Informatics College Pokhara

Emerging Programming Platforms and Technologies

CS5004NP

Group Coursework 1

Submitted By:	**London Met ID**	**Submitted To:**
Kushal Gurung	18029022	Mr. Sandip Adhikari
Sanjit Adhikari	18029085	Module Leader
Sadikshya Baral	18029069	Emerging Programming
Hrithik Sunar	18029017	Platforms and
Sagar Nepali	18029075	Technologies

Group: L2C3

Date: 14-Jan-2020

Proposal

This coursework has been assigned to us as a group coursework composed of five members. In this project, we were assigned to create a ‘Car Accessories Information System’ in Java Programming Language using NetBeans application with the additional concept of binary search algorithm. In this project, we will be creating a Graphical User Interface (GUI) application which allows users to add car accessories details, search car accessories either from category or from price. For searching car accessories details through price option, binary search algorithm will be used. The main motive of the project is to learn and use the knowledge of Java programming to create and design a significant graphical user interface.

Purpose

The purpose of this project was to develop an Information System for “Car Accessories Details” with the help of GUI elements like frame, panel, table, radio buttons, combo box, check box, text fields and many Java’s pre-defined library classes to make it attractive and responsive. In this project, the system have the functionality to search items based on price and category. For searching accessories based on category, user can easily search car accessories easily by selecting options from category combo box. Similarly, for searching items based on price, a text filed is present where user will have to enter the amount to search for car accessories that have the same entered price. If two or more car accessories have the same price, then only the first matching car accessories will be displayed in a JOptionPane message box and in case if there’s no car accessories for the entered price, the JOptionPane message box will display meaningful message. For searching accessories based on price, binary search algorithm will be used.

Aims and Objectives

This project was aimed to develop an application that can be very much user-friendly, interactive and helpful to keep records of the data of car accessories. The aim was to gain as much knowledge and ideas about the GUI elements and different pre-defined classes and use them to make the car accessories information system functional.

Proposed Approach

The proposal were approached through following task:

a. Apache NetBeans software was used for all the coding and designing of the Car Accessories Information System.
b. The information system was made through frame, panels, menu bar, menus, menu items, table, text fields, labels, combo boxes, radio buttons, check boxes, buttons, sorting method, binary search algorithm and many pre-defined java classes.
c. After the information system was developed, various testing were done in order to make the system function properly.

Target Audience

Once the car accessories is developed, it can be used by students, researchers and programmers related to the field of java programming language. The system also targets the current shops related to car accessories to make it easier for them to update and view all the car accessories details in the table to make their working environment safe and effective.

1. Title (Car Accessories Details Information System)

Car Accessories Details Information System is a system where accessories details of car are stored. In this system, user can also add accessories details, search the accessories either by entering price or by selecting a category. Then the system will display the accessories details based on the searching of user. At first, CAD_IS project is created and a JFrame named "CAD_Info" is created in it. Then inside the CAD_Info JFrame, the car accessories information system is developed with the help of panels, menu bar which consists of menus and menu items to open file, view help, know about the developers names and exit the application functions, similarly, table, text fields, labels, radio buttons for range level, check boxes for recommendations, combo box for selecting categories, add button, clear buttons and exit button, sorting method and many pre-defined java classes.

The coursework also includes the use of binary search algorithm. Basically, binary search algorithm is a search algorithm that finds the position of a target value within a sorted array. In our project, binary search algorithm is used to find and display the accessories details based on the price which is searched by the user.

2. Brief Information about Car Accessories Information System

2.1 List of data:

There are seven types of data used in developing Car Accessories Details Information System which are as follow:

- ID: This field displays the id number for all the car accessories stored in the JTable. There are 15 accessories already in the table which are given ID numbers from 1 up to 15. So, when a new accessories details is added in the table, that accessory will automatically have the ID number 16 and the id number will increase by 1 based on the increment of new accessories.
- Category: This field is a category of car accessories with different options. In our application, JComboBox is used to display 5 types of category. They are: Comfort, Ambience, Media, Sensors and Gauge & meters.
- Accessories Name: This field in a car accessories information system is used to store the name of the car accessories.
- Accessories Number: This field holds the accessories number of the car.
- Range Level: This field describes the range of the price of car accessories. JRadioButton is used for range level with 3 options which are low, medium and high.
- Price: This field is used to store the price of the car accessories.
- Recommendation: This field describes the recommendation of the car accessories. For entering data into this field, three jCheckBox are used, They are Company, Community and No recommendation. User can select company and community at the same time. But, when no recommendation is selected, s/he cannot select other options.

2.2 List of features:

- The system is user friendly and interactive. Any user can easily use this system without an issue. It shows different kinds of message dialog boxes to the user when user performs any specific task.
- Open menu item is present in the menu bar which helps to open any item from desktop interface.
- View help menu item is also present in the Help menu which guides the user on how to use the Car Accessories Information System.
- User can add and delete the accessories details in the in table.
- User can search accessories by entering price amount in the text field. The system will display the name of accessory in the message dialog box who has the same price. If two or more accessories have the same price, then the first matching car accessory name will be displayed.
- User can also search accessories by choosing a category from the combo box. Once, a category is selected, the list of car accessories will be displayed in a message dialog box.
- Input fields in the system do not allow faulty data to be entered. If wrong data is entered, error message will be displayed in a message dialog box.
- Multiple Testing with different test cases are done to guard the system from various bugs.
- The system is easy to understand and ready to use.
- The system is light and responsive and can run in any computers regardless of its specifications.

3. Tools used for the project along with their justification:

✓ Apache NetBeans

Figure 1: Apache NetBeans

NetBeans is one of the most popular integrated development environment (IDE) for java. It is an open source project which is easy to use, supports development of java application and also creates friendly easily configurable development environment. It is highly interactive as it highlights the errors and also shows the reasons for the error. Just after the installation, one can easily create meaningful applications.

NetBeans was the most important tools that we used for this project because it is the tool from where we developed our Car Accessories Information System. It suggested different pre-defined library classes which we used on our project for different purposes. It helped us in correcting our run time errors or any errors in our codes by displaying a red exclamatory sign on the line which contained the error. The designing of the system was very simple with the help of drag and drop feature of Apache NetBeans.

✓ Microsoft Word

Figure 2: Microsoft Word

Microsoft Word is a word processing program used to create documents. It is a very popular word processor developed by Microsoft. Microsoft Word is very user-friendly, we can easily change our font type, size, colour according to our choice. We can also attach different kinds of file extensions which helps to makes our documentation attractive. Documentation formatting, providing captions in figures and tables, creating header and footer are all very easy in the Microsoft Word, so for these reasons, the documentation of our coursework was written in the Microsoft Word.

Table of Contents

List of Figures

List of Tables

1. Introduction

The group coursework was assigned to us on 3rd week where each group of five members were assigned to develop a Car Accessories information system. The coursework was aimed to increase our capability to work in a group and utilize all the knowledge gained till date on Emerging Programming Platforms and Technologies to build a Car Accessories Information System. On our project, a Car Accessories Information System is developed with the help of Apache NetBeans software. The information system that we developed is very much functional, attractive, user friendly and was tested several times through different cases to remove the bugs.

For the development of our project, Apache NetBeans application was used for coding, designing and testing. Generally, Apache NetBeans is an open source, Java based integrated development environment that allows developers to create mobile, desktop, and web applications with multiple languages including Java, CSS, PHP, HTML, and C++. This IDE is a cross-platform architecture which means that this IDE can run on Windows, OS X, Linux, and other UNIX based Operating systems. Apache NetBeans was the pillar for our project's development, since the whole car accessories information system was developed on it. First, a JFrame CAD_Info was created in a project CAD_IS. Then, all the designing and coding was done in the CAD_Info JFrame. Our Car Accessories Information System will be able to add new accessories details in the table after all the details are filled in the text fields and range level, category, recommendations are selected. Similarly, the system has clear button which will clear all the previously entered and selected values in text fields, radio buttons and check boxes. User can also search for accessories based on price or category. Binary search algorithm is used for price searching. Similarly, user can remove all the information in the table by clicking on clear all button and can also remove a selected detail by clicking on Clear Selected button. Also, system has open menu item to open any file from desktop interface, exit menu item and exit button to close the application, View Help menu item to obtain information on how to run the Car Accessories Information System and About menu item to know about the version and developers of the Information System.

We know that the traditional method of storing all the details of the car accessories in hard copies are not easy to manage and are difficult to retrieve the details in short time if there are many items in the table. For that, our information system will help to replace the traditional method and help all the car accessories shops to add, delete search and store the details of their car accessories digitally in a very short period of time. Our system also provides protection and security to the car accessories details stored in the system and will help to improve the maintenance of the car accessories details effectively.

2. Individual Task

For the development of the project, we split the project into 5 tasks and each member was assigned with a task. After the project was split to 5 tasks, each member were provided their responsibilities and were assigned to get the work done. The table below displays the tasks assigned to each members:

Member Name	Assigned tasks
Kushal Gurung	Creation of table, addition of all buttons, menu items, testing the functionality of all buttons and menu items, report writing.
Sanjit Adhikari	Creation of text fields, combo boxes, check boxes and their testing to making them functional and responsive, report writing.
Sadikshya Baral	Sorting, swapping and Binary searching method, report writing.
Hrithik Sunar	Designing of the system, testing the system after completing the designing, report writing.
Sagar Nepali	Testing over all functionality of the system, report writing.

Table 1: Individual Task

Even though separate tasks were assigned to the respective members, the work was done as a unit because each task of one member was related to the task of another members. Therefore, working together helped to develop the system smoothly where ideas and knowledge were also shared amongst the group members which contributed greatly for finishing the development of the project on time. The most difficult part of the project was binary searching method and price searching button where we had to try different codes and search for ideas on internet to find the solution.

In this way, we helped each other to finish each other's respective individual task because finishing everyone's individual task means finishing the whole group task.

3. Binary Search Algorithm

Binary search is a divide and conquer approach to search a sorted array from a particular value. It is used to search a key element from multiple elements. In case of binary search, array elements must be in ascending order. Comparatively, binary search is faster than the linear search. This algorithm will be more efficient only if the list of items is sorted else the only solution will be linear search.

In case, there is an array of 1000 elements, the comparison must be done in worst case. However, in our coursework binary search looks for the items based on the price in the ‘Car Accessories Information System’.

The binary search algorithm searches for a number in a pre-sorted array by determining the mid-point of the array and if the element is not the required number, then it proceeds to check the smaller array on the left or right of midpoint depending on whether the element is smaller or larger. In our code, the binary Search function have exactly two parameters, i.e.: the array to be searched and element to search for. For example:

In our code, key variable is a element to search for in the list. Firstly, the mid-point (mid) of the list is determined and the element at index mid is compared to ‘key’ element to find out if the target is in the first or second half of the list. In our case, the mid point “mid” is determined with the help of two other variables. The variable “low” denotes 0 and the variable “high” denotes the total size of the list -1. The mid value can be found as:

Here, we terminate first half of the list considering the targeted number is greater than midpoint element in the list with just one comparison and accordingly the other half is also tested unless we find the target or check if it is not in the list (tutorialspoint, 2020).

In-order to find the middle of the list, we need three variables:

- ⇨ one to identify the beginning of the list, i.e. low
- ⇨ one to identify the middle of the list, i.e. mid
- ⇨ one to identify the end of the list, i.e. high

Therefore, the mid value is found as:

⇨ mid=(low+high) / 2

Importantly, we analyse two cases here either the target is in the list or the target is not in the list.

In the figure given below, the successful testing of binary search is shown:-

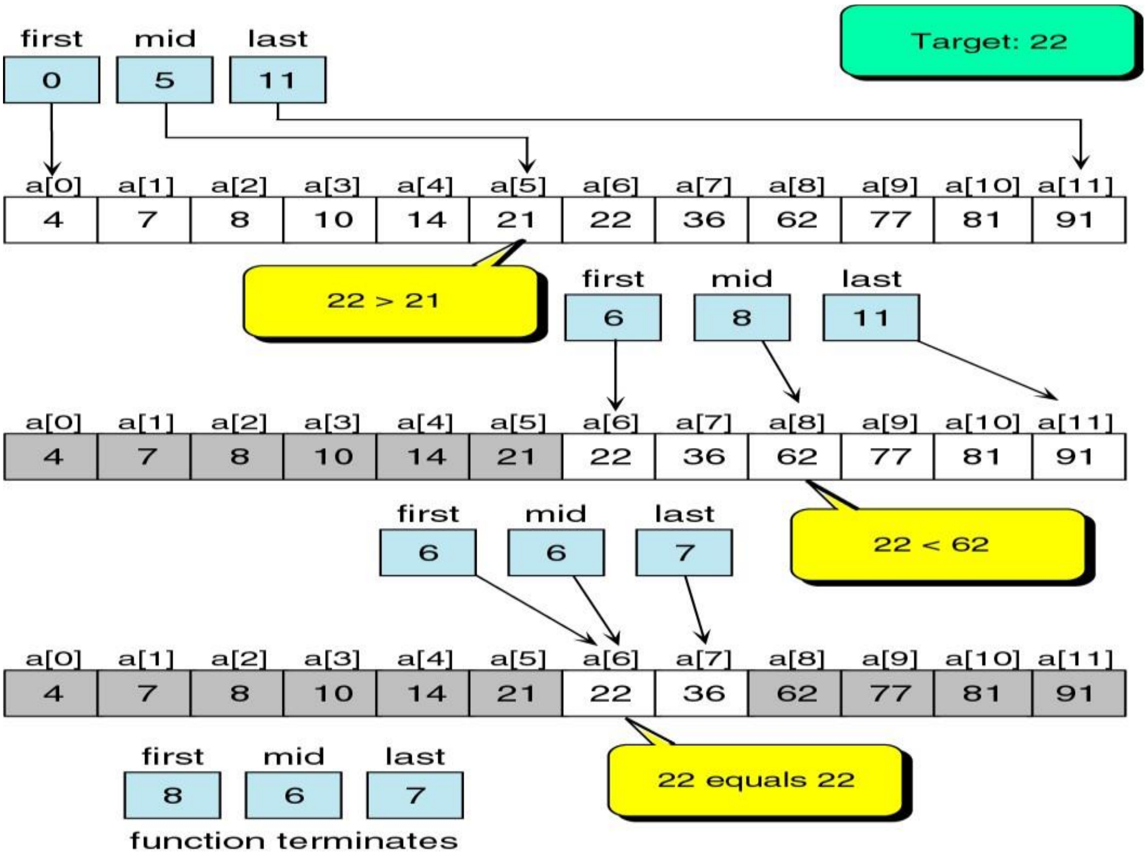


Figure 1: Successful search of Binary search

3.1 Use of Binary Search

The main propose of Binary Search Algorithm in our project was to search and display the correct car accessories when we search the item from the price text field. If we enter certain price in the text field, let us suppose 350 is our key value which is highlighted in green colour, then following procedure will take place:

Price	100	150	200	250	300	350	400
Index	0	1	2	3	4	5	6

Table 2: Binary Search Algorithm step 1

Here, we have 7 elements are in the list, then binary search algorithm first calculates the mid value with sum of low index and high index of the list and divides them by 2. In this scenario,

Mid value (mid) = (0(first index) + 6(last index)) / 2 = 3

Therefore, our mid value will be a[3]= 250.

