Daniel Lambert

1)
1.1)
```
sqlite3 account.db
SQLite version 3.7.17 2013-05-20 00:56:22
Enter ".help" for instructions
Enter SQL statements terminated with a ";"
```

1.2)
```
create table lab4_account (account_number  char(10) primary key, balance double not null);
```

1.3)
```
insert into lab4_account values('A001', 100);
insert into lab4_account values('B001', 999);
insert into lab4_account values('C001', 99.99);
```

1.4)
```
select * from lab4_account;
A001|100.0
B001|999.0
C001|99.99
```

1.6)
```
ls -l accounts.db
-rwx------. 1 djlamber isilon_du 3072 Apr 24 09:31 account.db
file account.db
account.db: SQLite 3.x database
```

1.3)

```
import java.sql.*;

public class SQLiteJDBC {
   public static void main(String args[]) {
         Connection c = null;
         Statement stmt = null;
         try {
                  Class.forName("org.sqlite.JDBC");
                  c = DriverManager.getConnection("jdbc:sqlite:account.db");
                  stmt = c.createStatement();
                  ResultSet rs = stmt.executeQuery("SELECT * FROM lab4_account;");
                  System.out.println("Acc# Balance");
                  while (rs.next()) {
```

```
                        String accNo = rs.getString("account_number");
                        double balance = rs.getDouble("balance");
                        System.out.println(accNo + " " + balance);
                }
                rs.close();
                stmt.close();
                c.close();
        } catch (Exception e) {
                System.err.println(e.getClass().getName() + ": " + e.getMessage());
                System.exit(0);
        }
   }
}
```

Compiled in Eclipse

Output:

```
Acc# Balance
A001 100.0
B001 999.0
C001 99.99
```

2.2.2)

a)

```
/*/*/*/RAM/data()
1024
512
512
1024
2048
1024
2048
```

b)

```
/*/*/*/@price/data()

2114
995
1150
2500
959
649
3673
239
100
```

200
c)
/*/*/@name/data()

A
B
H
d)
/*/Maker[Printer/Type = "laser"]/@name/data()

H
e)
/*/Maker[PC or Laptop]/@name/data()

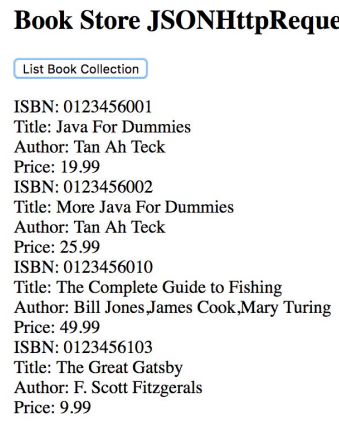
A
B
f)
/*/Maker[count(PC)>=2]/@name/data()

A
B

2.3)
```
let $products := doc("products.xml")
for $maker in $products//Maker
where count($maker/PC[Speed > 2.0]) >=2
return <Maker name="{$maker/@name}" />
```

<Maker name="A"/>

3)
https://classdb.it.mtu.edu/~djlamber/listbookJSON.html



Source Code:

```
<!DOCTYPE html>
<html>
<body>
<h2>Book Store</h2>

<button type="button" onclick="LoadJSONDoc()"> List Book Collection JSONHttpRequest
</button>
<p id="demo"> Data will go here</p>

<script>
function LoadJSONDoc(){
  var xhttp=new XMLHttpRequest();
  xhttp.onreadystatechange = function(){
    if(xhttp.readyState == 4 && xhttp.status == 200){
      displayBook(xhttp.responseText);
    }
  }
  xhttp.open("GET", "bookstore.json", true);
  xhttp.send();
}

function displayBook(bookdata){
  var bookstore = JSON.parse(bookdata);
  var books = bookstore.book;
```

```
  var details = "";

  for(i=0; i<books.length; i++){
    var book = books[i];
    details = details + "ISBN: " + book["-ISBN"]+"<br/>";
    details = details + "Title: " + book["title"]+"<br/>";
    details = details + "Author: " + book["author"]+ "<br/>";
    details = details + "Price: " + book["price"]+ "<br/>";

  }
  document.getElementById("demo").innerHTML=details;
}

</script>
</body>
</html>
```

4)
https://classdb.it.mtu.edu/~djlamber/listbookXML.html

Book Store XMLHttpRequest

List Book Collection

Book0
ISBN:0123456001
Title: Java For Dummies
Author: Tan Ah Teck
Price: 19.99

Book1
ISBN:0123456002
Title: More Java For Dummies
Author: Tan Ah Teck
Price: 25.99

Book2
ISBN:0123456010
Title: The Complete Guide to Fishing
Author: Bill Jones
Author: James Cook
Author: Mary Turing
Price: 49.99

Book3
ISBN:0123456219
Title: The Great Gatsby
Author: F. Scott Fitzgerald
Price: 9.99

```
<!DOCTYPE html>
<html>
<body>

<h2>Book Store XMLHttpRequest</h2>

<button type="button" onclick="LoadXMLDoc()"> List Book Collection </button>
<p id="demo"> Data will be here</p>
<script>
```

```
function LoadXMLDoc(){
  var xhttp = new XMLHttpRequest();
  xhttp.onreadystatechange = function (){
    if(xhttp.readyState == 4 && xhttp.status == 200){
      displayBook(xhttp.responseXML);
    }
  }
  xhttp.open("GET","bookstore.xml", true);
  xhttp.send();
}

function displayBook(xmlDoc){
  var information="";
  books = xmlDoc.getElementsByTagName("book");
  for(i=0; i<books.length; i++){
    information = information.concat("Book", i, "<br/>");
    information = information.concat(" ", "ISBN:", books[i].attributes[0].nodeValue, "<br/>");

    children = books[i].childNodes;
    for(j=0; j<children.length; j++){
      if(children[j].nodeType == 1 && children[j].nodeName == "title") {
        information = information.concat(" ", "Title: ");
        information = information.concat(children[j].childNodes[0].nodeValue,"<br/>");
      }
      if(children[j].nodeType == 1 && children[j].nodeName == "author"){
        information = information.concat(" ", "Author: ");
        information = information.concat(children[j].childNodes[0].nodeValue, "<br/>");
      }
      if(children[j].nodeType == 1 && children[j].nodeName == "price"){
        information = information.concat(" ", "Price: ");
        information = information.concat(children[j].childNodes[0].nodeValue, "<br/>");
      }
    }
  information = information.concat("<br/>");
  }
document.getElementById("demo").innerHTML=information;
}
</script>
</body>
</html>
```

5)
https://classdb.it.mtu.edu/~djlamber/activity.html