```java
/*
 * This Is the Client of the Table Expense Calculator
 * This program will have 1 super class and 1 sub class
 * the super class will be the table creator
 * the subclass will have all the expense calculation methods and other methods
 *
 */
import java.util.Scanner;
import javax.swing.JOptionPane;
public class TableExpenseCalculatorClient
{
	public static void main(String[] args)
	{
		String tableSize ="", tableTitle="";
		Integer row = null ,col = null;
		TableCalculations table1 = new TableCalculations("Sample Table", 2, 3);
		System.out.println(table1);
		while (tableSize.length() ==0 || tableSize.length() <= 2)
		{
			tableSize = JOptionPane.showInputDialog(null ,
					"What size is the table?"
						+ "\neg... 2x2, 4x4, 5x6?\n"
						+ "Type the number of rows followed by
an 'x' with no space!", "Write table Size!" ,

	JOptionPane.PLAIN_MESSAGE).toLowerCase();
			tableTitle = JOptionPane.showInputDialog(null,
					"What is the title of the Table you are making",
					"Enter Title Here",
					JOptionPane.CLOSED_OPTION);
			String [] colrow = tableSize.split("x");
			if (colrow.length ==2)
			{
				row = Integer.valueOf(colrow[0]);
				col = Integer.valueOf(colrow[1]);
			}
			else
			{
				tableSize  = "";
			}// end if to check if use has correct number of "x"
		}// end while loop
		TableCalculations table0 = new TableCalculations(tableTitle, row, col);
// creates table
		table0.setValueIntAt(1, 1, 100);
		table0.setValueStringAt(1, 0, "100 at right");

		System.out.println(table0);// prints table

	}// end main
}// end client
```

```java
/*
 * This is the super class for the tableexpensecalculator
 *
 */
import java.util.ArrayList;
// table format
public class TableMaker
{
	// PIV
	private String title; // this is the main title of the table
	private int row;
	private int col;
	ArrayList<String>[][] ListString;
	ArrayList<Integer>[][] ListInt;


	// Constructor
	public  TableMaker()
	{
		title = "TitleHere";
		row = 2; // change default value here
		col = 1; // change default value here
		ListString = new ArrayList[row][col];
		ListInt  = new ArrayList[row][col];
		// automatically creates 2 x 1 table;
		for (int i =0; i< ListString.length ; i ++)
		{
			for (int j=0; j < ListString[i].length; j ++)
			{
				ListString[i][j] = new ArrayList<>();  // add arrayLis for
each row
				ListInt[i][j] = new ArrayList<>();            // add
arrayLis for each row
			}// Can only add values to Col....
		}
	}// end default constructor
	public TableMaker (String title, int row, int col)
	{
		this.title = title;
		this.row = row;
		this.col = col;
		ListString  = new ArrayList[row][col];
		ListInt  = new ArrayList[row][col];
		for (int i =0; i< row ; i ++)
		{
			for (int j=0; j < col; j ++)
			{
				ListString[i][j] = new ArrayList<>();  // add arrayLis for
each row
				ListInt[i][j] = new ArrayList<>();            // add
arrayLis for each row
			}// Can only add values to Col....
		}
	}
	// set title
```

```
	public void setTitle(String title)
	{
		this.title = title;
	}
	// AddMethod for String
	public void addRowString(int x, int y, String value)
	{
		ListString [x][y].add(value);
		ListInt [x][y].add(0);
	}
	public void setRowString(int x, int y, int position, String value) // user
must enter col and row, and then the Nth place they want to replace the item with

	{
		ListString [x][y].set(position, value);
		ListInt [x][y].set(position, 0);
	}
	// add method for int
	public void addRowInt(int x, int y, int num)
	{
		String number = "";
		number+= num +"|";
		ListString [x][y].add(number);
		ListInt [x][y].add(num);
	}
	public void setRowInt(int x, int y, int position, int num) // user must enter
col and row, and then the Nth place they want to replace the item with

	{
		String number = "";
		number+= num +"|";
		ListString [x][y].set(position, number);
		ListInt [x][y].set(position, num);
	}
	// remove method
	public void removeRow(int x, int y, int position) // user must enter the col
and row then the Nth place they want to remove.
	{
		ListString [x][y].remove(position);
		ListInt [x][y].remove(position);
	}
	/*
	public void addCol(int colToAdd)
	{
		col = col + colToAdd;
	}
		public void delCol(int colToDel)
	{
		col = col + colToDel;
	}
	*/
	// RemoveMethod
	// SetMethod
	//ArrayList Printer to Strings
	public String stringArrayList(ArrayList<String> list)
```

```
	{
		String fin="";
		for(String thing : list)
		{
			fin += thing +" ";
		}
		return fin;
	}
	// table printer
	private String printFinal(ArrayList[][] table)
	{
		String fin ="";
		for (int i =0; i< table.length ; i ++)
		{
			for (int j=0; j < table[i].length; j ++)
			{
				fin += stringArrayList(table[i][j]);
			}
			fin +="\n";
		}
		return fin;
	}
	// TablePrint method (toString)
	public String  toString()
	{
		return "\n" + title + "\n\n" +
				printFinal(ListString);
	}

}// end class
```

```java
/*
 * this is the sub class for Table Maker
 * this will calculate and do all the math stuff-
 */
public class TableCalculations extends TableCreator
{
	// constructor
	public TableCalculations ()
	{
		super();

	}
	public TableCalculations (String title, int row, int col)
	{
		super(title, row, col);

	}
	public int getTotal()
	{
		int tempTotal=0;

		return tempTotal;
	}
	public String toString()
	{
		return (super.toString() + "\n"
				+ "Total: " + getTotal() + "\n");
	}
}// end sub class - math
```

Program description:

This program generates a table with the size user wants.

User enters a row followed by an x then column (3x2, 3x5… etc).

The input is not case sensitive, but you cannot put spaces between. There is a split program that will split the letter x and separate the two strings. Those string will then convert to integer values.

```
            JOptionPane.CLOSED_OPTION);
   String [] colrow = tableSize.split("x");
   if (colrow.length ==2)
   {
        row = Integer.valueOf(colrow[0]);
        col = Integer.valueOf(colrow[1]);
   }
   else
   {
       tableSize  = "";
   }// end if to check if use has correct number of "x"
```

As shown above the two integers will be added to a Static Array with the number of splits (in this case it is two).

Then user can input letters or numbers by using the input command:

```
TableCalculations table0 = new TableCalculations(tableTitle, row, col); // creates table
table0.setValueIntAt(1, 1, 100);
table0.setValueStringAt(1, 0, "100 at right");

System.out.println(table0);// prints table
```

.setValueIntAt(int row, int col, int num)

.setValueString(int row, int col, String value)

When the table is filled out, user can choose to add all the integer values if necessary.

TableCalculations program is a subclass for TableMaker. It has all the Math tools needed for the table.

```java
public class TableCalculations extends TableCreator
{
    // constructor
    public TableCalculations ()
    {
        super();

    }
    public TableCalculations (String title, int row, int col)
    {
        super(title, row, col);

    }
    public int getTotal()
    {
        int tempTotal=0;

        return tempTotal;
    }
    public String toString()
    {
        return (super.toString() + "\n"
                + "Total: " + getTotal() + "\n");
    }
}// end sub class - math
```

Purpose of the program:

The purpose of this program is to develop knowledge about super class and sub class.

Convert String to Integer and be familiar with splitting texts and using the number of texts split as the array size.

Be familiar with 2D static array and iterate through them.