CIS 36B – 6th Class / Lab Assignment – **10 Points-**
Student Name KaChiLau Student ID 10819338 Point Total

Section 1: Define / Answer

Pg. 217-224, Java Programming *A comprehensive Introduction*

http://www.javatpoint.com/method-overloading-in-java

http://beginnersbook.com/2013/05/method-overloading/

http://beginnersbook.com/2013/05/constructor-overloading/

Define-

Polymorphism-

Polymorphism- The word 'polymorphism' literally means 'a state of having many shapes' or 'the capacity to take on different forms'. When applied to object oriented programming languages like Java, it describes a language's ability to process objects of various types and classes through a single, uniform interface

Method Overloading-(Define and Give a short code example)

Method Overloading is a feature that allows a class to have two or more methods having same name, if their argument lists are different. In the last tutorial we

discussed constructor overloading that allows a class to have more than one constructors having different argument lists.

Constructor chaining-(Define and Give a short code example)

Calling another constructor in the same class from another constructor is called constructor chaining. By using this() we can call another constructor in the same class. Incase we want to call another constructor, this() should be the first line in the constructor. Below example shows code for constructor chaining.

```
/* Constuctor overloading
   StudentInfo(){
      //give default
   }


   StudentInfo(int a, int b){
   }


   StudentInfo(String a, String b){


   }
   */


   /* Method overloading
   void sum(int a, int b) {


   }


   void sum(int a, int b, int c) {


   }
   */
```

CIS 36B – 6th Class / Lab Assignment – **10 Points-**

Student Name KaChiLau Student ID 10819338 Point Total

ONLINE RESOURCE: http://tutorials.jenkov.com/java-io/file.html

Understand and define methods in the Java File class-

```
java.io.File file = new java.io.File("filelocation.filetype");


    System.out.println("Does it exist?" + file.exists());


    System.out.println("The file has " + file.length() + " bytes ");


    System.out.println("Can it be read?" + file.canRead());


    System.out.println("Can it be written to?" + file.canWrite());


    System.out.println("Is it a directory" + file.isDirectory());


    System.out.println("Is it a file?" + file.isFile());


    System.out.println("Is it absolute? " + file.isAbsolute());

    System.out.println("Absolute path is " + file.getAbsolutePath();
    System.out.println("Is  it Hidden? " + file.isHidden());


    System.out.println("Last  Modified on  " + file.lastModified());


    System.out.println("Last  Modified on  " +

        new java.util.Date(file.lastModified()));
```

Task 1-

USE OBJECT ORIENTATED PROGRAM DESIGN TO SOLVE PROBLEM

Create a class called StudentInfo.

The class should contain a constructor for student ID, First Name, Last Name, DOB, and address.

Create 10 student objects matching the criteria.

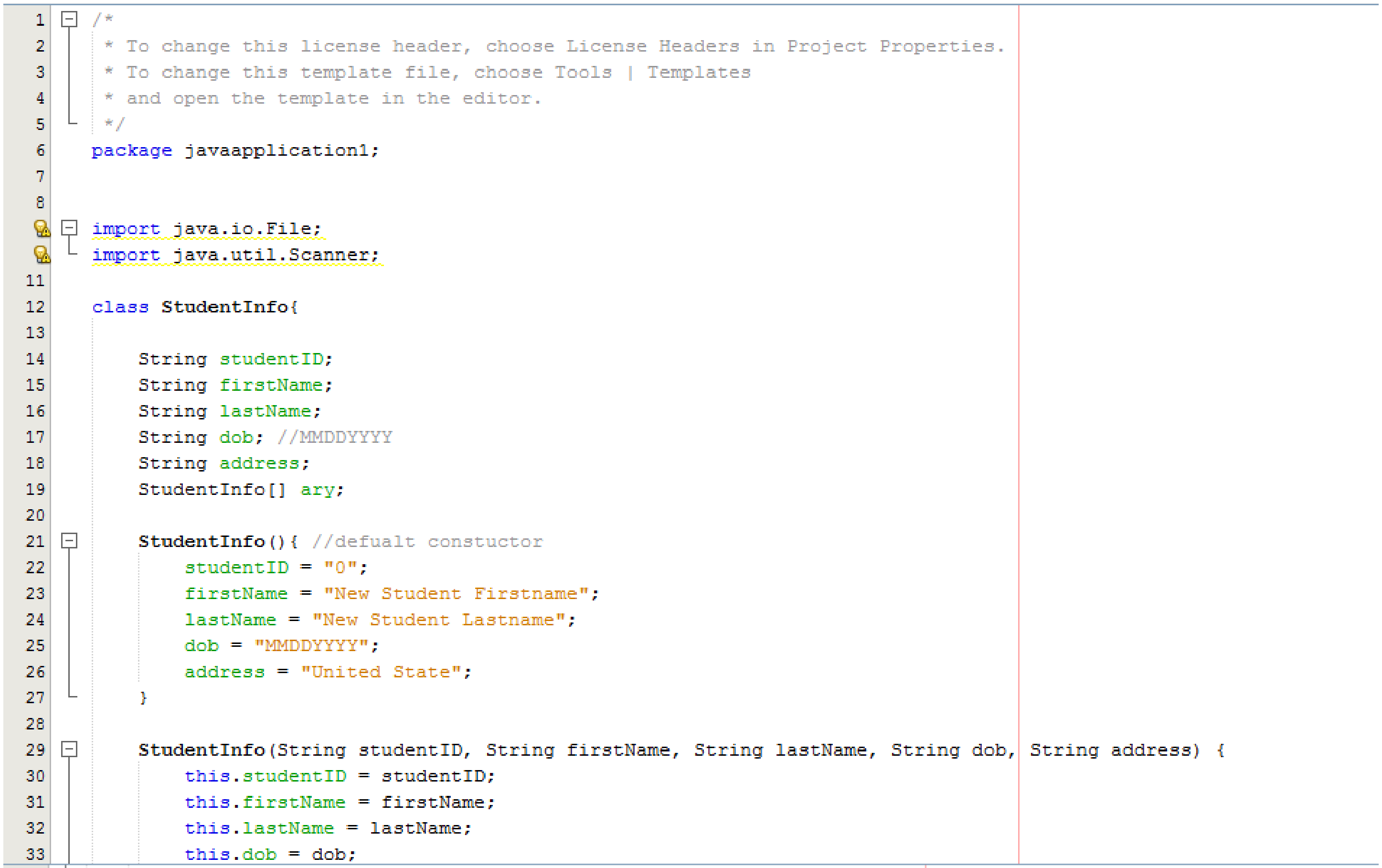
Output:

