1、简述HTTP

答：tcp应用值HTTP传输

http协议 ---> 超文本传输协议 应用层协议

用途：网页的获取，基于网站的数据传输

基于http协议的数据传输

特点：

1.一个应用层协议，传输层使用tcp传输

2.简单灵活，和多种语言对接方便

3.无状态协议，不记录用户的通信内容

4.成熟稳定http1.1

工作模式：

1.使用http双方均遵循http协议规定发送接收消息体。

2.请求方，根据协议住址请求内容发送给对象

3.服务方，收到内容按照协议解析

4.服务方，将回复内容按照协议组织发送给请求方

5.请求方，收到回复根据协议解析

2.Selenium

答：Selenium [1] 是一个用于Web应用程序测试的工具。Selenium测试直接运行在浏览器中，就像真正的用户在操作一样。支持的浏览器包括IE（7, 8, 9, 10, 11），Mozilla Firefox，Safari，Google Chrome，Opera等。这个工具的主要功能包括：测试与浏览器的兼容性——测试你的应用程序看是否能够很好得工作在不同浏览器和操作系统之上。测试系统功能——创建回归测试检验软件功能和用户需求。支持自动录制动作和自动生成 .Net、Java、Perl等不同语言的测试脚本。

3、简述一下sql去重

答：

数据库中的去重操作（删除数据库中重复记录的SQL语句）主要有二种方法

（1）、group by 方法

主要用于分组统计,一般都是使用在聚合函数中使用；

（2）、distinct方法

一般用于比较小的表进行去重,会过滤掉多余的重复记录，返回不重复的记录或字段；

4.简述判断链表是否为循环链表

答：

为了验证链表是不是循环链表，首先要创建循环链表，其次是打印循环链表，验证循环链表是不是创建成功，最后写函数判断是不是循环链表。

创建循环链表代码如下所示：

/\*\*

\* @author jcm

\* @see 创建循环链表

\* @param list

\*/

public void createCircularLinkList(Node head) {

//构建循环链表

Node current = head;

while(current.next != null){

current = current.next;

}

current.next = head;

}

打印循环链表代码如下所示：

/\*\*

\* @author jcm

\* @see 打印循环链表

\* @param head

\*/

public void printCircularLinkList(Node head){

Node current = head;

while(current.next != head){//判断最后一个结点是不是指向头结点

System.out.print(current.data+" ");

current = current.next;

}

System.out.print(current.data+" ");

}

判断是不是循环链表代码如下所示：

/\*\*

\* @author jcm

\* @see 判断是不是循环链表

\* @param head

\* @return

\*/

public boolean isCircularLinkList(Node head){

if(head == null || head.next == null){

return false;

}

Node firstNode = head;

Node secondNode = head;

while(firstNode != null){//first走的快，如果条件不成立，说明可定不是循环链表

if(firstNode != null){

firstNode = firstNode.next;//每次循环要走两步

secondNode = secondNode.next;

}

if(firstNode != null){

firstNode = firstNode.next;

}

if(firstNode == secondNode){//判断是不是相等

return true;

}

}

return false;

}

5、简述爬虫如何处理附件问题

答：

6.简述内键函数Map resure filter 区别

答：

map() 会根据提供的函数对指定序列做映射。

第一个参数 function 以参数序列中的每一个元素调用 function 函数，返回包含每次 function 函数返回值的新列表。

filter() 函数用于过滤序列，过滤掉不符合条件的元素，返回由符合条件元素组成的新列表。该接收两个参数，第一个为函数，第二个为序列，序列的每个元素作为参数传递给函数进行判，然后返回 True 或 False，最后将返回 True 的元素放到新列表中。

reduce（）函数

reduce 函数可以按照给定的方法把输入参数中上序列缩减为单个的值，具体的做法如下：首先从序列中去除头两个元素并把它传递到那个二元函数中去，求出一个值，再把这个加到序列中循环求下一个值，直到最后一个值 。