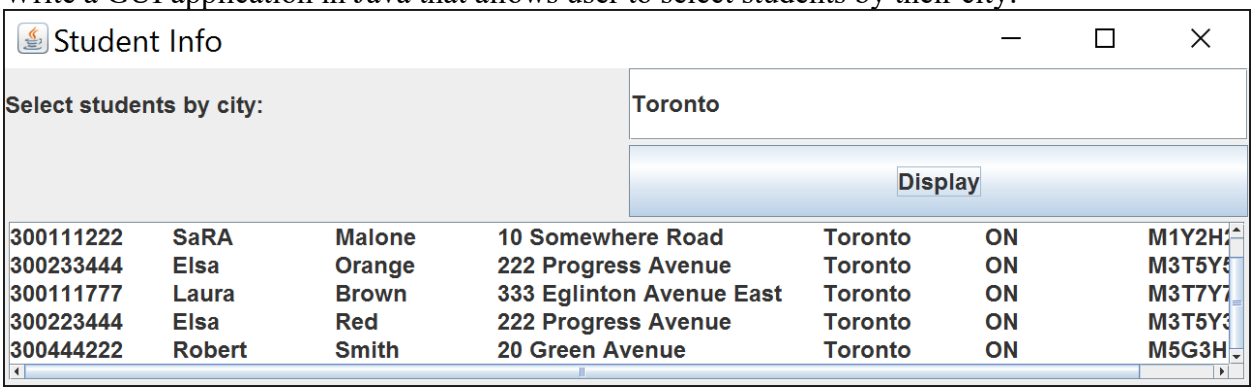
TEST2 (Hands-On part)

- There are **two exercises** in this test.
- You need to create **two** Eclipse Java projects to solve the **two** exercises given below. You should zip those Eclipse Java projects and submit the zip file through the dropbox link on eCentennial.

Exercise 1 (10 marks)

Create a new Eclipse Java project. Name the project: **YourFullName_Ex1**. For example: JohnSmith_Ex1. **DO NOT** create a package in this project. Let Eclipse create the default package.

Write a GUI application in Java that allows user to select students by their city.



The user should be able to enter the *city*. The application should retrieve the student information from an SQLite table named **Student**. Here is the definition of the table:

```
CREATE TABLE Student (
   studentID char(9) NOT NULL,
   firstName varchar (20) NOT NULL,
   lastName varchar (20) NOT NULL,
   address varchar (30) NOT NULL,
   city varchar(30) NOT NULL,
   province char(2) NOT NULL,
   postalCode char(6) NOT NULL,
   PRIMARY KEY (studentID)
);
```

Populate the table with several rows as below:

```
insert into Student values('300111222','Sam', 'Malone', '10
Somewhere Road', 'Toronto','ON','M1Y2H2');
```

If necessary, you may change the syntax of above SQL statements to be compliant with SQLite.

The information should be displayed in a JTextArea component which has scrolling abilities. Use "\t" and "\n" to format the display.

Use the most appropriate layout manager classes to implement the layout of this GUI.

Exercise 2 (10 marks)

Create a new Eclipse Java project. Name the project: **YourFullName_Ex2**. For example: JohnSmith_Ex2. **DO NOT** create a package in this project. Let Eclipse create the default package.

Below are two classes Element and Chain. An Element object has a *data* and a reference *next* to point to the next element. A Chain object contains a chain of Element objects. A Chain object has an instance variable called ***head***. The ***head*** always points to the first element of the chain. The Chain class should support the following methods:

- **`void add(int d)`** : adds an element with data d as the first element of the chain.
- **`void print()`** : prints the data of the elements (present in the chain).

Fill up the code for the above methods in the class Chain so that when the ***main*** method is run, the output should be:

```
3
2
```

Do not add any instance variable or static variable to the Chain class. You may add local variables in `add` and `print` methods of Chain class. **Do not change** the Element class. **You will receive 0 mark** in this question if you do not adhere to these instructions.

```
class Element {
    private int data;
    private Element next;

    public Element(int i) {data = i; next = null;}
    public int getData() { return data; }
    public Element getNext() { return next; }
    public void setNext(Element n) { next = n; }
}
public class Chain {
    private Element head;
    public Chain() { head = null; }

    public void add(int d) {
       //create an element with d as its data

       //add the element when chain has currently no element

       //add the element when chain has currently at least one element

    }
```

```
    void print() {
        //access the first element of the chain


        //iterate over the elements of the chain and print
        //data present in each element

    }

    public static void main(String[] args) {
        Chain c = new Chain();
        c.add(2); c.add(3); c.print();
    }
}
```