A Header Row

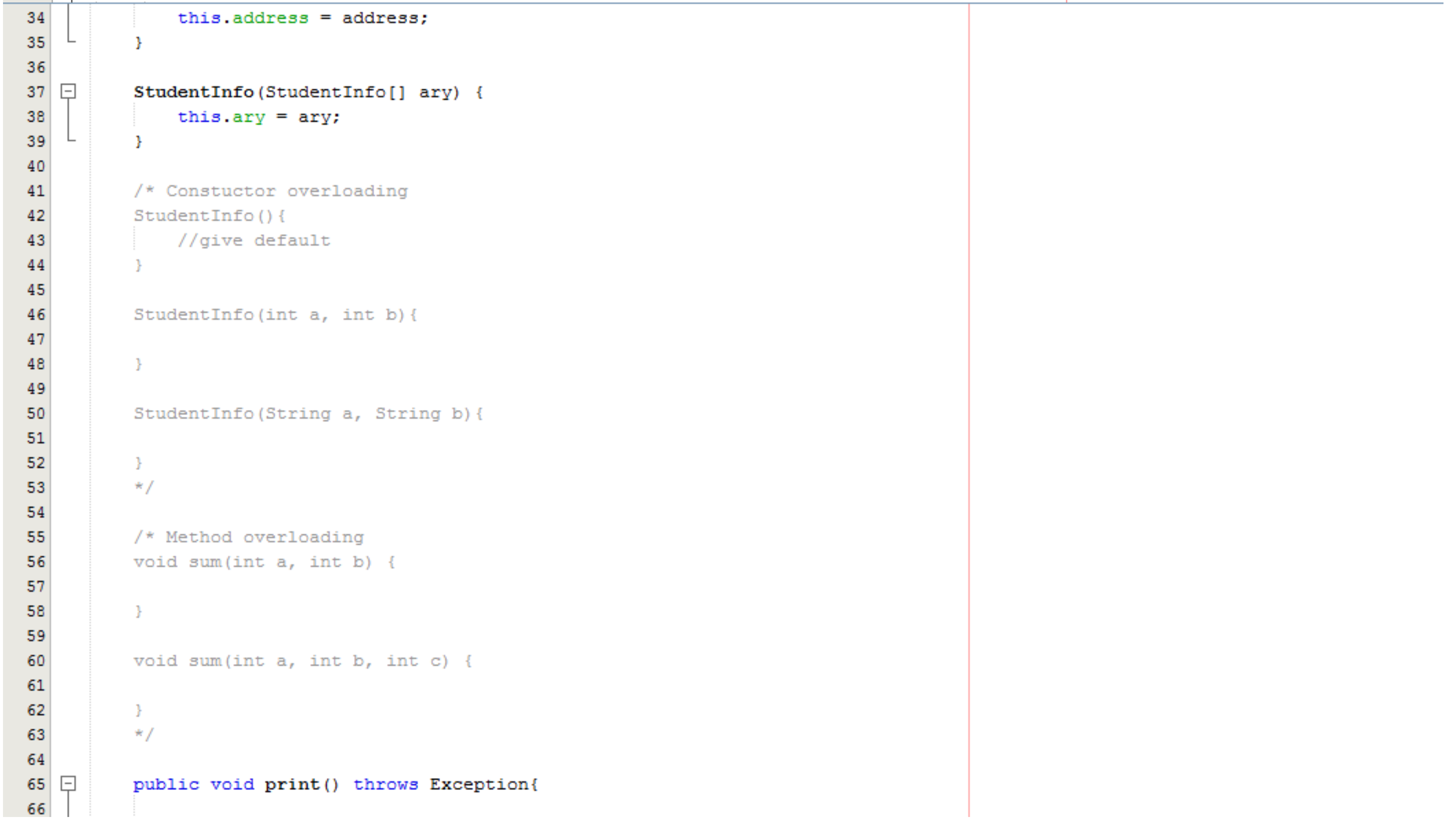
Then the 10 student objects in a .txt file

