|  |  |  |  |
| --- | --- | --- | --- |
| **Instructor** |  | **Due Date** |  |

**PROJECT Decision Controls and the Java Language 50 points**

**Objective** To write a Java program involving a **switch** statement.

***PROJECT DESCRIPTION***

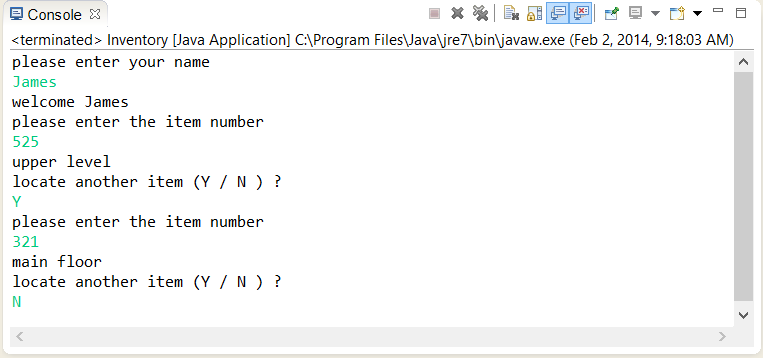
The location of inventory items in the various floors of a downtown music superstore are classified according to their inventory numbers, as shown in the table below.

Use a **switch** statement to write a program that requests the inventory number of an item in, the music superstore, as input and displays the location of the item.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Inventory Number** |  | **Location** |
|  | 100 to 199 |  | lower level |
|  | 200 to 499 |  | main floor |
|  | 500 to 999 |  | upper level |

A screen snapshot of the initial program code in use is shown below.

**Inventory Program Screen Snapshot**



***Information about This Project***

This particular program illustrates an example of a Java **switch** statement. Such a statement is used in our program in the style given below.

**switch(intLocate) {**

**case 1: System.out.println("lower level"); break;**

**case 2: System.out.println("main floor"); break;**

**case 3: System.out.println("upper level"); break;**

**default: System.out.println("invalid number"); break;**

**}**

***Steps To Complete This Project***

**STEP 1**  **Open Eclipse or JCreator on Your Computer**

Open the Eclipse or JCreator Java text editor. Click File and choose New to create and save a new file and name it as: **Inventory.java**

Within the **Inventory.java** file, type only the Java language program code given below. Insert your own name in place of " Sammy Student " .

**PROJECT Decision Controls and the Java Language**

**Figure 1 Java Language Source Code**

|  |
| --- |
|  |

|  |
| --- |
| **/\* Programmer: Sammy Student, filename: Inventory.java \*/**  **import java.io.\*;**  **// the class definition**  **public class Inventory**  **{**  **// declare a BufferedReader object**  **static BufferedReader kb = new**  **BufferedReader(new InputStreamReader(System.in));**  **public static void main(String args[]) throws IOException**  **{**  **// declare the variables that are local to main()**  **int itemNum = 0, intLocate = 0;**  **String strName = " ";**  **char again = 'Y';**  **// request and receive data from the user**  **System.out.println("please enter your name");**  **strName = kb.readLine();**  **System.out.println("welcome " + strName);**    **// enter a looping structure**  **while (again == 'Y')**  **{**  **System.out.println("please enter the item number");**  **itemNum = Integer.parseInt(kb.readLine());**  **// enter a group of decision statements**  **if (itemNum >= 100 && itemNum <= 199) intLocate = 1;**  **if (itemNum >= 200 && itemNum <= 499) intLocate = 2;**  **if (itemNum >= 500 && itemNum <= 999) intLocate = 3;**    **switch(intLocate)**  **{**  **case 1: System.out.println("lower level"); break;**  **case 2: System.out.println("main floor"); break;**  **case 3: System.out.println("upper level"); break;**  **default: System.out.println("invalid number"); break;**  **}**  **intLocate =0;**  **System.out.println("locate another item (Y / N ) ?");**  **again = kb.readLine().charAt(0);**  **if (again == 'N') { System.exit(0); kb.close(); }**  **}**  **}**  **}** |
|  |

|  |
| --- |
|  |

**PROJECT Decision Controls and the Java Language**

After you type the above code into your Java file, save the file.

Note: since the Java class within the code is named **Inventory** , your file name must be also named as **Inventory.java** .

**STEP 2**  **Build, Compile the Program and Run the Program**

Once you have successfully compiled your program, run your program and enter your own name when prompted and enter an inventory number and observe the program’s output.

**STEP 3 Return to Your Java Code Editor and Modify the Program Code**

You will now modify the existing Java language source code for the

**Inventory.java** program. Alter the source code such that the inventory numbers are classified according to the following chart.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Inventory Number** |  | **Location** |  |
|  | 1000 to 1999 |  | lower level |  |
|  | 2000 to 5000 |  | main floor |  |
|  | 5001 to 9000 |  | upper level |  |
|  | 9001 to 9999 |  | mezzanine |  |

For your modified program also include a block of code that indicates that inventory items 1621 , 2175 and 9876 are no longer available for sale.

Save your modified program.

**STEP 4**  **Compile and Run the Program**

In a fashion similar to **STEP 2** above, compile and test your **Inventory.java** program. When running the program, enter your own name when prompted. Enter an inventory number and observe your output.

**STEP 5**  **Take Screen Snapshots**

Take screen snapshots of the complete operation of your **Inventory.java** program when you enter an inventory number from each of the above locations as well as a number greater than the stated numbers, i.e. 9999 . Also use at least one the three items that are no longer available.

Paste and label these snapshots including your program code, in a professional manner, in a word processing document. Submit the document for credit.