Amount	100	150	200	250	300	350	400
Index	0	1	2	3	4	5	6

Table 3: Binary Search Algorithm step 2

Since the blue highlighted part a[3] has price 250 which is not equal to 350. Since, 250 < 350, it now searches the right half. For that, the value of low variable will be increased by 1 as low = mid + 1 and mid value will now be:

mid = 4+6 / 2 = 5.

Therefore, the element in our mid value is a[5] = 350.

Amount	100	150	200	250	300	350	400
Index	0	1	2	3	4	5	6

Table 4: Binary Search Algorithm step 3

Hence, the targeted price is equal to mid value i.e. 350. So, the binary search was successful so now it will return key element that is 350. In the worst case, if the element was not found then it will return “-1”.

The same procedure is applied in our code. When a price is entered in the text field, it will search and display the accessory which has the same price amount. If in case, there are two accessories with the same price, the first matching accessory will be displayed in a message dialog box.

3.2 Pseudo code

```
binary Search(int [] a, key){    // [] a is the array where

low= 0

high = size - 1            // size = total element in the list.

while(low<=high){
        mid=(low+high)/2
        IF (a[mid] == key)
                return a[mid]
        ELSE IF (key > a[mid])
                low = mid +1
        ELSE
                high = mid-1;
        END IF
}
return -1
}
```

3.3 Algorithm

Step 1: Start

Step 2: Initialize array “a”

Step 3: Input the key element to be searched for in the list

Step 4: Initialize variable low = 0 and high variable = size -1 // size is the total element in the list.

Step 5: While low <= high, Go to step 6 else Go to step 9.

Step 6: mid = (low + high) / 2

Step 6: If a[mid] == key, go to step 8 else go to step 7

Step 7: If key > a[mid]

Step 8: low = mid +1 else high = mid -1, Go to step 6

Step 8: Return a[mid] and Go to Step 10.

Step 9: Return -1

Step 10: End

3.4 Flowchart

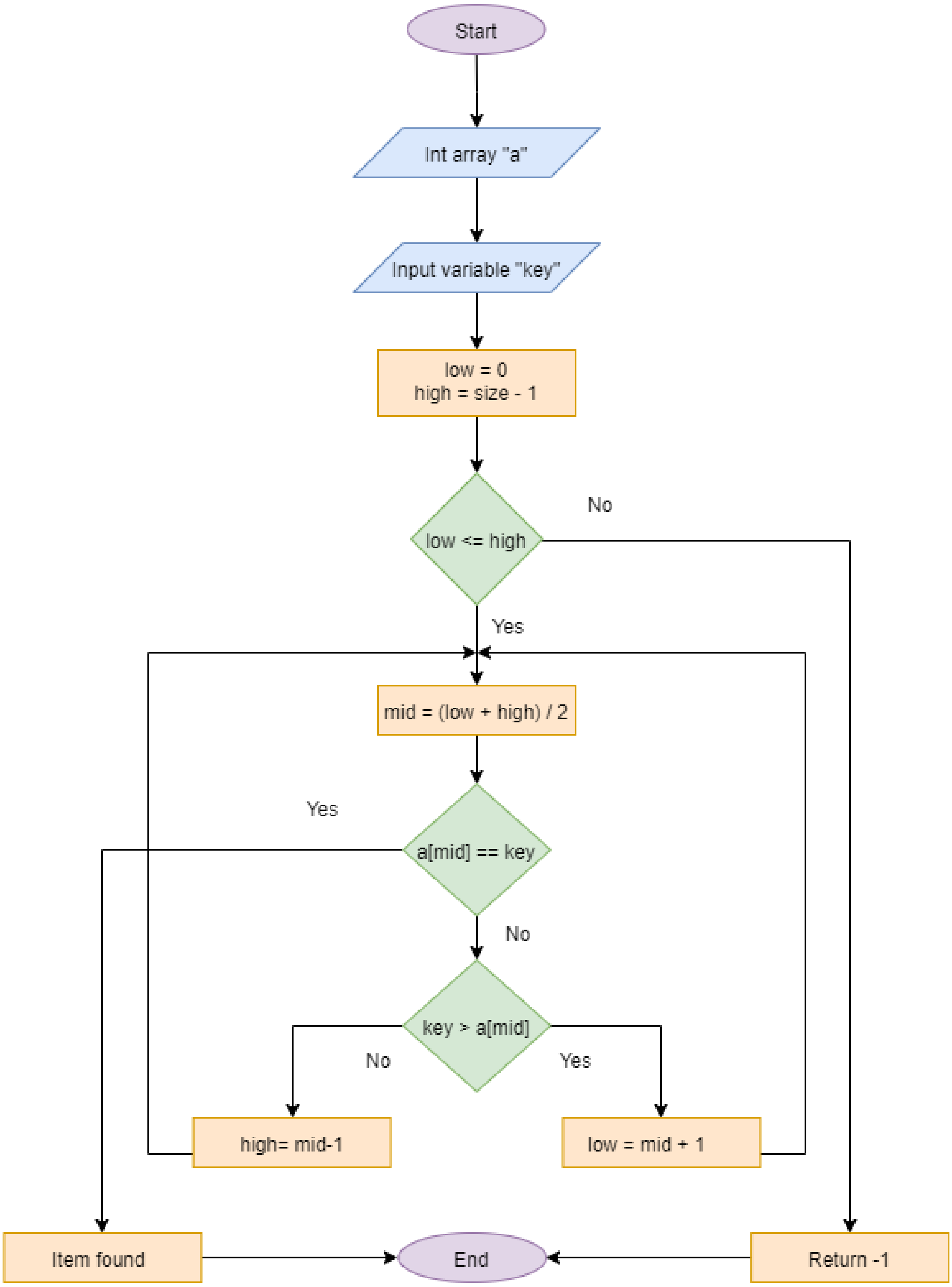


Figure 2: Flow chart of the Binary Search Algorithm

4. Method Description

The information system was developed with the help of many methods. They are:

a) CAD_Info():

This is a special type of method also known as the constructor. Inside this method, the initComponents() method is called which is generated at the back end when we use the drag and drop components. Similarly, multi-dimensional array of Object data type is declared and initialized with 15 rows and is added to the table with the help of DefaultTableModel class which is an implementation of the table model. At last, a titled border is also created for the table and was named “Car Accessories Details” and the font colour of the titled border is set to white.

b) sort(int [] a):

This method will sort the value of an array which is passed as an argument in this method's parameter. In our case, this method will sort the price of all the car accessories. It is a parameterised method where an array “a” of int data type is declared as a parameter. It is an array which is to be sorted when an argument will be passed in this method. Then it calls the method minimumValue first and then calls swap method under a for loop.

c) minimumValue(int [] a, int form):

This method is used for finding the minimum value. It compares with a forward number and if it is smaller than the previous number, the smallest value will be stored in the minVal position.

d) swap(int [] a, int i, int j):

This method is used for swapping the position of the minimum value and the greater value if the greater number is positioned ahead of the minimum value.

e) binarySearch(int [] a, int key):

This method first calls the sort method to sort all the accessories. After that, mid value of the array is calculated through two variables low and high. The variable low is assigned with value 0 and high is assigned with value (size-1) where size is the length of the array. If the mid value is equal to the key value, the key value will be returned, if key value is greater than mid value, the value of low variable will be assigned with mid + 1 or else if the key value is less than the mid value, the variable high will be assigned with mid – 1. The process will repeat until the key element == a[mid].

f) communityCheckBoxActionPerformed(java.awt.event.ActionEvent evt)

When community check box is selected in the recommendation option, Ro Recommendation option will be set to false.

g) exitButtonActionPerformed(java.awt.event.ActionEvent evt)

This method helps to close the information system when Exit button is clicked.

h) addButtonActionPerformed(java.awt.event.ActionEvent evt)

This method helps to add accessories details in the table and also shows message dialog box, once the accessories is added in the table. It also shows error messages when certain text fields are left empty or when value of wrong data type are entered in the text field.

i) noRecommendationCheckBoxActionPerformed(java.awt.event.ActionEven t evt)

When No Recommendation check box is selected in the Recommendation option, the Company and Community check boxes will be set to false.

j) companyCheckBoxActionPerformed(java.awt.event.ActionEvent evt)

When Company check box is selected in the Recommendation option, No Recommendation check box will be set to false.

k) priceSearchButtonActionPerformed(java.awt.event.ActionEvent evt)

This method is used to search for an accessory based on price searching. Since our price column is at the 5th index, a variable data [i] will store the value of 5th index. Similarly, variable searchItem stores the value which will be entered in the price text field for searching. Then, these two variables data and searchItem will be passed as arguments to the binarySearch method which will then search for the accessories with the same price which was entered in the text field for searching.

l) categorySearchButtonActionPerformed(java.awt.event.ActionEvent evt)

This method is used to search for an accessory based on category selection. Once, a category is selected for example: Comfort, a message dialog box will appear which will display all the accessories that lies under Comfort category.

m) exitMenuItemActionPerformed(java.awt.event.ActionEvent evt)

This method helps to close the information system, when the Exit menu item is clicked from the File menu.

n) openMenuItemActionPerformed(java.awt.event.ActionEvent evt)

This method helps to provide a file chooser open dialog box for user to select and open any files from desktop.

o) aboutMenuItemActionPerformed(java.awt.event.ActionEvent evt)

This method displays the version and the developer names of the Car Accessories Information System in an information message box.

p) viewHelpMenuItemActionPerformed(java.awt.event.ActionEvent evt)

This method guides the user on how to run and use different functions of the information system by displaying guidelines in a plain message box.

q) clearButtonActionPerformed(java.awt.event.ActionEvent evt)

This method helps to clear the text fields, radio buttons and check boxes which were previously entered and selected by the user.

r) priceSearchClearButtonActionPerformed(java.awt.event.ActionEvent evt)

This method will clear the price search text field value entered by user.

s) clearSelectedButtonActionPerformed(java.awt.event.ActionEvent evt)

This button helps to clear the selected accessory item from the table.

t) accessoriesNameTextFieldKeyTyped(java.awt.event.KeyEvent evt)

This method will disable the user to enter numerical value in accessories name.

u) main(String args[])

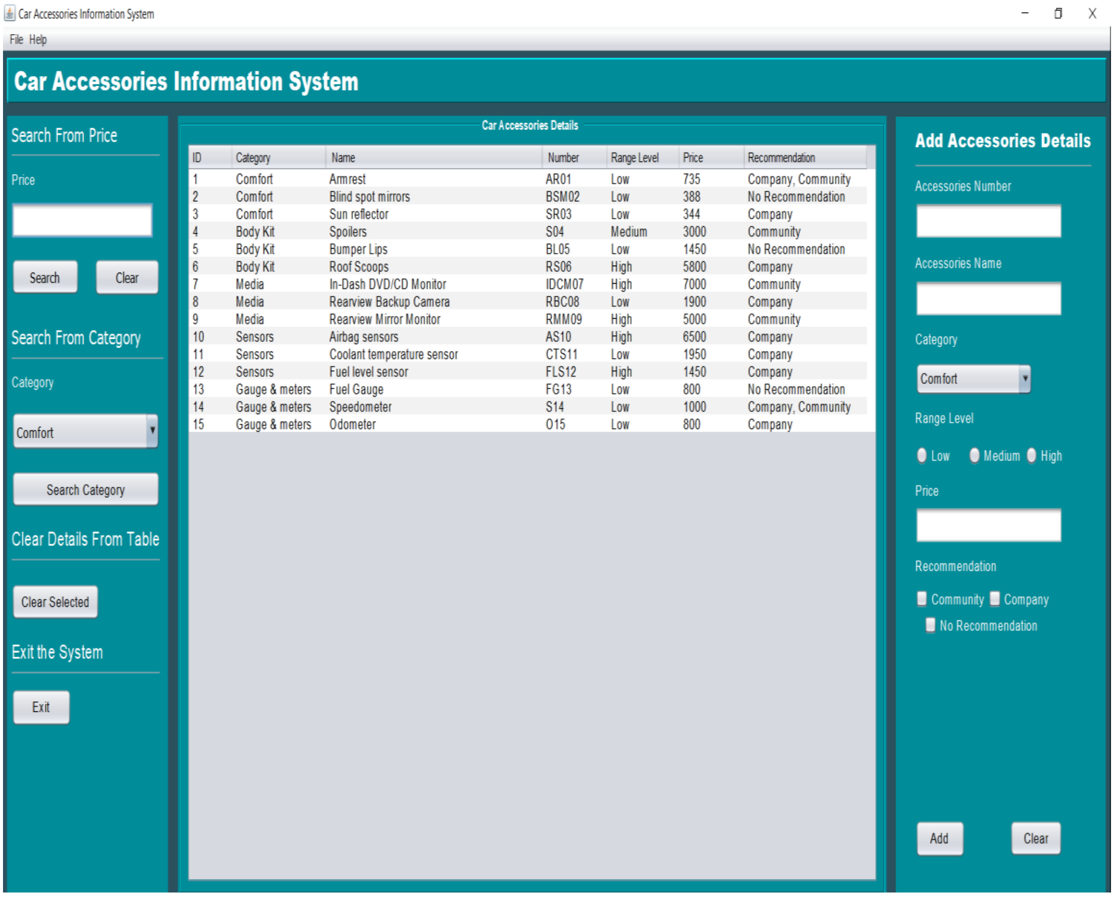
This is the main method of the CAD_Info JFrame which consists of a method run() inside it where the constructor CAD_Info() is called and it's visibility is set to true.

5. Testing

5.1 Test case 1: To check whether the program runs or not.

Test	1
Action	Run project CAD_IS.
Excepted Result	Application will run and Car Accessories Information System will be displayed.
Actual Result	Application ran smoothly and car accessories information system was displayed.
Conclusion	Successful.