CIS 36B – 6th Class / Lab Assignment – **10 Points-**

Student Name **KaChiLau** **Student ID** **10819338** **Point Total**



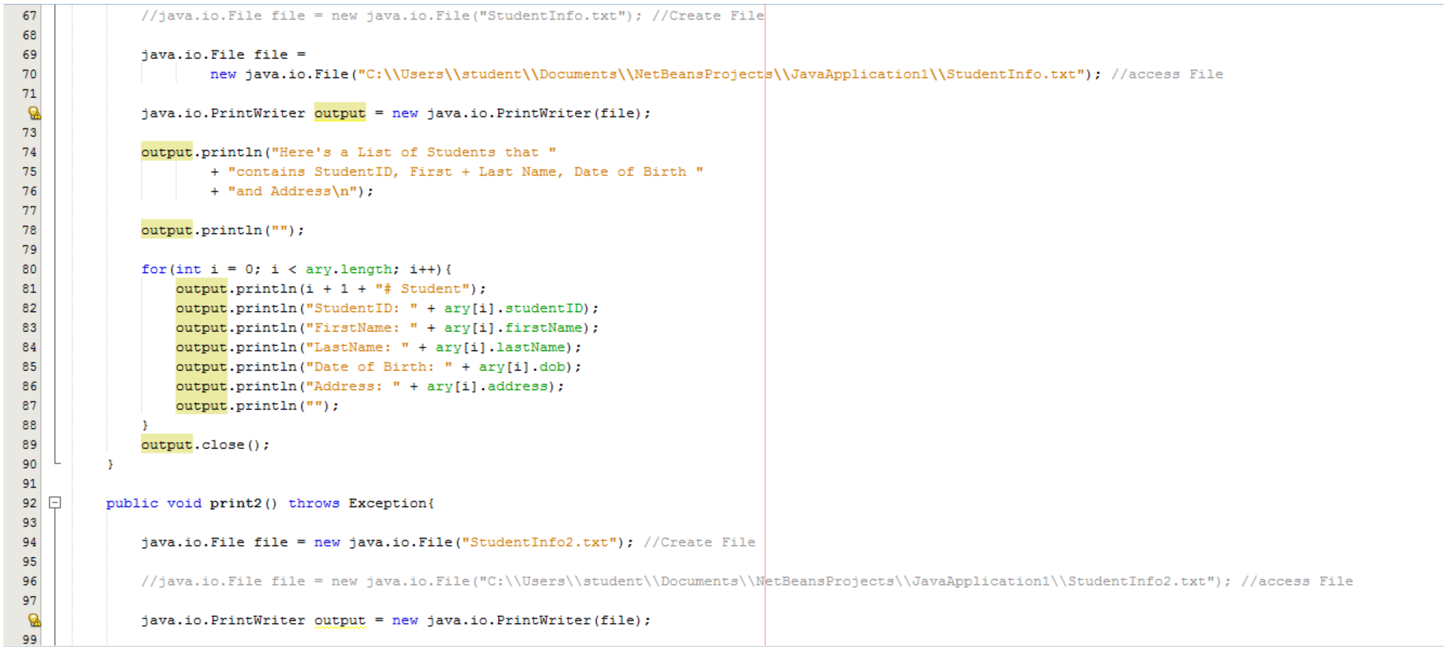
```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package javaapplication1;


import java.io.File;
import java.util.Scanner;

class StudentInfo{

    String studentID;
    String firstName;
    String lastName;
    String dob; //MMDDYYYY
    String address;
    StudentInfo[] ary;

    StudentInfo(){ //defualt constuctor
        studentID = "0";
        firstName = "New Student Firstname";
        lastName = "New Student Lastname";
        dob = "MMDDYYYY";
        address = "United State";
    }

    StudentInfo(String studentID, String firstName, String lastName, String dob, String address) {
        this.studentID = studentID;
        this.firstName = firstName;
        this.lastName = lastName;
        this.dob = dob;
```



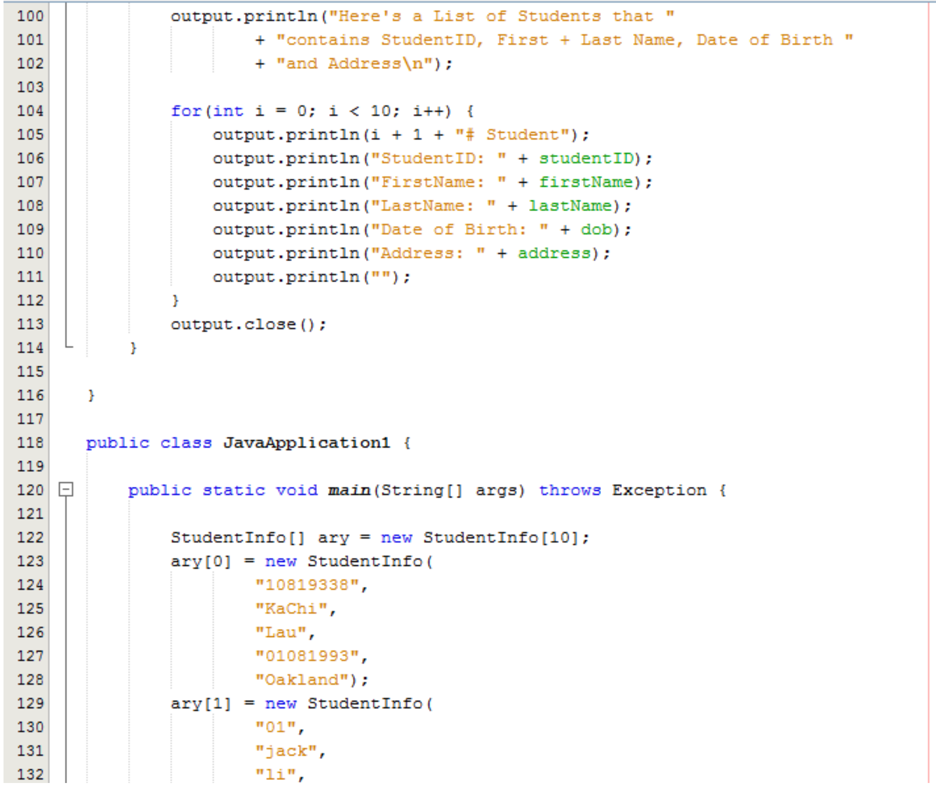
```
        this.address = address;
    }

    StudentInfo(StudentInfo[] ary) {
        this.ary = ary;
    }

    /* Constuctor overloading
    StudentInfo(){
        //give default
    }

    StudentInfo(int a, int b){

    }

    StudentInfo(String a, String b){

    }
    */

    /* Method overloading
    void sum(int a, int b) {

    }

    void sum(int a, int b, int c) {

    }
    */

    public void print() throws Exception{
```

