadow：存放linux密码文件夹、profile：环境变量文件夹）

◎lost+found：丢失+找回，linux系统意外断电时某些正在运行的资源文件可以在此文件夹找回

◎mnt：mount[mao te],挂载，用于挂载外部存储设备（如u盘）

◎proc：process[pu rou sai si]进程，存放linux所有进程（运行中的程序）文件

◎sbin：super[su po] binary[bai na rui]，超级用户二进制文件

◎srv：service[se vi se]服务，存放各种服务信息文件

◎tmp：temp[tai pu]临时，存放临时文件

◎var：variable[wai a ri bo]可变的，存放可变信息文件（如日志）。

linux日志存放路径：/var/log

文件类型：

d 开头，文件夹

- 开头，普通文件

l 开头，链接文件

远程连接：

ssh +服务器ip地址 然后输入管理员账号密码

显示所有文件的详细信息:

ls -al /加文件夹名 显示指定文件夹的所有文件的详细信息

进入目录：

cd 进入目录

cd ..返回到上层目录

创建目录：

mkdir -p一次创建多层目录

删除空目录

rmdir -p 一次性删除多个空目录

清除屏幕缓存：

clear 清除屏幕缓存。

rm 删除文件或文件夹：

-r 递归删除（一层一层的删除目录里面的内容）

-f 强制删除

cp 复制文件或目录：

cp /要复制的文件的完整路径 /要放置文件的目标路径，加-r 递归复制文件目录

mv 剪切文件或目录：

mv 源文件或源目录(多个文件) 目标位置

touch创建一个txt文件

history 查看历史记录：

file 查看文件类型：file 文件名

cat 从第一行查看，只适用于小文件：

tac 从最后一行开始显示：

more 分页查看文件内容，只能向翻页：

less 分页查看文件内容，可以上下翻页（more不能上下翻页），按q键退出：

head 指定从第一行到多少行，显示文件内容：

tail 语法与head 相反：

动态查看日志tail -f 文件名（动态查看日志），tail -5f 文件名 （实时查看五行日志），ctrl+c 退出查看日志

关机的方式：

shutdown -h(定时关机），shutdown -r（定时重启）

init 0关机，3命令模式，5图形模式，6重启

Killall 加进程名，杀死进程

ps -ef 查看进程id ;process snapshot 进程快照，查看进程

killall -9 加进程id -9立即杀死

find查的目录中的文件：

-name根据名字查询

-user 根据用户

find ./ -name 文件名 根据名字查询

find -name文件名 -exec ls -l {} \; 根据名字查询详细信息

su用户切换：

> >>搜索历史保存文件重定向:

>重定向将命令的执行结果，保存到文件，会覆盖之前的内容

>>追加，将命令结果保存到文件后面，不会覆盖之前的内容

echo $PATH #查看环境变量

添加永久变量：

vi /etc/profile 用vi打开etc中的profile文件，然后将要添加变量路径添加到文件的最后面

| 管道：

查找命令历史记录中的PATH命令

history | grep PATH

查看文件a中的第八行，用head跟tail组合

head -8 a | tail -1（用head查看第八行，|管道连接再用tail查看最后一行）

ls -al /etc | grep -c ^- （统计etc中普通文件的数量）

grep命令是一种强大的文本搜索工具：

－c：只输出匹配行的计数。（重点记）

grep -c cp a （查询文件a中所有cp的数量）

^开头

$结尾

.表示任意字符。

查看当前目录pwd：print working directory

Tree安装命令：

yum前端软件包管理工具

1、检测是否已经安装，已经安装则删除

rpm -qa | grep mysql

2、查看在软件仓库中是否存在

yum list | grep mysql

3、从软件仓库中下载并安装

yum install -y

添加用户：useradd 用户名：

删除用户userdel -r 彻底删除，直接用rm -rf删除更加快捷

修改当前用户密码Passwd 用户名；

Vi 查看文件内容编辑：

vi +文件名

按insert键，进入编辑模式

Esc键退出编辑模式

进入vi编辑器，按i方可进行编辑

按(:)号进入命令模式

编辑完成，按ESC-->:w保存修改内容

编辑完成，按ESC-->:q保存文档

强制退出，按ESC-->:q!不会保存退出vi

用户文件权限：

-rwxrw-r-- : -、rwx、rw-、r--（r，读、w，写、x，运行）

第1位，文件类型（d：目录，-:普通文件，l：链接文件）

第2-4位，所属用户权限，用u（user）表示

第5-7位，所属组权限，用g（group）表示

第8-10位，其他用户权限，用o（other）表示

第2-10位，表示所有的权限，用a（all）表示

chown改变文件的所属用户或组，命名格式如下：

chown 用户 文件名或目录

chmod；change mode，修改权限，格式如下：

chmod o+w 文件名

任何权限修改限制都对管理员无效！！！

用数字来表示权限：(r=4 ,w=2 ,x=1 ,-=0)

rwx二进制是111，十进制是4+2+1=7

r-x二进制是101，十进制是4+0+1=5

tar 归档（压缩解压）：

解压：tar -xvf 压缩包名 tar -xvf 111.zip

压缩：tar -zcvf 压缩包名 文件名tar -zcvf 111.zip 111

ps -ef 查看当前所有运行的进程：

top 检测cpu，内存

查询端口占用 ：

netstat -anp | grep +端口号 ：查询端口占用

端口号书写方式” 80 ”

ifconfig查看ip地址

重启网络服务：

service network restart

service:服务管理命令：

常用操作包括：start启动、stop停止、restart重启、status查看状态

关闭网卡，service network stop

开启网卡，service network start

关闭防火墙服务，service iptables stop

关闭防火墙自动启动，chkconfig iptables off

set 可以显示所有变量