Table 5: Test case 1



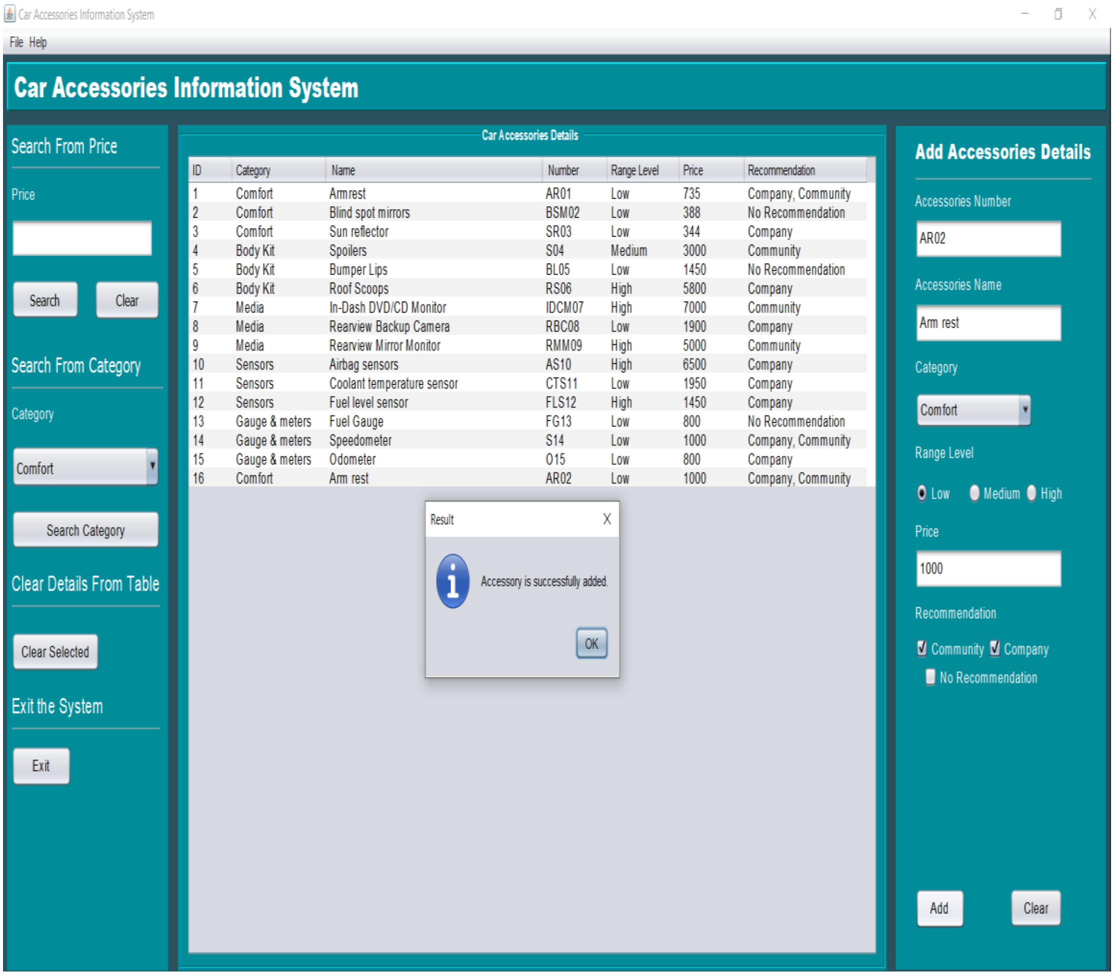
ID	Category	Name	Number	Range Level	Price	Recommendation
1	Comfort	Armrest	AR01	Low	735	Company, Community
2	Comfort	Blind spot mirrors	BSM02	Low	388	No Recommendation
3	Comfort	Sun reflector	SR03	Low	344	Company
4	Body Kit	Spoilers	S04	Medium	3000	Community
5	Body Kit	Bumper Lips	BL05	Low	1450	No Recommendation
6	Body Kit	Roof Scoops	RS06	High	5800	Company
7	Media	In-Dash DVD/CD Monitor	IDCM07	High	7000	Community
8	Media	Rearview Backup Camera	RBC08	Low	1900	Company
9	Media	Rearview Mirror Monitor	RMM09	High	5000	Community
10	Sensors	Airbag sensors	AS10	High	6500	Company
11	Sensors	Coolant temperature sensor	CTS11	Low	1950	Company
12	Sensors	Fuel level sensor	FLS12	High	1450	Company
13	Gauge & meters	Fuel Gauge	FG13	Low	800	No Recommendation
14	Gauge & meters	Speedometer	S14	Low	1000	Company, Community
15	Gauge & meters	Odometer	O15	Low	800	Company

Figure 3: Test case 1

5.2 Test case 2: To check whether the accessory item will be added or not in the table.

Test	2
Action	All the values were added in the text fields and options were selected in the check boxes and radio buttons.
Excepted Result	The item should be added and displayed in the table.
Actual Result	Item was added and displayed in the table.
Conclusion	Successful.

Table 6: Test case 2



Category: Comfort

Car Accessories Details

ID	Category	Name	Number	Range Level	Price	Recommendation
1	Comfort	Armrest	AR01	Low	735	Company, Community
2	Comfort	Blind spot mirrors	BSM02	Low	388	No Recommendation
3	Comfort	Sun reflector	SR03	Low	344	Company
4	Body Kit	Spoilers	S04	Medium	3000	Community
5	Body Kit	Bumper Lips	BL05	Low	1450	No Recommendation
6	Body Kit	Roof Scoops	RS06	High	5800	Company
7	Media	In-Dash DVD/CD Monitor	IDCM07	High	7000	Community
8	Media	Rearview Backup Camera	RBC08	Low	1900	Company
9	Media	Rearview Mirror Monitor	RMM09	High	5000	Community
10	Sensors	Airbag sensors	AS10	High	6500	Company
11	Sensors	Coolant temperature sensor	CTS11	Low	1950	Company
12	Sensors	Fuel level sensor	FLS12	High	1450	Company
13	Gauge & meters	Fuel Gauge	FG13	Low	800	No Recommendation
14	Gauge & meters	Speedometer	S14	Low	1000	Company, Community
15	Gauge & meters	Odometer	O15	Low	800	Company
16	Comfort	Arm rest	AR02	Low	1000	Company, Community

Result

Accessory is successfully added.

Accessories Number: AR02

Accessories Name: Arm rest

Category: Comfort

Range Level: ● Low ○ Medium ○ High

Price: 1000

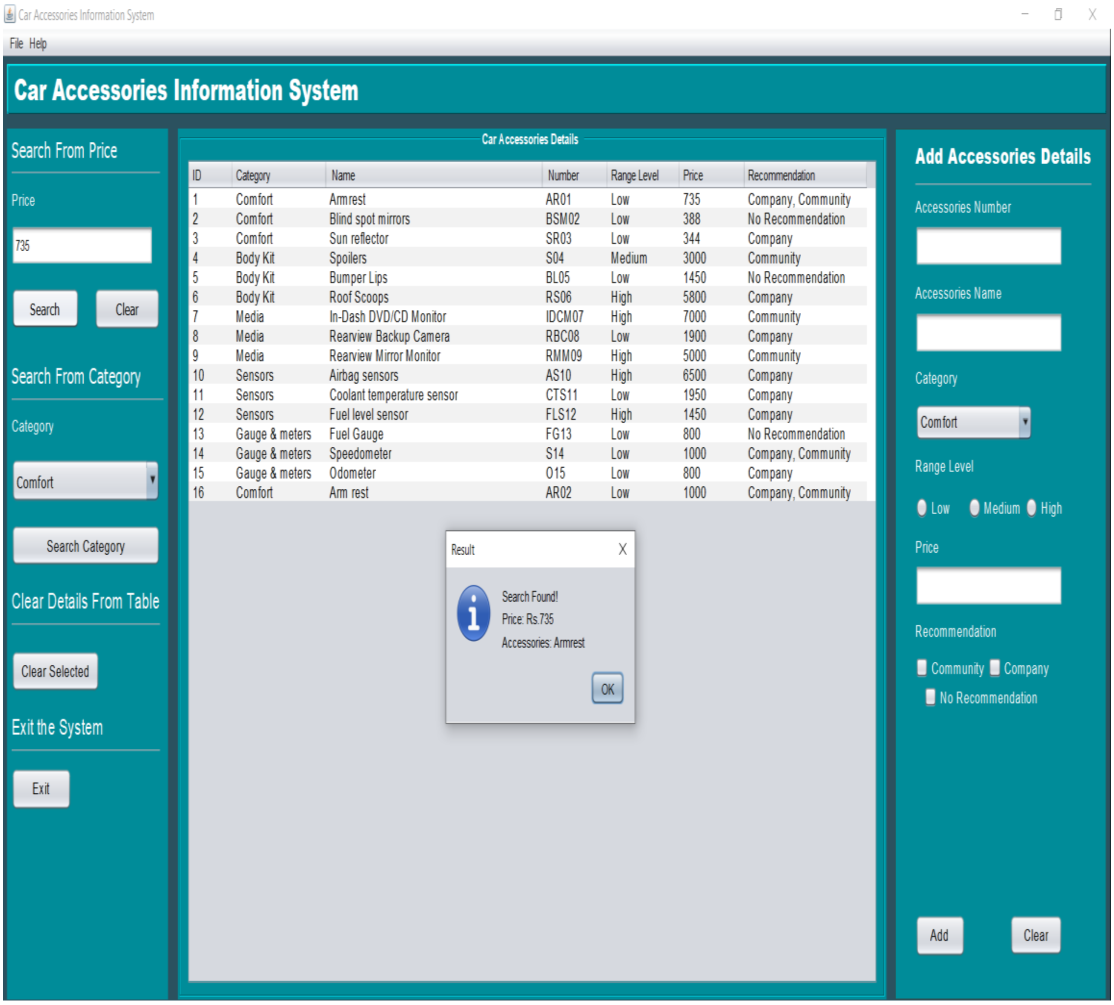
Recommendation: ☑ Community ☑ Company ☐ No Recommendation

Figure 4: Test case 2

5.3 Test case 3: Searching item based on price in table

Test	3
Action	Price amount is entered in the text field and Search button is clicked.
Excepted Result	The searched item will be displayed in a message dialog box.
Actual Result	The searched item displayed in a message dialog box.
Conclusion	Successful.

Table 7: Test case 3



Car Accessories Information System

Search From Price

Price: 735

Car Accessories Details

ID	Category	Name	Number	Range Level	Price	Recommendation
1	Comfort	Armrest	AR01	Low	735	Company, Community
2	Comfort	Blind spot mirrors	BSM02	Low	388	No Recommendation
3	Comfort	Sun reflector	SR03	Low	344	Company
4	Body Kit	Spoilers	S04	Medium	3000	Community
5	Body Kit	Bumper Lips	BL05	Low	1450	No Recommendation
6	Body Kit	Roof Scoops	RS06	High	5800	Company
7	Media	In-Dash DVD/CD Monitor	IDCM07	High	7000	Community
8	Media	Rearview Backup Camera	RBC08	Low	1900	Company
9	Media	Rearview Mirror Monitor	RMM09	High	5000	Community
10	Sensors	Airbag sensors	AS10	High	6500	Company
11	Sensors	Coolant temperature sensor	CTS11	Low	1950	Company
12	Sensors	Fuel level sensor	FLS12	High	1450	Company
13	Gauge & meters	Fuel Gauge	FG13	Low	800	No Recommendation
14	Gauge & meters	Speedometer	S14	Low	1000	Company, Community
15	Gauge & meters	Odometer	015	Low	800	Company
16	Comfort	Arm rest	AR02	Low	1000	Company, Community

Result

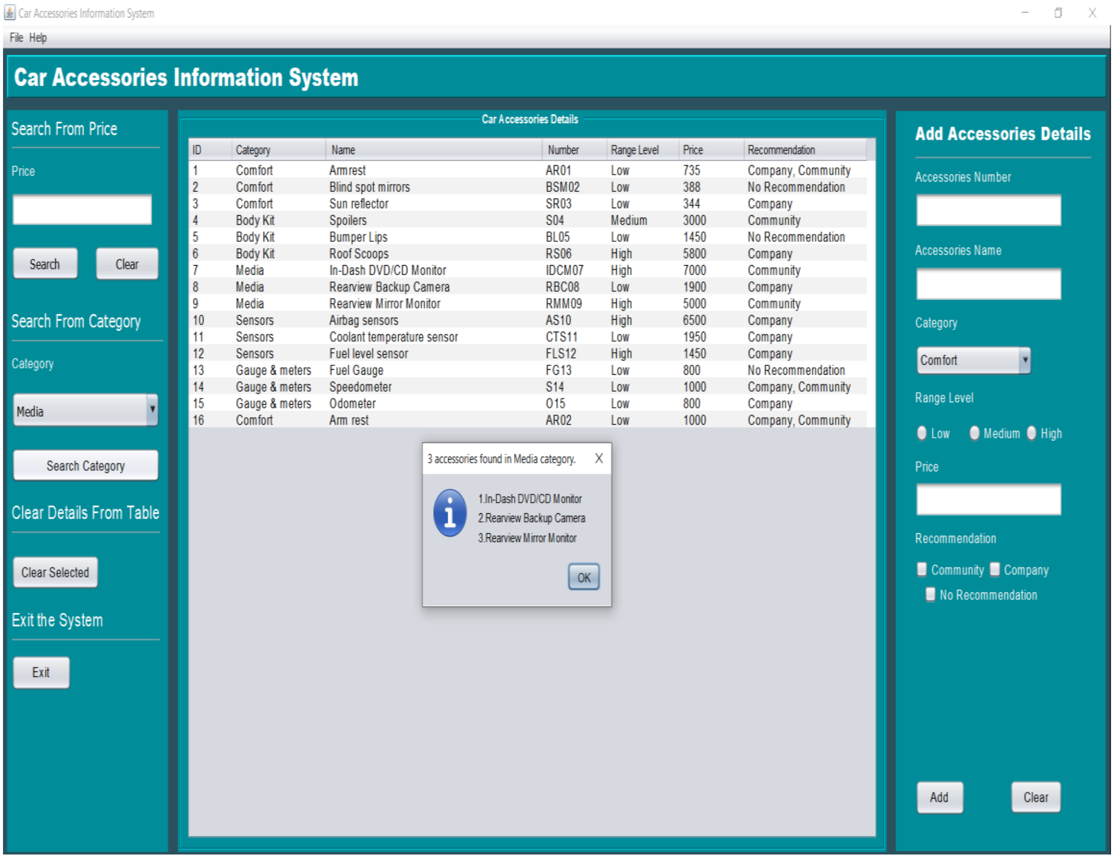
Search Found!
Price: Rs.735
Accessories: Armrest

Figure 5: Test case 3

5.4 Test case 4: Searching items based on category in table

Test	4
Action	Media category was selected and “Search Category” button was clicked.
Excepted Result	The items under Media category will be displayed in a message dialog box.
Actual Result	Three items under Media category displayed in a message dialog box.
Conclusion	Successful.