CIS 36B – 6th Class / Lab Assignment – **10 Points-**

Student Name KaChiLau Student ID 10819338 Point Total



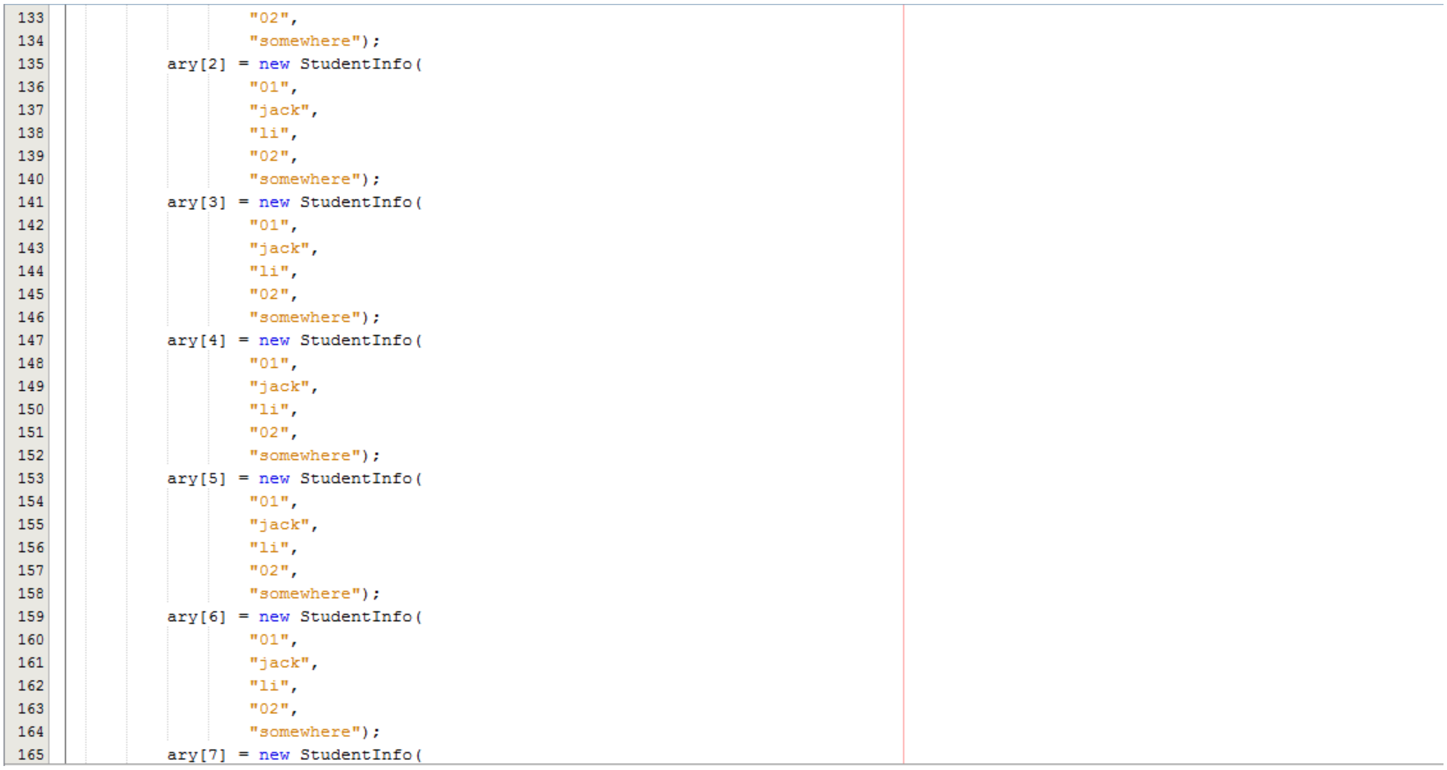
```
        //java.io.File file = new java.io.File("StudentInfo.txt"); //Create File

        java.io.File file =
                new java.io.File("C:\\Users\\student\\Documents\\NetBeansProjects\\JavaApplication1\\StudentInfo.txt"); //access File

        java.io.PrintWriter output = new java.io.PrintWriter(file);

        output.println("Here's a List of Students that "
                + "contains StudentID, First + Last Name, Date of Birth "
                + "and Address\n");

        output.println("");

        for(int i = 0; i < ary.length; i++){
            output.println(i + 1 + "# Student");
            output.println("StudentID: " + ary[i].studentID);
            output.println("FirstName: " + ary[i].firstName);
            output.println("LastName: " + ary[i].lastName);
            output.println("Date of Birth: " + ary[i].dob);
            output.println("Address: " + ary[i].address);
            output.println("");
        }
        output.close();
    }

    public void print2() throws Exception{

        java.io.File file = new java.io.File("StudentInfo2.txt"); //Create File

        //java.io.File file = new java.io.File("C:\\Users\\student\\Documents\\NetBeansProjects\\JavaApplication1\\StudentInfo2.txt"); //access File

        java.io.PrintWriter output = new java.io.PrintWriter(file);

