Question 1:
Select a domain-specific language, and give an example of how a general-purpose language (possibly with a library) could be used to provide the same functionality.

Java to implement the same functionality of SQL by using the library of java.sql.
A simple example of SQL to create a valid SQL SELECT query can be:

```
SELECT Lname FROM Customers
WHERE Snum = 2001;
```

The java code that can do the same thing look like:

```
Statement stmt = conn.creatStatement();
ResultSet rs = stmt.executeQuery(“SELECT Lname FROM Customers WHERE Snum =
2001”);
```

Question 2:
Select a domain-specific language, and mention its domain of use. Write the signature for four functions that an equivalent application library would offer. What disadvantages would be imposed on a developer when using this application library as opposed to using the domain-specific language?

Using Java JDBC API to implement the functionality of SQL:

```
import java.sql.*;

DriverManager.registerDriver(
new oracle.jdbc.driver.OracleDriver()
);
Connection conn = DriverManager.getConnection (
"jdbc:oracle:thin:@Server",
“Username", “Password”);

//Passing query
Statement sttable = conn.createStatement();
sttable.executeUpdate(
"CREATE TABLE Fruit(Name VARCHAR(10),Amount INT)“
);
sttable.close();
Statement stinsert1 = conn.createStatement();
stinsert1.executeUpdate(
"INSERT INTO Fruit VALUES('Apple', 5)“
);
stinsert1.close();
Statement stinsert2 = conn.createStatement();
stinsert2.executeUpdate(
"INSERT INTO Fruit VALUES(‘Pumpkin', 1)“
);
stinsert2.close();

//Processing query results
Statement stresult = conn.createStatement();
```

```
ResultSet fruit = stresult.executeQuery(
"SELECT * FROM Fruit"
);
while(fruit.next()) {
System.out.println(
fruit.getString("Name")+ ", " +
fruit.getInt("Amount"));
}
fruit.close();

//Updating a row

// Make an updatable Statement
Statement result2 = conn.createStatement(
ResultSet.TYPE_SCROLL_SENSITIVE,
ResultSet.CONCUR_UPDATABLE);
ResultSet rset2 = result2.executeQuery(
"SELECT Name, Amount FROM Fruit");
rset2.absolute(2);// set current row to second
rset2.updateInt("Amount", 3); //
rset2.updateRow(); // updates the second row

//Inserting a row

// rset2 is set up as in the previous example
// Get ready to insert a row
rset2.moveToInsertRow();
// Put the values of the new row in each column
rset2.updateString(“Name", “Orange”);
rset2.updateInt(“Amount", 7);
// Add this row
rset2.insertRow();
// Go back to the row we were at before inserting
rset2.moveToCurrentRow();

//Closing
Connection conn;
try {
...
} finally {
if (conn != null) {
try {
conn.close();
} catch (...) {
// what to do?
}
}
}
```

To do the four functions stated above in SQL can be much simpler and the length of codes can be shorter. Because general purpose languages need to consider more about the functionality of general uses and normally implement the specific functionality in low level, it needs to describe in more details and to consider more circumstances round the functions. Indirectly, the general purpose language to implement specific functionality is more likely to have bugs due to the longer contents.

Question 3:
Select a domain-specific language, and provide an example of a design pattern that could not be applied to the language. Provide an explanation to why this is the case. Give an example of a design pattern that could be applied to the language.

HTML only describes what should appear on a webpage - it does not specify the possible interactions with that webpage, so it can’t support the patterns that need the interaction between the users and the servers, for example, MVC. HTML is a language specific for the webpages that provided be the servers for users to browse, so it supports the client-server pattern on the models that only provide the users static information to pull from the servers.