Table 8: Test case 4



Category: Media

Category: Comfort

ID	Category	Name	Number	Range Level	Price	Recommendation
1	Comfort	Armrest	AR01	Low	735	Company, Community
2	Comfort	Blind spot mirrors	BSM02	Low	388	No Recommendation
3	Comfort	Sun reflector	SR03	Low	344	Company
4	Body Kit	Spoilers	S04	Medium	3000	Community
5	Body Kit	Bumper Lips	BL05	Low	1450	No Recommendation
6	Body Kit	Roof Scoops	RS06	High	5800	Company
7	Media	In-Dash DVD/CD Monitor	IDCM07	High	7000	Community
8	Media	Rearview Backup Camera	RBC08	Low	1900	Company
9	Media	Rearview Mirror Monitor	RMM09	High	5000	Community
10	Sensors	Airbag sensors	AS10	High	6500	Company
11	Sensors	Coolant temperature sensor	CTS11	Low	1950	Company
12	Sensors	Fuel level sensor	FLS12	High	1450	Company
13	Gauge & meters	Fuel Gauge	FG13	Low	800	No Recommendation
14	Gauge & meters	Speedometer	S14	Low	1000	Company, Community
15	Gauge & meters	Odometer	O15	Low	800	Company
16	Comfort	Arm rest	AR02	Low	1000	Company, Community

3 accessories found in Media category. X

1.In-Dash DVD/CD Monitor
2.Rearview Backup Camera
3.Rearview Mirror Monitor

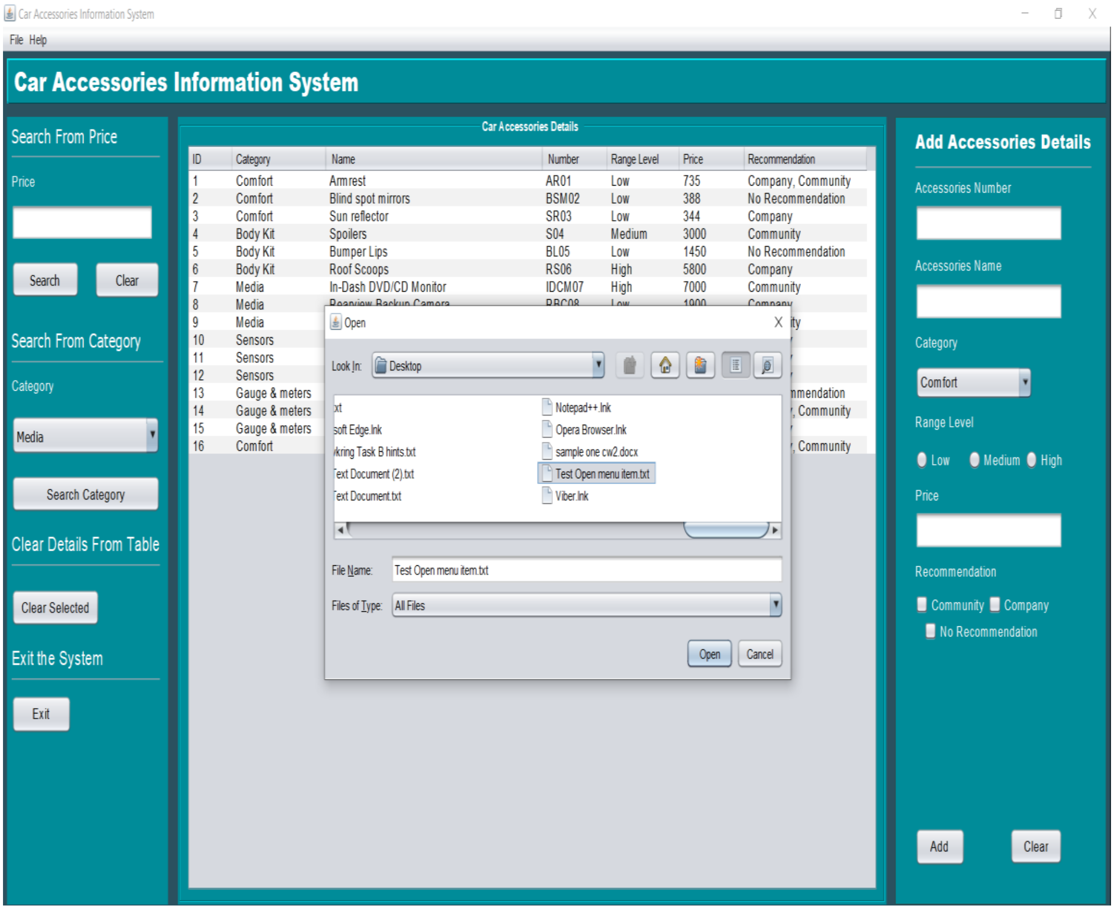
OK

Figure 6: Test case 4

5.5 Test case 5: To check whether a file can be opened from Open menu item

Test	5
Action	"Open" menu item inside the File menu was clicked. Then, a doc file named Java was clicked to open.
Excepted Result	The Java doc file will open.
Actual Result	The Java doc file opened.
Conclusion	Successful

Table 9: Test case 5



Category: Media

Category: Comfort

ID	Category	Name	Number	Range Level	Price	Recommendation
1	Comfort	Armrest	AR01	Low	735	Company, Community
2	Comfort	Blind spot mirrors	BSM02	Low	388	No Recommendation
3	Comfort	Sun reflector	SR03	Low	344	Company
4	Body Kit	Spoilers	S04	Medium	3000	Community
5	Body Kit	Bumper Lips	BL05	Low	1450	No Recommendation
6	Body Kit	Roof Scoops	RS06	High	5800	Company
7	Media	In-Dash DVD/CD Monitor	IDCM07	High	7000	Community
8	Media	Rearview Backup Camera	RBC08	Low	1900	Company
9	Media					ity
10	Sensors					
11	Sensors					
12	Sensors					
13	Gauge & meters					mmendation
14	Gauge & meters					, Community
15	Gauge & meters					
16	Comfort					, Community

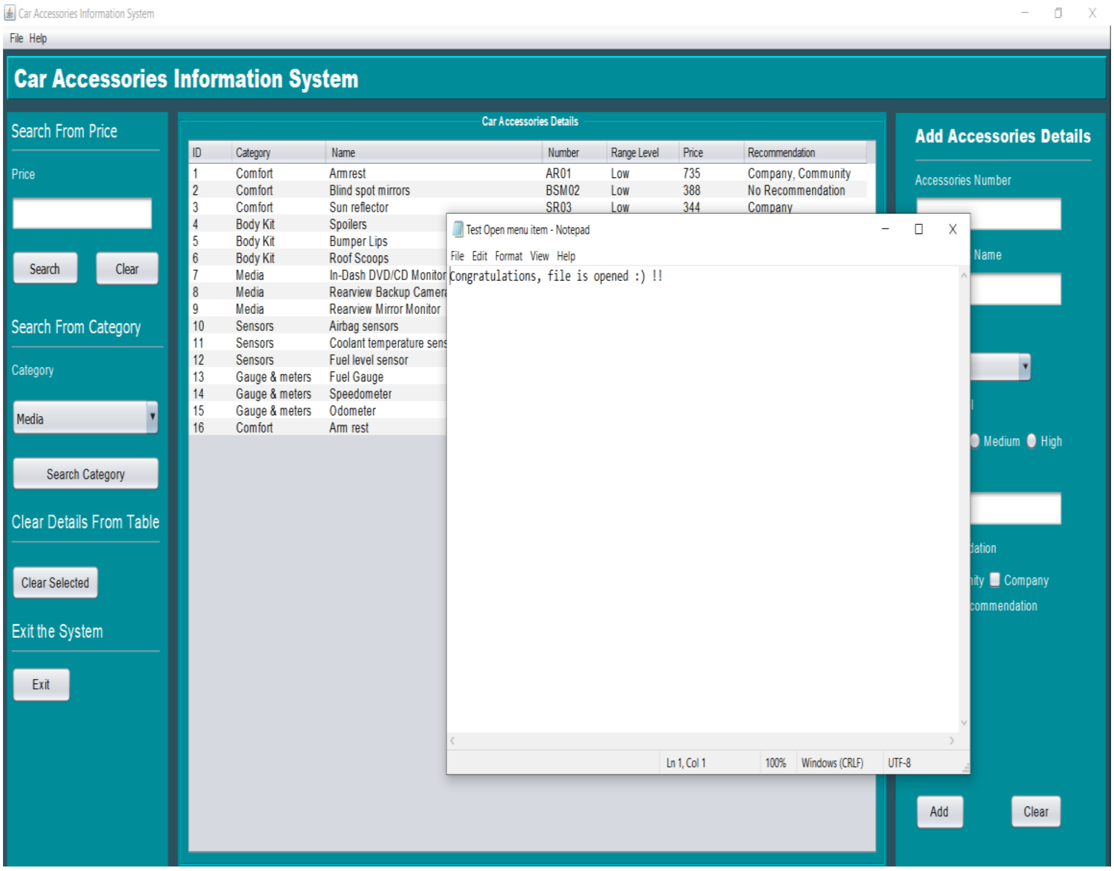
Look In: Desktop

xt
soft Edge.lnk
kring Task B hints.txt
Text Document (2).txt
Text Document.txt
Notepad++.lnk
Opera Browser.lnk
sample one cw2.docx
Test Open menu item.txt
Viber.lnk

File Name: Test Open menu item.txt

Files of Type: All Files

Figure 7: Test case 5



ID	Category	Name	Number	Range Level	Price	Recommendation
1	Comfort	Armrest	AR01	Low	735	Company, Community
2	Comfort	Blind spot mirrors	BSM02	Low	388	No Recommendation
3	Comfort	Sun reflector	SR03	Low	344	Company
4	Body Kit	Spoilers				
5	Body Kit	Bumper Lips				
6	Body Kit	Roof Scoops				
7	Media	In-Dash DVD/CD Monitor				
8	Media	Rearview Backup Camera				
9	Media	Rearview Mirror Monitor				
10	Sensors	Airbag sensors				
11	Sensors	Coolant temperature sens				
12	Sensors	Fuel level sensor				
13	Gauge & meters	Fuel Gauge				
14	Gauge & meters	Speedometer				
15	Gauge & meters	Odometer				
16	Comfort	Arm rest				

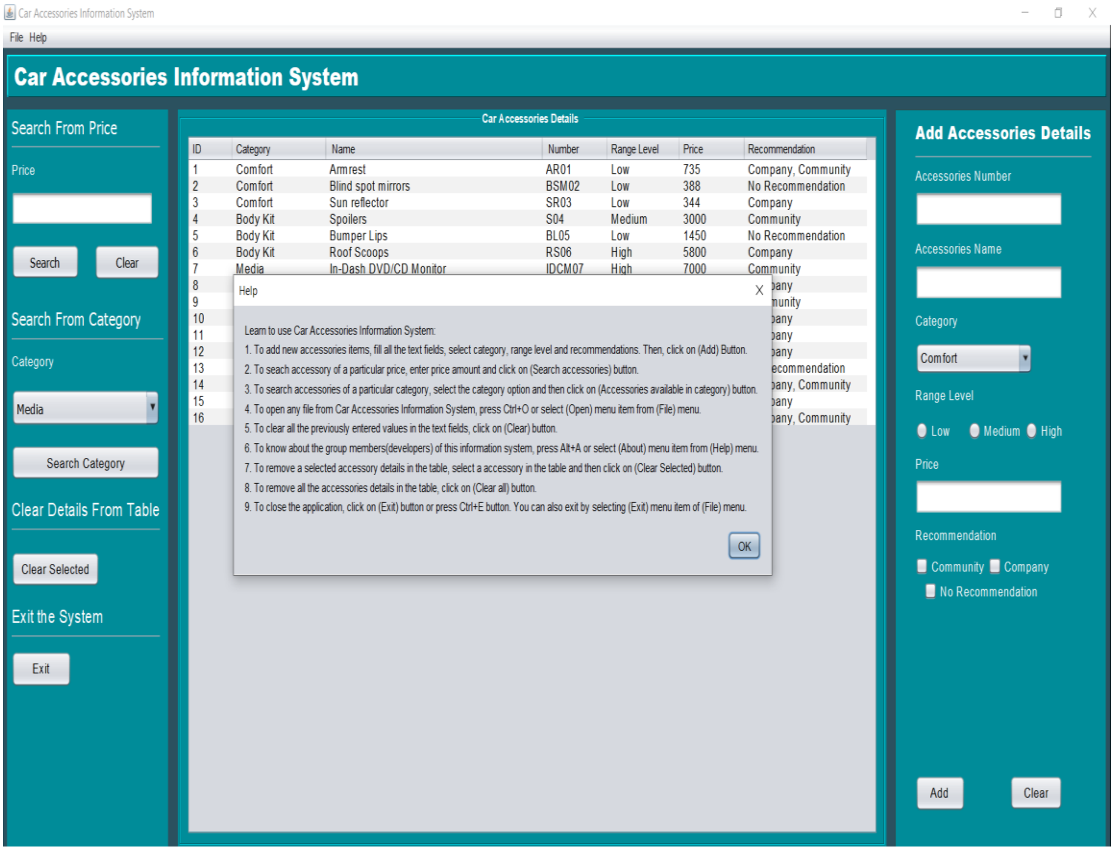
```
Congratulations, file is opened :) !!
```

Figure 8: Test case 5

5.6 Test case 6: To check whether “View Help” menu item of “Help” menu will function or not.

Test	6
Action	View Help menu item inside the Help menu was clicked.
Excepted Result	A message dialog box will appear.
Actual Result	A message dialog box appeared with some instructions on how to run the features of the application.
Conclusion	Successful

Table 10: Test case 6



ID	Category	Name	Number	Range Level	Price	Recommendation
1	Comfort	Armrest	AR01	Low	735	Company, Community
2	Comfort	Blind spot mirrors	BSM02	Low	388	No Recommendation
3	Comfort	Sun reflector	SR03	Low	344	Company
4	Body Kit	Spoilers	S04	Medium	3000	Community
5	Body Kit	Bumper Lips	BL05	Low	1450	No Recommendation
6	Body Kit	Roof Scoops	RS06	High	5800	Company
7	Media	In-Dash DVD/CD Monitor	IDCM07	High	7000	Community
8						pany
9						munity
10						pany
11						pany
12						pany
13						ecommendation
14						pany, Community
15						pany
16						pany, Community

Learn to use Car Accessories Information System:

1. To add new accessories items, fill all the text fields, select category, range level and recommendations. Then, click on (Add) Button.
2. To seach accessory of a particular price, enter price amount and click on (Search accessories) button.
3. To search accessories of a particular category, select the category option and then click on (Accessories available in category) button.
4. To open any file from Car Accessories Information System, press Ctrl+O or select (Open) menu item from (File) menu.
5. To clear all the previously entered values in the text fields, click on (Clear) button.
6. To know about the group members(developers) of this information system, press Alt+A or select (About) menu item from (Help) menu.
7. To remove a selected accessory details in the table, select a accessory in the table and then click on (Clear Selected) button.
8. To remove all the accessories details in the table, click on (Clear all) button.
9. To close the application, click on (Exit) button or press Ctrl+E button. You can also exit by selecting (Exit) menu item of (File) menu.

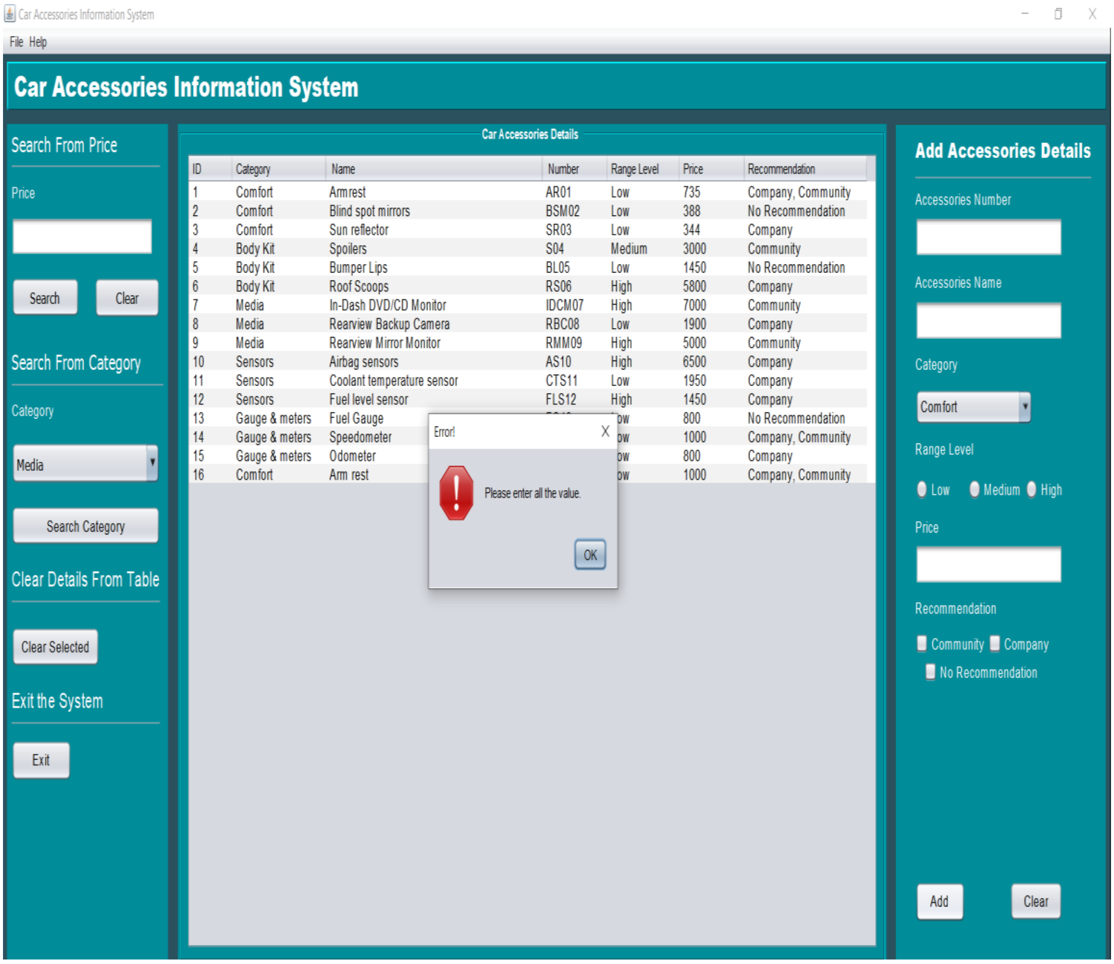
Figure 9: Test case 6

5.7 Test case 7: To check whether data will be inserted into the table if the text fields, check boxes and radio buttons are left empty.