```



```
        output.println("Here's a List of Students that "
                + "contains StudentID, First + Last Name, Date of Birth "
                + "and Address\n");

        for(int i = 0; i < 10; i++) {
            output.println(i + 1 + "# Student");
            output.println("StudentID: " + studentID);
            output.println("FirstName: " + firstName);
            output.println("LastName: " + lastName);
            output.println("Date of Birth: " + dob);
            output.println("Address: " + address);
            output.println("");
        }
        output.close();
    }

}

public class JavaApplication1 {

    public static void main(String[] args) throws Exception {

        StudentInfo[] ary = new StudentInfo[10];
        ary[0] = new StudentInfo(
                "10819338",
                "KaChi",
                "Lau",
                "01081993",
                "Oakland");
        ary[1] = new StudentInfo(
                "01",
                "jack",
                "li",
```

CIS 36B – 6th Class / Lab Assignment – **10 Points-**

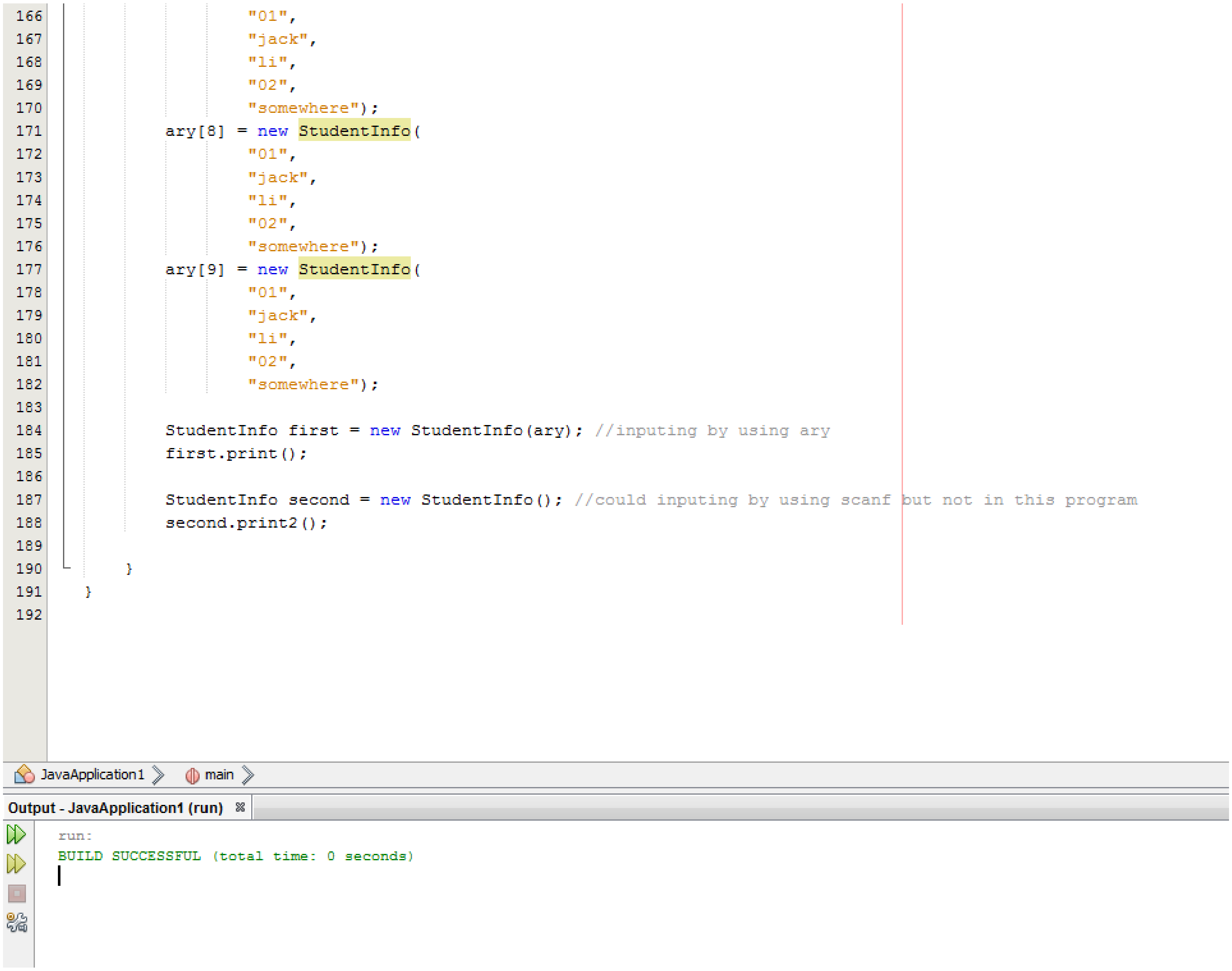
Student Name KaChiLau Student ID 10819338 Point Total



```
                "02",
                "somewhere");
        ary[2] = new StudentInfo(
                "01",
                "jack",
                "li",
                "02",
                "somewhere");
        ary[3] = new StudentInfo(
                "01",
                "jack",
                "li",
                "02",
                "somewhere");
        ary[4] = new StudentInfo(
                "01",
                "jack",
                "li",
                "02",
                "somewhere");
        ary[5] = new StudentInfo(
                "01",
                "jack",
                "li",
                "02",
                "somewhere");
        ary[6] = new StudentInfo(
                "01",
                "jack",
                "li",
                "02",
                "somewhere");
        ary[7] = new StudentInfo(
```

CIS 36B – 6th Class / Lab Assignment – **10 Points-**

Student Name **KaChiLau** **Student ID** **10819338** **Point Total**

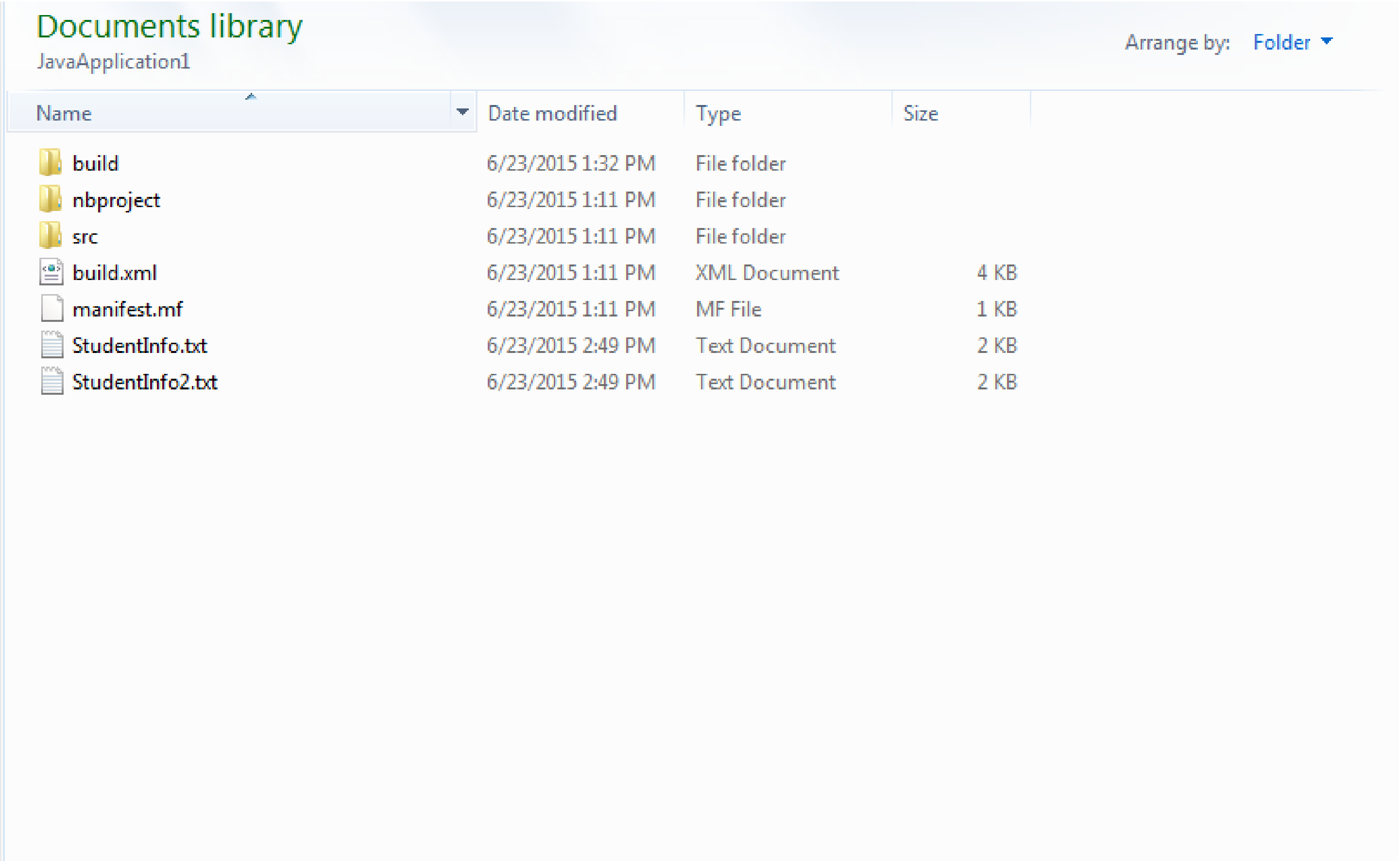


```
                "01",
                "jack",
                "li",
                "02",
                "somewhere");
        ary[8] = new StudentInfo(
                "01",
                "jack",
                "li",
                "02",
                "somewhere");
        ary[9] = new StudentInfo(
                "01",
                "jack",
                "li",
                "02",
                "somewhere");

        StudentInfo first = new StudentInfo(ary); //inputing by using ary
        first.print();

        StudentInfo second = new StudentInfo(); //could inputing by using scanf but not in this program
        second.print2();

    }
}
```