7.1 No values were entered.

Test	7.1
Action	Add button was clicked without inserting any details.
Excepted Result	An error message box will appear which will inform to enter accessories number.
Actual Result	Error message box appeared which informed us to enter accessories number.
Conclusion	Successful.

Table 11: Test case 7.1



ID	Category	Name	Number	Range Level	Price	Recommendation
1	Comfort	Armrest	AR01	Low	735	Company, Community
2	Comfort	Blind spot mirrors	BSM02	Low	388	No Recommendation
3	Comfort	Sun reflector	SR03	Low	344	Company
4	Body Kit	Spoilers	S04	Medium	3000	Community
5	Body Kit	Bumper Lips	BL05	Low	1450	No Recommendation
6	Body Kit	Roof Scoops	RS06	High	5800	Company
7	Media	In-Dash DVD/CD Monitor	IDCM07	High	7000	Community
8	Media	Rearview Backup Camera	RBC08	Low	1900	Company
9	Media	Rearview Mirror Monitor	RMM09	High	5000	Community
10	Sensors	Airbag sensors	AS10	High	6500	Company
11	Sensors	Coolant temperature sensor	CTS11	Low	1950	Company
12	Sensors	Fuel level sensor	FLS12	High	1450	Company
13	Gauge & meters	Fuel Gauge		ow	800	No Recommendation
14	Gauge & meters	Speedometer		ow	1000	Company, Community
15	Gauge & meters	Odometer		ow	800	Company
16	Comfort	Arm rest		ow	1000	Company, Community

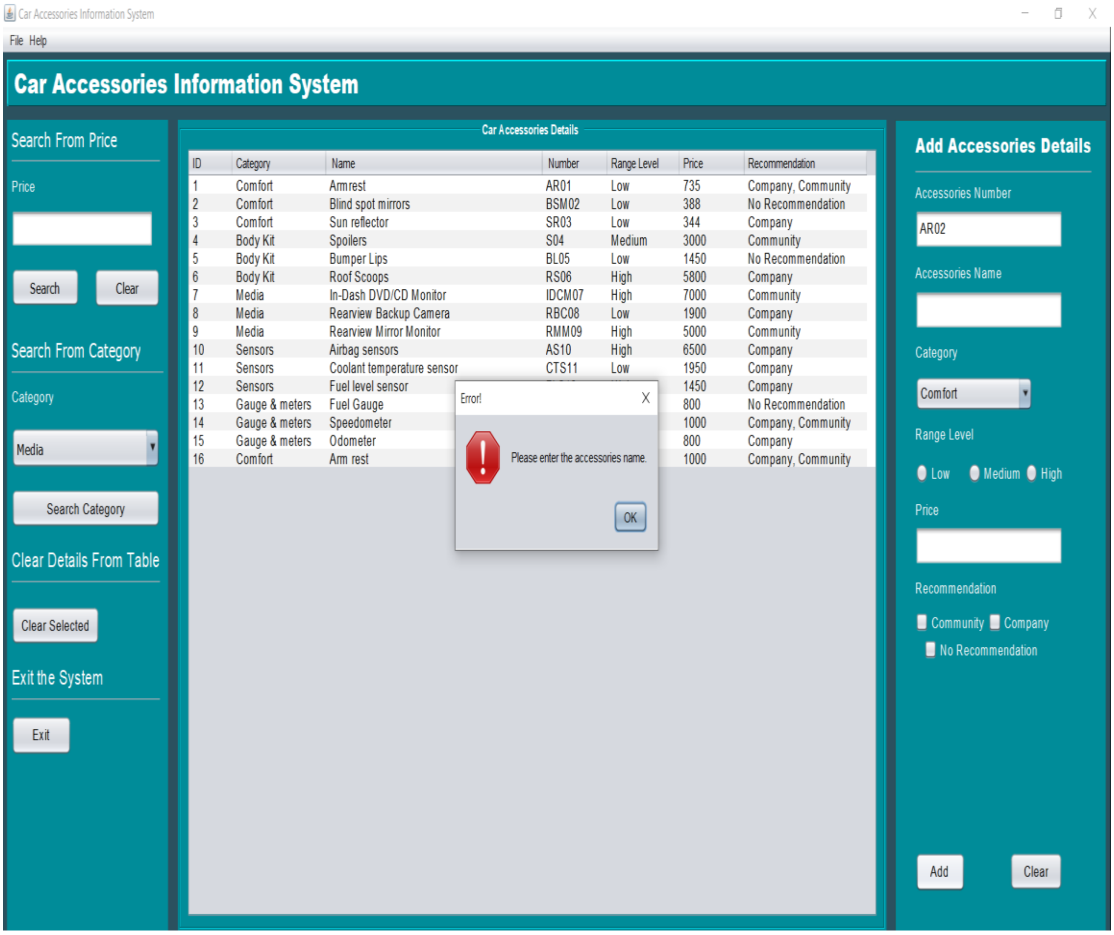
Please enter all the value.

Figure 10: Test case 7.1

7.2 Only Accessories Number was entered.

Test	7.2
Action	Add button was clicked after entering accessories number only.
Excepted Result	An error message box will appear which will inform to enter accessories name.
Actual Result	Error message box appeared which informed us to enter accessories name.
Conclusion	Successful.

Table 12: Test case 7.2



Category: Media

Accessories Number: AR02

Category: Comfort

Car Accessories Details

ID	Category	Name	Number	Range Level	Price	Recommendation
1	Comfort	Armrest	AR01	Low	735	Company, Community
2	Comfort	Blind spot mirrors	BSM02	Low	388	No Recommendation
3	Comfort	Sun reflector	SR03	Low	344	Company
4	Body Kit	Spoilers	S04	Medium	3000	Community
5	Body Kit	Bumper Lips	BL05	Low	1450	No Recommendation
6	Body Kit	Roof Scoops	RS06	High	5800	Company
7	Media	In-Dash DVD/CD Monitor	IDCM07	High	7000	Community
8	Media	Rearview Backup Camera	RBC08	Low	1900	Company
9	Media	Rearview Mirror Monitor	RMM09	High	5000	Community
10	Sensors	Airbag sensors	AS10	High	6500	Company
11	Sensors	Coolant temperature sensor	CTS11	Low	1950	Company
12	Sensors	Fuel level sensor			1450	Company
13	Gauge & meters	Fuel Gauge			800	No Recommendation
14	Gauge & meters	Speedometer			1000	Company, Community
15	Gauge & meters	Odometer			800	Company
16	Comfort	Arm rest			1000	Company, Community

Error!

Please enter the accessories name.

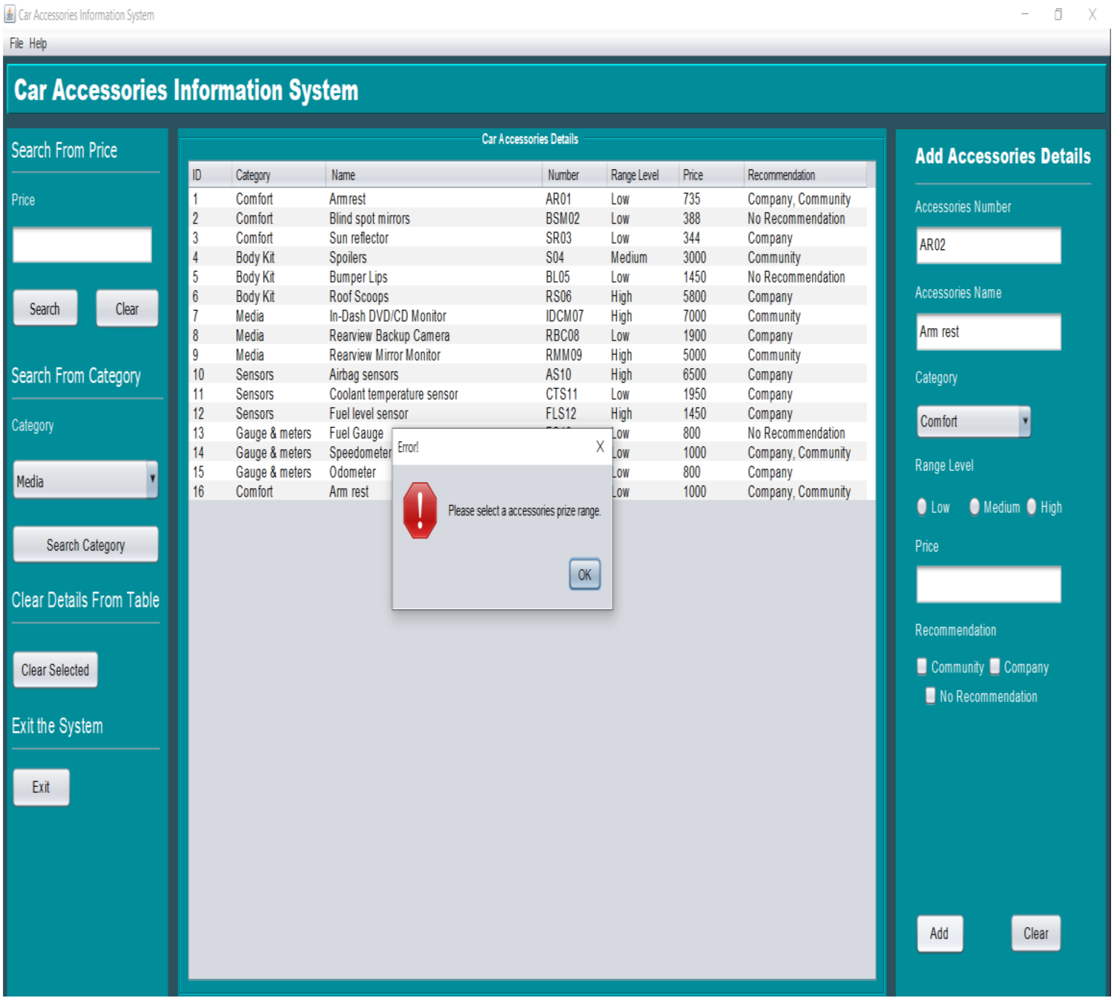
OK

Figure 11: Test case 7.2

7.3 Accessories number and name was entered.

Test	7.3
Action	Add button was clicked after entering accessories number and name.
Excepted Result	An error message box will appear which will inform to select a prize range.
Actual Result	Error message box appeared which informed us to select a prize range.
Conclusion	Successful.

Table 13: Test case 7.3



ID	Category	Name	Number	Range Level	Price	Recommendation
1	Comfort	Armrest	AR01	Low	735	Company, Community
2	Comfort	Blind spot mirrors	BSM02	Low	388	No Recommendation
3	Comfort	Sun reflector	SR03	Low	344	Company
4	Body Kit	Spoilers	S04	Medium	3000	Community
5	Body Kit	Bumper Lips	BL05	Low	1450	No Recommendation
6	Body Kit	Roof Scoops	RS06	High	5800	Company
7	Media	In-Dash DVD/CD Monitor	IDCM07	High	7000	Community
8	Media	Rearview Backup Camera	RBC08	Low	1900	Company
9	Media	Rearview Mirror Monitor	RMM09	High	5000	Community
10	Sensors	Airbag sensors	AS10	High	6500	Company
11	Sensors	Coolant temperature sensor	CTS11	Low	1950	Company
12	Sensors	Fuel level sensor	FLS12	High	1450	Company
13	Gauge & meters	Fuel Gauge		Low	800	No Recommendation
14	Gauge & meters	Speedometer		Low	1000	Company, Community
15	Gauge & meters	Odometer		Low	800	Company
16	Comfort	Arm rest		Low	1000	Company, Community

Error!

Please select a accessories prize range.

OK

Accessories Number: AR02

Accessories Name: Arm rest

Category: Comfort

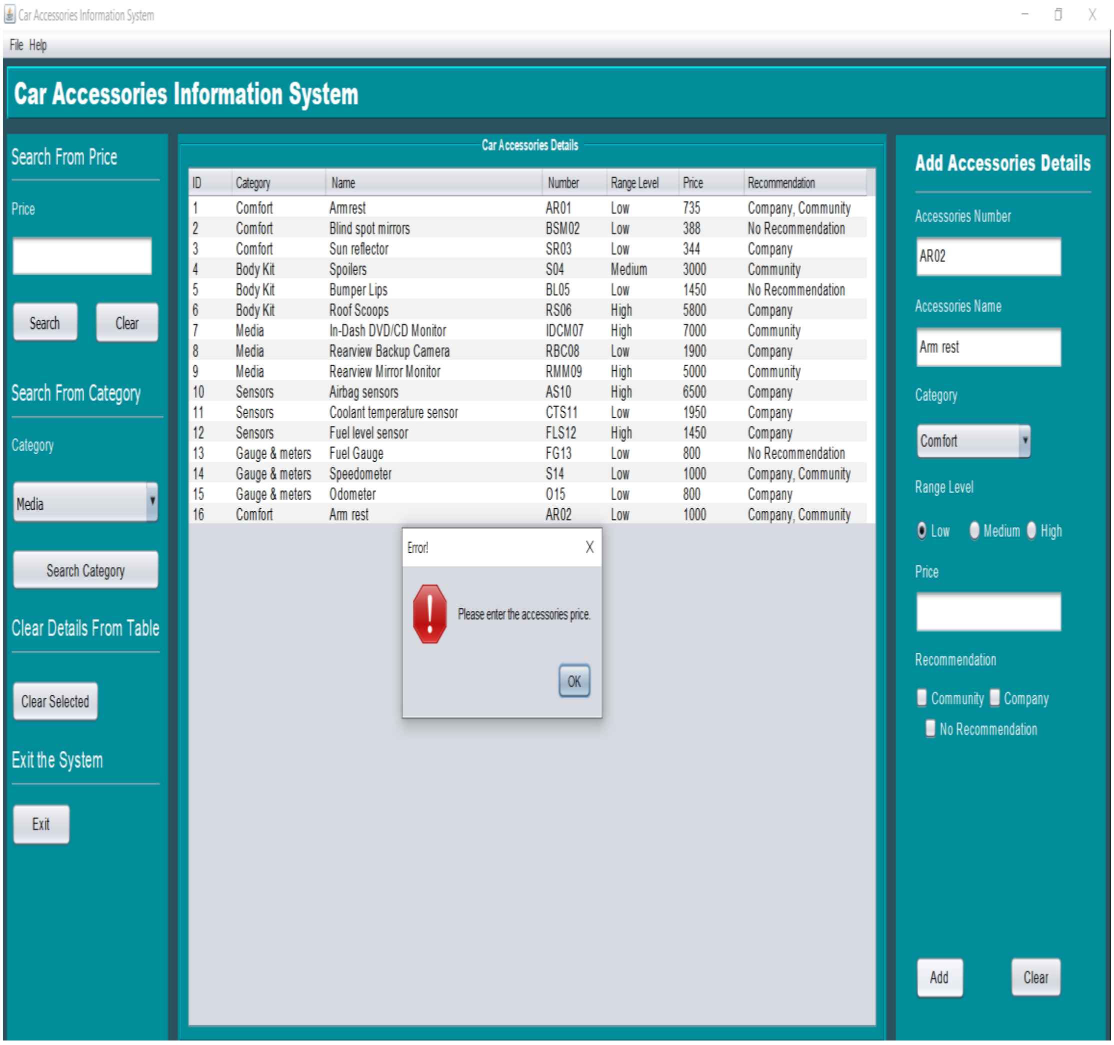
Price:

Figure 12: Test case 7.3

7.4 Price was not entered.

Test	7.4
Action	Add button was clicked without entering the price amount.
Excepted Result	An error message box will appear which will inform to enter the price.
Actual Result	Error message box appeared which informed us to enter a price.
Conclusion	Successful.

Table 14: Test case 7.4



Category: Media

ID	Category	Name	Number	Range Level	Price	Recommendation
1	Comfort	Armrest	AR01	Low	735	Company, Community
2	Comfort	Blind spot mirrors	BSM02	Low	388	No Recommendation
3	Comfort	Sun reflector	SR03	Low	344	Company
4	Body Kit	Spoilers	S04	Medium	3000	Community
5	Body Kit	Bumper Lips	BL05	Low	1450	No Recommendation
6	Body Kit	Roof Scoops	RS06	High	5800	Company
7	Media	In-Dash DVD/CD Monitor	IDCM07	High	7000	Community
8	Media	Rearview Backup Camera	RBC08	Low	1900	Company
9	Media	Rearview Mirror Monitor	RMM09	High	5000	Community
10	Sensors	Airbag sensors	AS10	High	6500	Company
11	Sensors	Coolant temperature sensor	CTS11	Low	1950	Company
12	Sensors	Fuel level sensor	FLS12	High	1450	Company
13	Gauge & meters	Fuel Gauge	FG13	Low	800	No Recommendation
14	Gauge & meters	Speedometer	S14	Low	1000	Company, Community
15	Gauge & meters	Odometer	O15	Low	800	Company
16	Comfort	Arm rest	AR02	Low	1000	Company, Community

Error!

Please enter the accessories price.

Accessories Number: AR02

Accessories Name: Arm rest

Category: Comfort

Range Level: ● Low ○ Medium ○ High

Price:

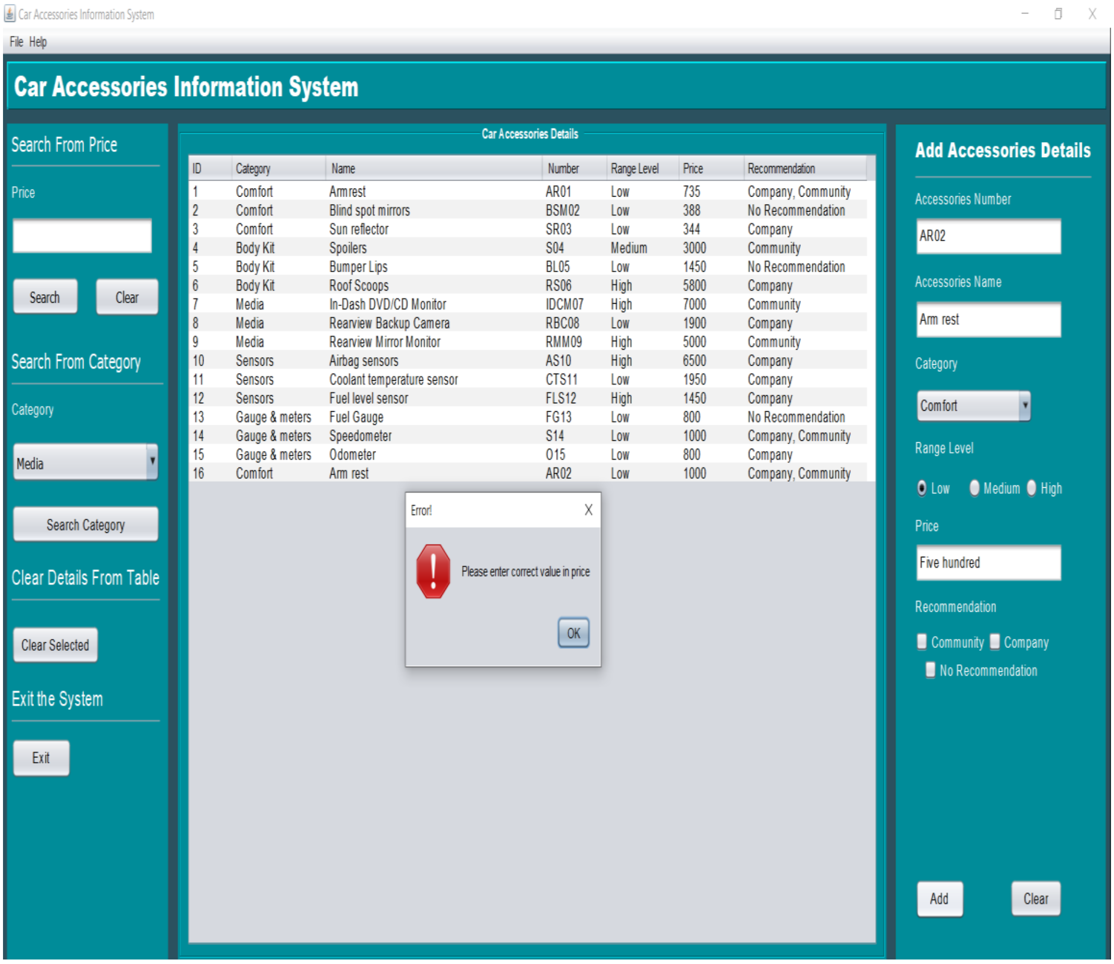
Recommendation: □ Community □ Company □ No Recommendation

Figure 13: Test case 7.4

7.5 String value was entered in the price.

Test	7.5
Action	String value was entered in the price.
Excepted Result	An error message box will appear which will inform to enter correct value in the price.
Actual Result	Error message box appeared which informed us to enter correct value in price.
Conclusion	Successful.

Table 15: Test case 7.5



Category: Media

ID	Category	Name	Number	Range Level	Price	Recommendation
1	Comfort	Armrest	AR01	Low	735	Company, Community
2	Comfort	Blind spot mirrors	BSM02	Low	388	No Recommendation
3	Comfort	Sun reflector	SR03	Low	344	Company
4	Body Kit	Spoilers	S04	Medium	3000	Community
5	Body Kit	Bumper Lips	BL05	Low	1450	No Recommendation
6	Body Kit	Roof Scoops	RS06	High	5800	Company
7	Media	In-Dash DVD/CD Monitor	IDCM07	High	7000	Community
8	Media	Rearview Backup Camera	RBC08	Low	1900	Company
9	Media	Rearview Mirror Monitor	RMM09	High	5000	Community
10	Sensors	Airbag sensors	AS10	High	6500	Company
11	Sensors	Coolant temperature sensor	CTS11	Low	1950	Company
12	Sensors	Fuel level sensor	FLS12	High	1450	Company
13	Gauge & meters	Fuel Gauge	FG13	Low	800	No Recommendation
14	Gauge & meters	Speedometer	S14	Low	1000	Company, Community
15	Gauge & meters	Odometer	015	Low	800	Company
16	Comfort	Arm rest	AR02	Low	1000	Company, Community

Error! X

Please enter correct value in price

OK

Accessories Number: AR02

Accessories Name: Arm rest

Category: Comfort

Range Level: ● Low ○ Medium ○ High

Price: Five hundred

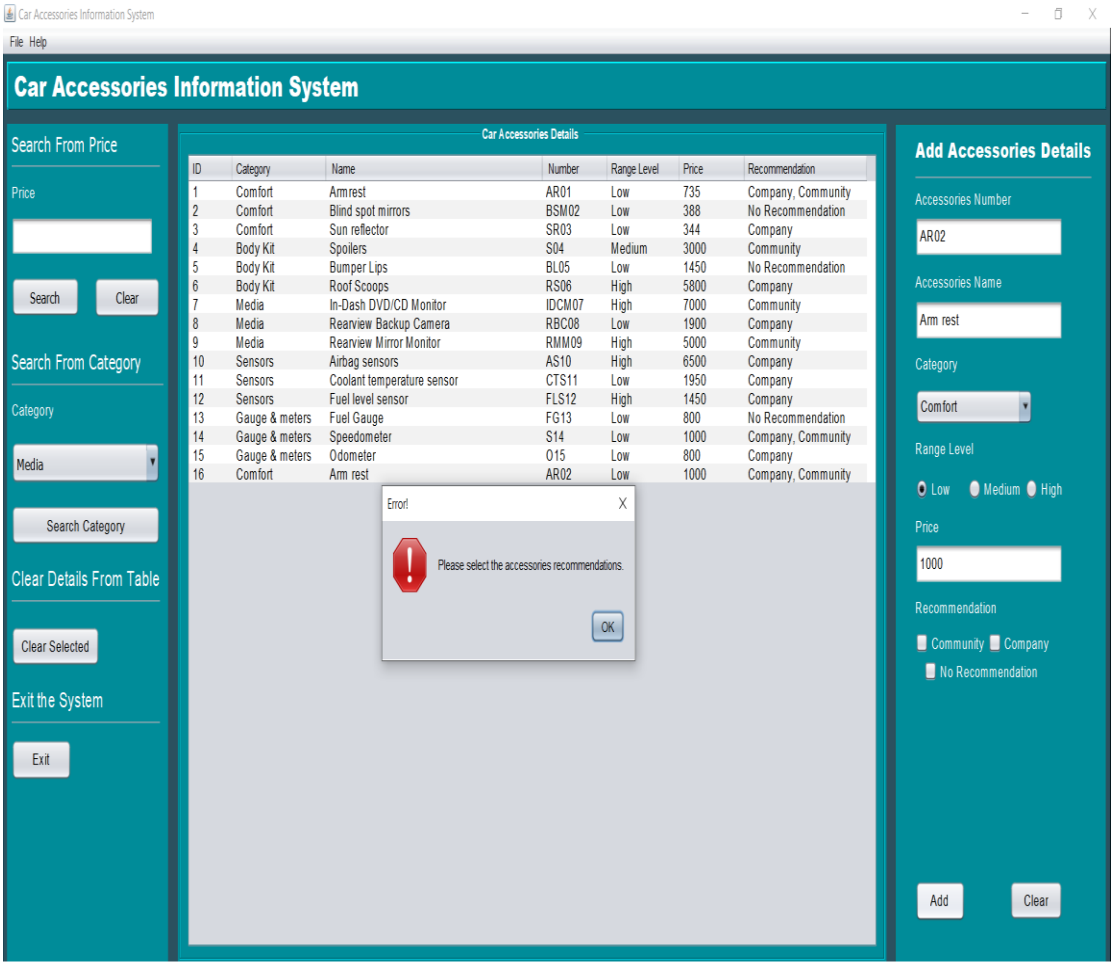
Recommendation: □ Community □ Company □ No Recommendation

Figure 14: Test case 7.5

7.6 Recommendation check boxes were not selected.

Test	7.6
Action	Recommendation check boxes were not selected.
Excepted Result	An error message box will appear which will inform us to select accessories recommendation.
Actual Result	Error message box appeared which informed us to select accessories recommendation.
Conclusion	Successful.

Table 16: Test case 7.6



Category: Media

Car Accessories Details

ID	Category	Name	Number	Range Level	Price	Recommendation
1	Comfort	Armrest	AR01	Low	735	Company, Community
2	Comfort	Blind spot mirrors	BSM02	Low	388	No Recommendation
3	Comfort	Sun reflector	SR03	Low	344	Company
4	Body Kit	Spoilers	S04	Medium	3000	Community
5	Body Kit	Bumper Lips	BL05	Low	1450	No Recommendation
6	Body Kit	Roof Scoops	RS06	High	5800	Company
7	Media	In-Dash DVD/CD Monitor	IDCM07	High	7000	Community
8	Media	Rearview Backup Camera	RBC08	Low	1900	Company
9	Media	Rearview Mirror Monitor	RMM09	High	5000	Community
10	Sensors	Airbag sensors	AS10	High	6500	Company
11	Sensors	Coolant temperature sensor	CTS11	Low	1950	Company
12	Sensors	Fuel level sensor	FLS12	High	1450	Company
13	Gauge & meters	Fuel Gauge	FG13	Low	800	No Recommendation
14	Gauge & meters	Speedometer	S14	Low	1000	Company, Community
15	Gauge & meters	Odometer	015	Low	800	Company
16	Comfort	Arm rest	AR02	Low	1000	Company, Community

Error!

Please select the accessories recommendations.

OK

Accessories Number: AR02

Accessories Name: Arm rest

Category: Comfort

Range Level: ● Low ○ Medium ○ High

Price: 1000

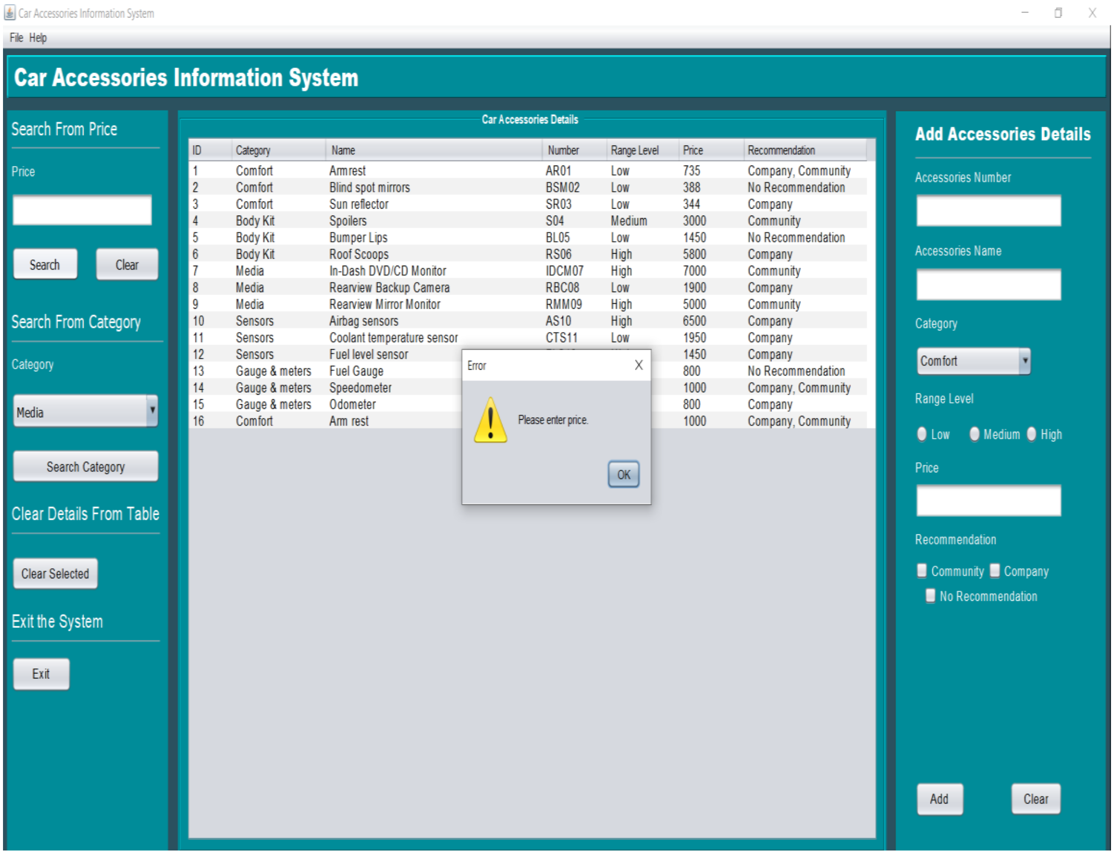
Recommendation: □ Community □ Company □ No Recommendation

Figure 15: Test case 7.6

5.8 Test case 8: To check whether a warning message dialog box will appear or not when price search button is clicked by leaving text field empty.