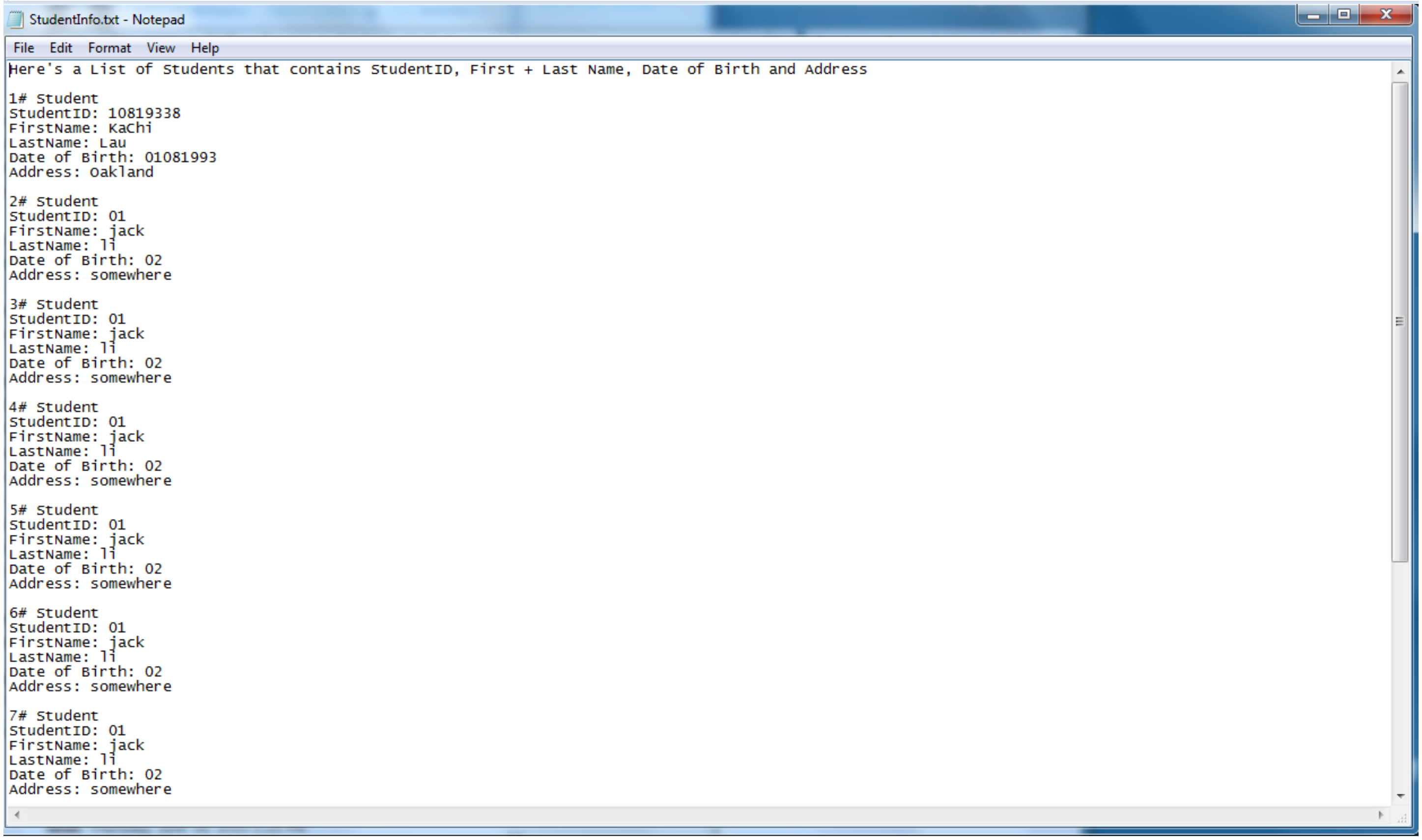
```
run:
BUILD SUCCESSFUL (total time: 0 seconds)
```