Test	8
Action	Price search button was clicked without entering value in text field.
Excepted Result	Warning message box will appear and will inform to enter price.
Actual Result	Warning message box appeared and informed to enter price.
Conclusion	Successful.

Table 17: Test case 8



ID	Category	Name	Number	Range Level	Price	Recommendation
1	Comfort	Armrest	AR01	Low	735	Company, Community
2	Comfort	Blind spot mirrors	BSM02	Low	388	No Recommendation
3	Comfort	Sun reflector	SR03	Low	344	Company
4	Body Kit	Spoilers	S04	Medium	3000	Community
5	Body Kit	Bumper Lips	BL05	Low	1450	No Recommendation
6	Body Kit	Roof Scoops	RS06	High	5800	Company
7	Media	In-Dash DVD/CD Monitor	IDCM07	High	7000	Community
8	Media	Rearview Backup Camera	RBC08	Low	1900	Company
9	Media	Rearview Mirror Monitor	RMM09	High	5000	Community
10	Sensors	Airbag sensors	AS10	High	6500	Company
11	Sensors	Coolant temperature sensor	CTS11	Low	1950	Company
12	Sensors	Fuel level sensor			1450	Company
13	Gauge & meters	Fuel Gauge			800	No Recommendation
14	Gauge & meters	Speedometer			1000	Company, Community
15	Gauge & meters	Odometer			800	Company
16	Comfort	Arm rest			1000	Company, Community

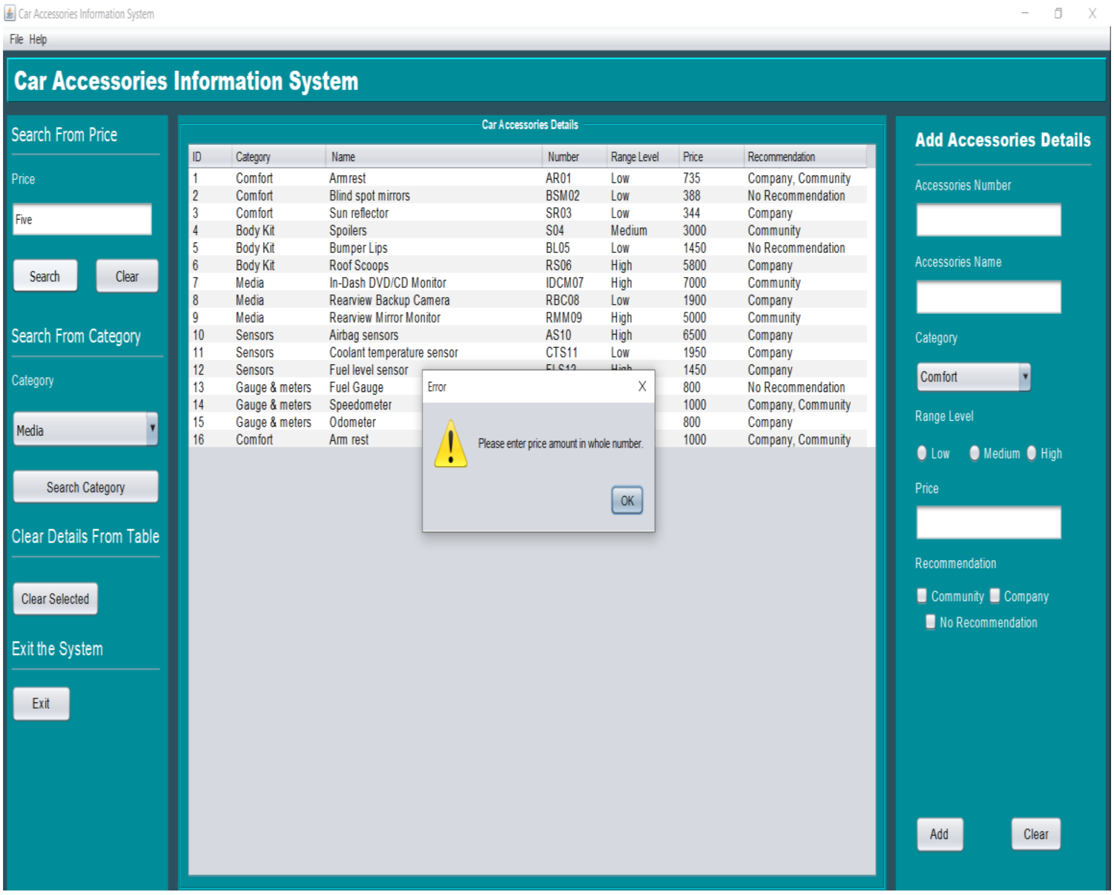
Please enter price.

Figure 16: Test case 8

5.9 Test case 9: To check whether warning message will appear or not when wrong value is entered in the price text field.

Test	9
Action	Five was entered in the text field and Search button was clicked.
Excepted Result	A warning message box will appear and inform us to enter price in number.
Actual Result	Warning message box appeared and informed to enter the price in number.
Conclusion	Successful.

Table 18: Test case 9



Price: Five

Category: Media

Category: Comfort

Price:

Car Accessories Details

ID	Category	Name	Number	Range Level	Price	Recommendation
1	Comfort	Armrest	AR01	Low	735	Company, Community
2	Comfort	Blind spot mirrors	BSM02	Low	388	No Recommendation
3	Comfort	Sun reflector	SR03	Low	344	Company
4	Body Kit	Spoilers	S04	Medium	3000	Community
5	Body Kit	Bumper Lips	BL05	Low	1450	No Recommendation
6	Body Kit	Roof Scoops	RS06	High	5800	Company
7	Media	In-Dash DVD/CD Monitor	IDCM07	High	7000	Community
8	Media	Rearview Backup Camera	RBC08	Low	1900	Company
9	Media	Rearview Mirror Monitor	RMM09	High	5000	Community
10	Sensors	Airbag sensors	AS10	High	6500	Company
11	Sensors	Coolant temperature sensor	CTS11	Low	1950	Company
12	Sensors	Fuel level sensor	FLS12	High	1450	Company
13	Gauge & meters	Fuel Gauge			800	No Recommendation
14	Gauge & meters	Speedometer			1000	Company, Community
15	Gauge & meters	Odometer			800	Company
16	Comfort	Arm rest			1000	Company, Community

Error

Please enter price amount in whole number.

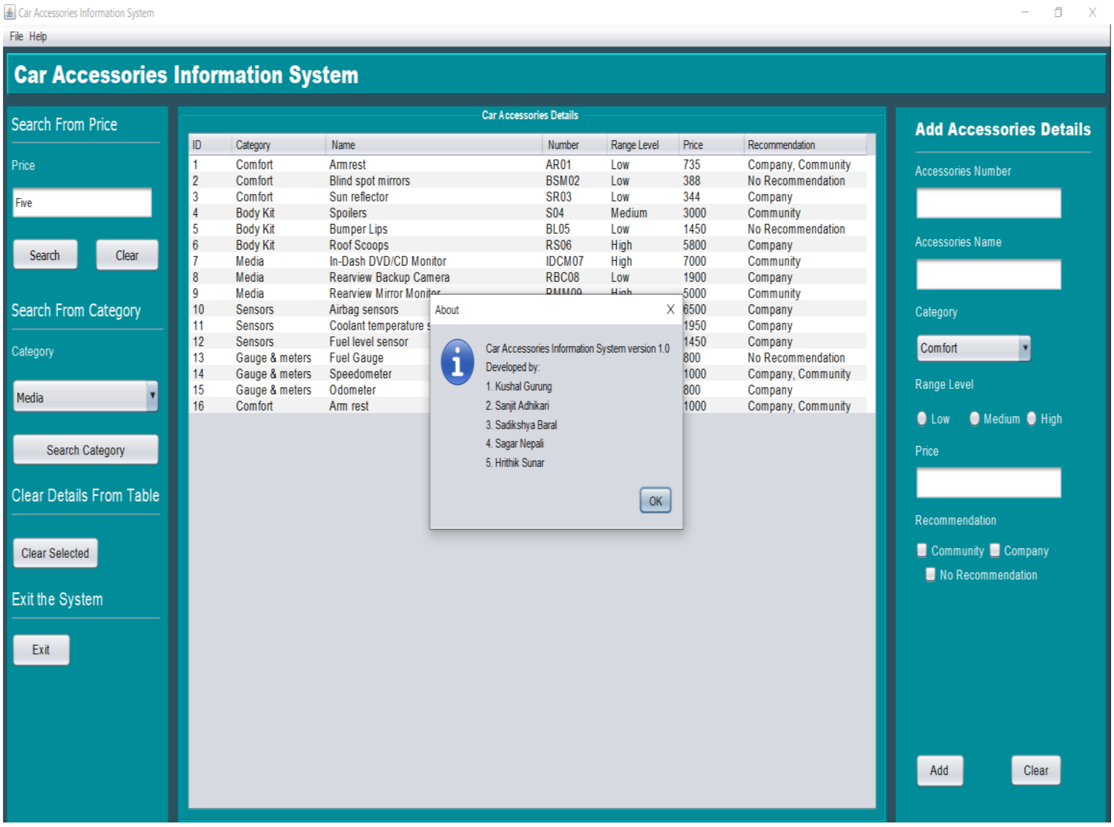
OK

Figure 17: Test case 9

5.10 Test case 10: To check whether “About” menu item of “Help” menu will function or not.

Test	10
Action	“About” menu item inside the “Help” menu was clicked.
Excepted Result	A message dialog box will appear.
Actual Result	A message dialog box appeared with the version and developer’s name of the information system.
Conclusion	Successful

Table 19: Test case 10



Price: Five

Category: Media

Category: Comfort

ID	Category	Name	Number	Range Level	Price	Recommendation
1	Comfort	Armrest	AR01	Low	735	Company, Community
2	Comfort	Blind spot mirrors	BSM02	Low	388	No Recommendation
3	Comfort	Sun reflector	SR03	Low	344	Company
4	Body Kit	Spoilers	S04	Medium	3000	Community
5	Body Kit	Bumper Lips	BL05	Low	1450	No Recommendation
6	Body Kit	Roof Scoops	RS06	High	5800	Company
7	Media	In-Dash DVD/CD Monitor	IDCM07	High	7000	Community
8	Media	Rearview Backup Camera	RBC08	Low	1900	Company
9	Media	Rearview Mirror Monitor	RMM09	High	5000	Community
10	Sensors	Airbag sensors			6500	Company
11	Sensors	Coolant temperature s			1950	Company
12	Sensors	Fuel level sensor			1450	Company
13	Gauge & meters	Fuel Gauge			800	No Recommendation
14	Gauge & meters	Speedometer			1000	Company, Community
15	Gauge & meters	Odometer			800	Company
16	Comfort	Arm rest			1000	Company, Community

About

Car Accessories Information System version 1.0
Developed by:
1. Kushal Gurung
2. Sanjit Adhikari
3. Sadikshya Baral
4. Sagar Nepali
5. Hrithik Sunar

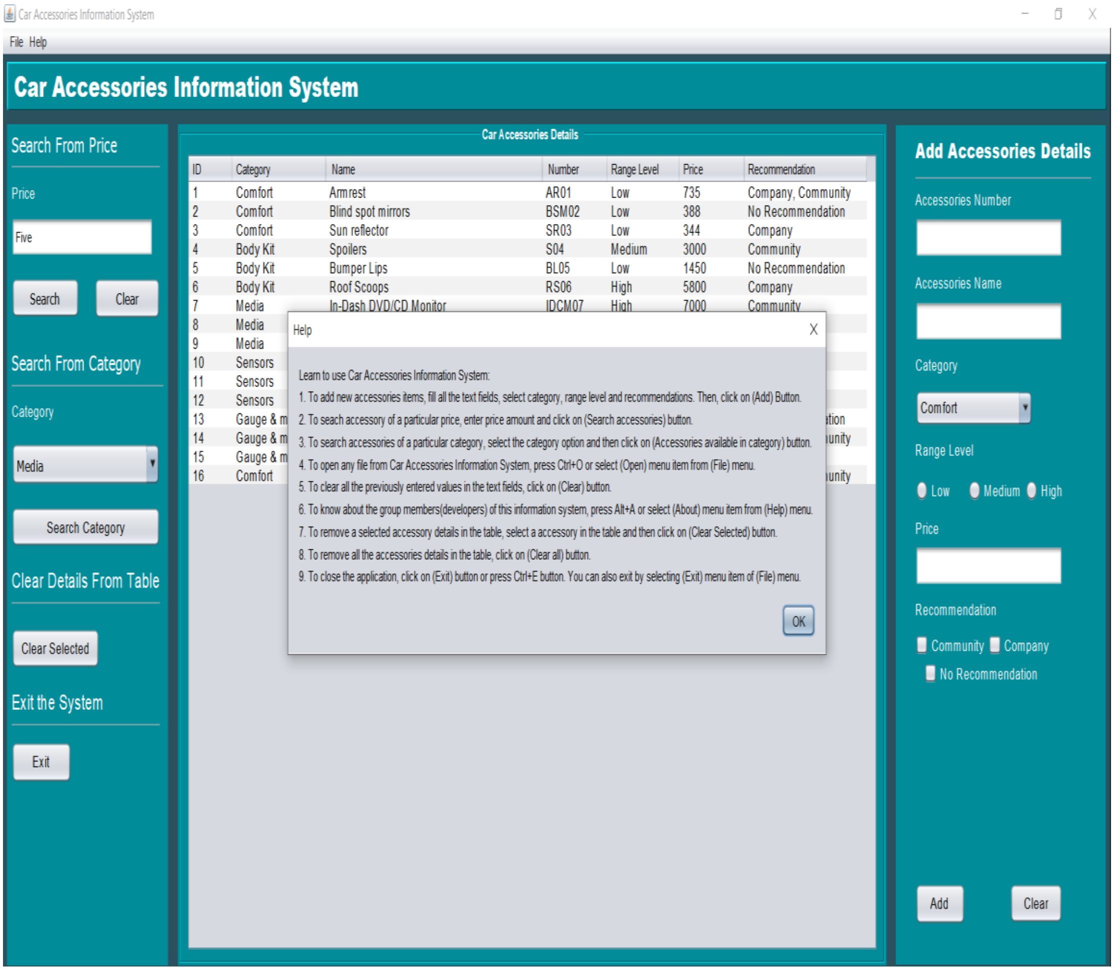
OK

Figure 18: Test case 10

5.11 Test case 11: To check whether “View Help” menu item of “Help” menu will function or not.

Test	11
Action	“View Help” menu item inside the “Help” menu was clicked.
Excepted Result	A message dialog box will appear.
Actual Result	A message dialog box appeared with the guidelines to run the program.
Conclusion	Successful

Table 20: Test case 11



Price: Five

Category: Media

Accessories Number:

Accessories Name:

Category: Comfort

Range Level: ○ Low ○ Medium ○ High

Price:

Recommendation: □ Community □ Company □ No Recommendation

ID	Category	Name	Number	Range Level	Price	Recommendation
1	Comfort	Armrest	AR01	Low	735	Company, Community
2	Comfort	Blind spot mirrors	BSM02	Low	388	No Recommendation
3	Comfort	Sun reflector	SR03	Low	344	Company
4	Body Kit	Spoilers	S04	Medium	3000	Community
5	Body Kit	Bumper Lips	BL05	Low	1450	No Recommendation
6	Body Kit	Roof Scoops	RS06	High	5800	Company
7	Media	In-Dash DVD/CD Monitor	IDCM07	High	7000	Community
8	Media					
9	Media					
10	Sensors					
11	Sensors					
12	Sensors					
13	Gauge & m					ition
14	Gauge & m					iunity
15	Gauge & m					
16	Comfort					iunity

Learn to use Car Accessories Information System:

1. To add new accessories items, fill all the text fields, select category, range level and recommendations. Then, click on (Add) Button.