CIS 36B – 6th Class / Lab Assignment – **10 Points-**

Student Name KaChiLau Student ID 10819338 Point Total



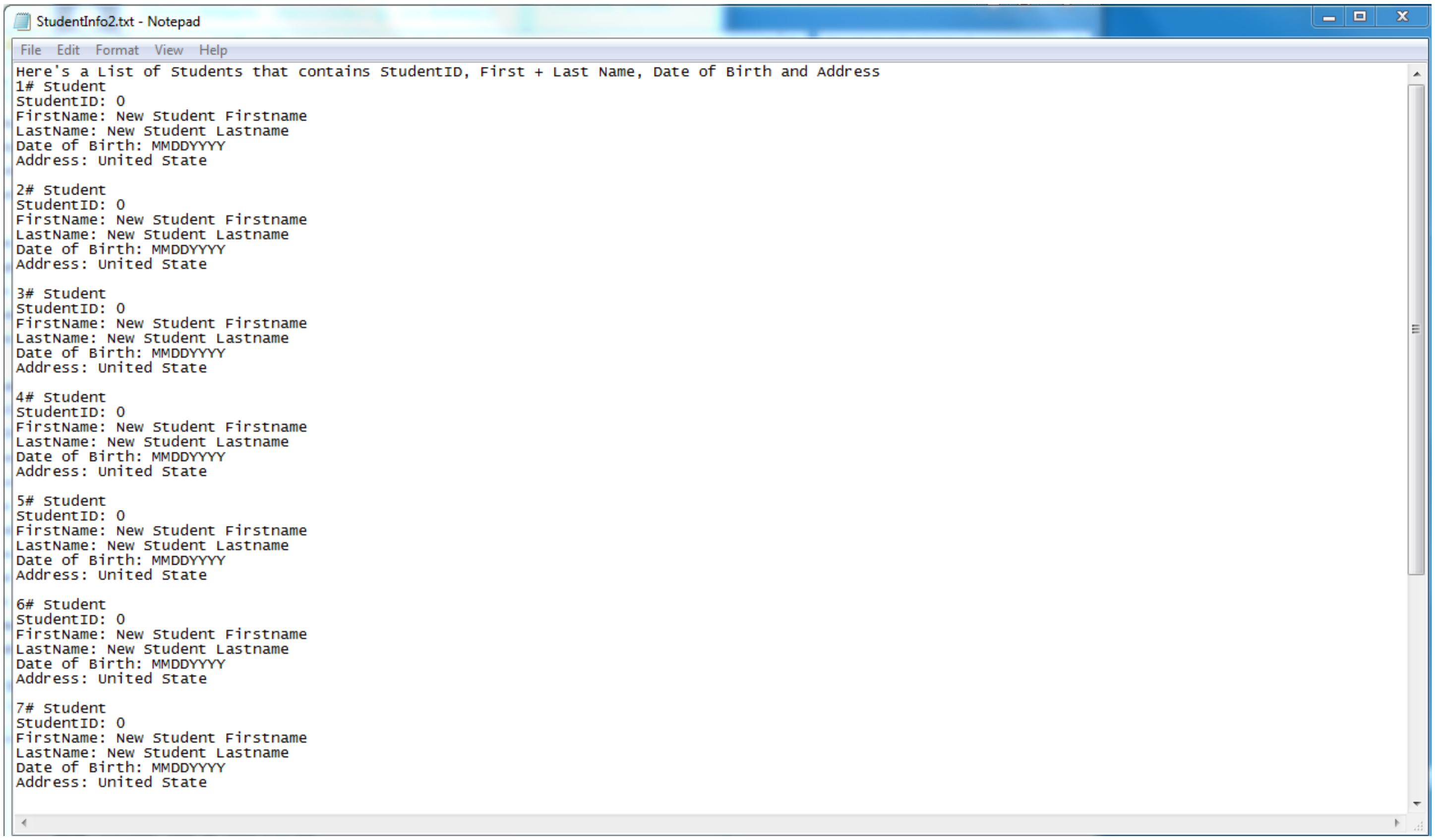
Name	Date modified	Type	Size
build	6/23/2015 1:32 PM	File folder	
nbproject	6/23/2015 1:11 PM	File folder	
src	6/23/2015 1:11 PM	File folder	
build.xml	6/23/2015 1:11 PM	XML Document	4 KB
manifest.mf	6/23/2015 1:11 PM	MF File	1 KB
StudentInfo.txt	6/23/2015 2:49 PM	Text Document	2 KB
StudentInfo2.txt	6/23/2015 2:49 PM	Text Document	2 KB



```
Here's a List of Students that contains StudentID, First + Last Name, Date of Birth and Address

1# Student
StudentID: 10819338
FirstName: KaChi
LastName: Lau
Date of Birth: 01081993
Address: Oakland

2# Student
StudentID: 01
FirstName: jack
LastName: li
Date of Birth: 02
Address: somewhere

3# Student
StudentID: 01
FirstName: jack
LastName: li
Date of Birth: 02
Address: somewhere

4# Student
StudentID: 01
FirstName: jack
LastName: li
Date of Birth: 02
Address: somewhere

5# Student
StudentID: 01
FirstName: jack
LastName: li
Date of Birth: 02
Address: somewhere

6# Student
StudentID: 01
FirstName: jack
LastName: li
Date of Birth: 02
Address: somewhere

7# Student
StudentID: 01
FirstName: jack
LastName: li
Date of Birth: 02
Address: somewhere
```

CIS 36B – 6th Class / Lab Assignment – **10 Points-**

Student Name _KaChiLau Student ID 10819338 Point Total



```
Here's a List of Students that contains StudentID, First + Last Name, Date of Birth and Address
1# Student
StudentID: 0
FirstName: New Student Firstname
LastName: New Student Lastname
Date of Birth: MMDDYYYY
Address: United State

2# Student
StudentID: 0
FirstName: New Student Firstname
LastName: New Student Lastname
Date of Birth: MMDDYYYY
Address: United State

3# Student
StudentID: 0
FirstName: New Student Firstname
LastName: New Student Lastname
Date of Birth: MMDDYYYY
Address: United State

4# Student
StudentID: 0
FirstName: New Student Firstname
LastName: New Student Lastname
Date of Birth: MMDDYYYY
Address: United State

5# Student
StudentID: 0
FirstName: New Student Firstname
LastName: New Student Lastname
Date of Birth: MMDDYYYY
Address: United State

6# Student
StudentID: 0
FirstName: New Student Firstname
LastName: New Student Lastname
Date of Birth: MMDDYYYY
Address: United State

7# Student
StudentID: 0
FirstName: New Student Firstname
LastName: New Student Lastname
Date of Birth: MMDDYYYY
Address: United State
```