2. To seach accessory of a particular price, enter price amount and click on (Search accessories) button.

3. To search accessories of a particular category, select the category option and then click on (Accessories available in category) button.

4. To open any file from Car Accessories Information System, press Ctrl+O or select (Open) menu item from (File) menu.

5. To clear all the previously entered values in the text fields, click on (Clear) button.

6. To know about the group members(developers) of this information system, press Alt+A or select (About) menu item from (Help) menu.

7. To remove a selected accessory details in the table, select a accessory in the table and then click on (Clear Selected) button.

8. To remove all the accessories details in the table, click on (Clear all) button.

9. To close the application, click on (Exit) button or press Ctrl+E button. You can also exit by selecting (Exit) menu item of (File) menu.

Figure 19: Test case 11

6. Conclusion

Finally, the group coursework for the development of Car Accessories Information System was completed with the desired outcome. The project that we developed is very decent and functions accurately according to the requirement of the coursework. The coursework was developed in Apache NetBeans with help of many GUI elements, pre-defined library classes and methods to make the GUI attractive and function-able. The project deals with the Car Accessories Details Information system. The features of our Car Accessories Information System includes adding new accessories items, clear the selected items, search for accessories based on price and category and open any file from the desktop interface when open menu item is selected.

To achieve the objective of our coursework, we faced many difficulties which took us a lot of effort to solve those problems. Firstly, coordinating with fellow members was difficult and due to clashing schedules, it was more difficult to carry on with the project regularly. During the development of the project, we faced many challenges on different stages of the coursework. The first problem that we faced was binary search algorithm to search items based on price due to error in the methods which was called by our binary search method. Similarly, making buttons function-able was the most challenging part of the coursework. Similarly, due to lack of enough exception handling, the program showed many errors when we entered value of wrong data type in the text fields. To solve all these problems, we discussed a lot with fellow group members, revised the lecture slides, tried different codes and asked for suggestions from the module leader and implemented different exception handlings and test cases to try and minimize the errors as much as possible.

This project was very fruitful for us since it helped to increase our capability to work and cope with the problems as a group that arose during the development of the project. Working in a group helped all of us to develop a strong team mentality, which will surely help us a lot in the future when we will work in a real software company with different other employees. From this coursework, we gained a lot of information on how carry out a real-world based project and to develop a car accessories details information system.

7. References

tutorialspoint. (2020). Retrieved January 02, 2020, from https://www.tutorialspoint.com/data_structures_algorithms/binary_search_algorithm.htm

8. Appendix

```
import java.awt.Color;

import java.awt.Desktop;

import java.awt.Toolkit;

import java.io.File;

import java.io.IOException;

import java.util.ArrayList;

import javax.swing.BorderFactory;

import javax.swing.JOptionPane;

import javax.swing.border.TitledBorder;

import javax.swing.table.DefaultTableModel;

import java.util.Arrays;

import javax.swing.JFileChooser;

/*

 * 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.

 */

/**

 *

 * @author Kushal Gurung 18029022

 * @author Sanjit Adhikari 18029085

 * @author Sadikshya Baral 18029069

 * @author Hrithik Sunar 18029017
 * @author Sagar Nepali 18029075
 */
public class CAD_Info extends javax.swing.JFrame
{
    DefaultTableModel model;
    ArrayList<Object> idNumberArrayList = new ArrayList<>();
    //Constructor method
    public CAD_Info()
    {
        initComponents();
        model = (DefaultTableModel) jTable1.getModel();
        //multi dimensional array
        Object[][] array =
        {
            {1, "Comfort", "Armrest", "AR01", "Low", "735", "Company, Community"},
            {2, "Comfort", "Blind spot mirrors", "BSM02", "Low", "388", "No Recommendation"},
            {3, "Comfort", "Sun reflector", "SR03", "Low", "344", "Company"},
            {4, "Body Kit", "Spoilers", "S04", "Medium", "3000", "Community"},
            {5, "Body Kit", "Bumper Lips", "BL05", "Low", "1450", "No Recommendation"},
            {6, "Body Kit", "Roof Scoops", "RS06", "High", "5800", "Company"},
            {7, "Media", "In-Dash DVD/CD Monitor", "IDCM07", "High", "7000", "Community"},
        {8, "Media", "Rearview Backup Camera", "RBC08", "Low", "1900",
"Company"},

        {9, "Media", "Rearview Mirror Monitor", "RMM09", "High", "5000",
"Community"},

        {10, "Sensors", "Airbag sensors", "AS10", "High", "6500", "Company"},

        {11, "Sensors", "Coolant temperature sensor", "CTS11", "Low", "1950",
"Company"},

        {12, "Sensors", "Fuel level sensor", "FLS12", "High", "1450",
"Company"},

        {13, "Gauge & meters", "Fuel Gauge", "FG13", "Low", "800", "No
Recommendation"},

        {14, "Gauge & meters", "Speedometer", "S14", "Low", "1000",
"Company, Community"},

        {15, "Gauge & meters", "Odometer", "O15", "Low", "800", "Company"}

    };


    //Addition of object data type values in array.

    for (Object [] rowValues : array)

    {

        model.addRow(rowValues);

        idNumberArrayList.add(rowValues);

    }


    //Creation of a titled border and setting the border into a panel.

    jPanel1.setBorder(BorderFactory.createTitledBorder

    (
            BorderFactory.createEtchedBorder(), "Car Accessories Details" , TitledBorder.CENTER, TitledBorder.TOP)
      );
      //Setting white font color of titled border.
      TitledBorder titledBorder = (TitledBorder)jPanel1.getBorder();
      titledBorder.setTitleColor(Color.WHITE);
   }
   /**
    * This method is called from within the constructor to initialize the form.
    * WARNING: Do NOT modify this code. The content of this method is always
    * regenerated by the Form Editor.
    */

    public void sort(int [] a)
    {
    // This method will sort the value of an array which is passed as an argument in this method's parameter.
        for(int i=0;i<a.length;i++)
      {
        int minVal = minimumValue(a,i);
        swap(a,minVal,i);
      }
    }

   public int minimumValue(int [] a, int form)
   {
    //This method finds the minimum value by comparing one number with another.
        int minVal = form;
        for(int i = form +1;i<a.length;i++)
      {
         if(a[i]<a[minVal])
         {
                minVal = i;
         }
        }
      return minVal;
   }


   public void swap(int [] a,int i,int j)
   {
   //This method will swap the position of smaller and greater number. It keeps smaller number at first.
        int temp= a[i];
        a[i] = a[j];
        a[j] = temp;
   }



   public int binarySearch(int [] a, int key)
```

```
  {
  //This is a binary search method which will be called later while searching for
accessories through price.
      sort(a);
      int size = a.length;
      int low = 0;
      int high = size-1;
      while (low <= high)
      {
          int mid = (low + high) / 2;
          if (a[mid] == key)
          {
              return a[mid];
          }
          else if (key > a[mid])
          {
              low = mid + 1;
          }
          else
          {
              high = mid - 1;
          }
       }
       return -1;
  }
```

```
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        buttonGroup1 = new javax.swing.ButtonGroup();
        jPanel5 = new javax.swing.JPanel();
        jPanel2 = new javax.swing.JPanel();
        jLabel8 = new javax.swing.JLabel();
        accessoriesNameTextField = new javax.swing.JTextField();
        accessoriesNumberTextField = new javax.swing.JTextField();
        jLabel6 = new javax.swing.JLabel();
        companyCheckBox = new javax.swing.JCheckBox();
        communityCheckBox = new javax.swing.JCheckBox();
        noRecommendationCheckBox = new javax.swing.JCheckBox();
        addButton = new javax.swing.JButton();
        jComboBox1 = new javax.swing.JComboBox<>();
        jLabel3 = new javax.swing.JLabel();
        jLabel4 = new javax.swing.JLabel();
        jLabel5 = new javax.swing.JLabel();
        jLabel7 = new javax.swing.JLabel();
        lowRadioButton = new javax.swing.JRadioButton();
        mediumRadioButton = new javax.swing.JRadioButton();
        highRadioButton = new javax.swing.JRadioButton();
        priceTextField = new javax.swing.JTextField();
```

```
jLabel2 = new javax.swing.JLabel();
clearButton = new javax.swing.JButton();
jSeparator1 = new javax.swing.JSeparator();
jPanel3 = new javax.swing.JPanel();
jLabel1 = new javax.swing.JLabel();
jLabel15 = new javax.swing.JLabel();
jPanel1 = new javax.swing.JPanel();
jScrollPane1 = new javax.swing.JScrollPane();
jTable1 = new javax.swing.JTable();
jPanel4 = new javax.swing.JPanel();
jLabel9 = new javax.swing.JLabel();
priceSearchTextField = new javax.swing.JTextField();
priceSearchButton = new javax.swing.JButton();
jLabel10 = new javax.swing.JLabel();
categoryComboBox = new javax.swing.JComboBox<>();
categorySearchButton = new javax.swing.JButton();
jLabel11 = new javax.swing.JLabel();
priceSearchClearButton = new javax.swing.JButton();
clearSelectedButton = new javax.swing.JButton();
jLabel12 = new javax.swing.JLabel();
jSeparator2 = new javax.swing.JSeparator();
jLabel13 = new javax.swing.JLabel();
jSeparator3 = new javax.swing.JSeparator();
jSeparator4 = new javax.swing.JSeparator();
```

```
        jLabel14 = new javax.swing.JLabel();
        jSeparator5 = new javax.swing.JSeparator();
        exitButton = new javax.swing.JButton();
        jMenuBar1 = new javax.swing.JMenuBar();
        fileMenu = new javax.swing.JMenu();
        openMenuItem = new javax.swing.JMenuItem();
        exitMenuItem = new javax.swing.JMenuItem();
        helpMenu = new javax.swing.JMenu();
        viewHelpMenuItem = new javax.swing.JMenuItem();
        aboutMenuItem = new javax.swing.JMenuItem();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        setTitle("Car Accessories Information System");

        jPanel5.setBackground(new java.awt.Color(43, 81, 95));

        jPanel2.setBackground(new java.awt.Color(0, 140, 153));
        jPanel2.setBorder(new javax.swing.border.LineBorder(new java.awt.Color(47, 79, 79), 1, true));
        jPanel2.setForeground(new java.awt.Color(215, 215, 215));

        jLabel8.setFont(new java.awt.Font("Arial Black", 0, 18)); // NOI18N
        jLabel8.setForeground(new java.awt.Color(255, 255, 255));
        jLabel8.setText("Add Accessories Details");
        accessoriesNameTextField.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N

        accessoriesNumberTextField.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
        accessoriesNumberTextField.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                accessoriesNumberTextFieldActionPerformed(evt);
            }
        });

        jLabel6.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
        jLabel6.setForeground(new java.awt.Color(255, 255, 255));
        jLabel6.setText("Recommendation");

        companyCheckBox.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
        companyCheckBox.setForeground(new java.awt.Color(255, 255, 255));
        companyCheckBox.setText("Company");
        companyCheckBox.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                companyCheckBoxActionPerformed(evt);
            }
```

```
    });

    communityCheckBox.setFont(new java.awt.Font("Arial", 0, 14)); //
NOI18N
    communityCheckBox.setForeground(new java.awt.Color(255, 255, 255));
    communityCheckBox.setText("Community");
    communityCheckBox.addActionListener(new
java.awt.event.ActionListener() {
      public void actionPerformed(java.awt.event.ActionEvent evt) {
        communityCheckBoxActionPerformed(evt);
      }
    });

    noRecommendationCheckBox.setFont(new java.awt.Font("Arial", 0, 14));
// NOI18N
    noRecommendationCheckBox.setForeground(new java.awt.Color(255,
255, 255));
    noRecommendationCheckBox.setText("No Recommendation");
    noRecommendationCheckBox.addActionListener(new
java.awt.event.ActionListener() {
      public void actionPerformed(java.awt.event.ActionEvent evt) {
        noRecommendationCheckBoxActionPerformed(evt);
      }
    });

    addButton.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
```

```
        addButton.setText("Add");
        addButton.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                addButtonActionPerformed(evt);
            }
        });

        jComboBox1.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
        jComboBox1.setModel(new  javax.swing.DefaultComboBoxModel<>(new
String[] { "Comfort", "Body Kit", "Media", "Sensors", "Gauge and meters" }));
        jComboBox1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jComboBox1ActionPerformed(evt);
            }
        });

        jLabel3.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
        jLabel3.setForeground(new java.awt.Color(255, 255, 255));
        jLabel3.setText("Accessories Name");

        jLabel4.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
        jLabel4.setForeground(new java.awt.Color(255, 255, 255));
        jLabel4.setText("Range Level");

        jLabel5.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
```

```
jLabel5.setForeground(new java.awt.Color(255, 255, 255));
jLabel5.setText("Price");

jLabel7.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
jLabel7.setForeground(new java.awt.Color(255, 255, 255));
jLabel7.setText("Accessories Number");

buttonGroup1.add(lowRadioButton);
lowRadioButton.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
lowRadioButton.setForeground(new java.awt.Color(255, 255, 255));
lowRadioButton.setText("Low");
lowRadioButton.setFocusCycleRoot(true);
lowRadioButton.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        lowRadioButtonActionPerformed(evt);
    }
});

buttonGroup1.add(mediumRadioButton);
mediumRadioButton.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
mediumRadioButton.setForeground(new java.awt.Color(255, 255, 255));
mediumRadioButton.setText("Medium");
mediumRadioButton.setFocusCycleRoot(true);
```

```
buttonGroup1.add(highRadioButton);
highRadioButton.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
highRadioButton.setForeground(new java.awt.Color(255, 255, 255));
highRadioButton.setText("High");
highRadioButton.setFocusCycleRoot(true);

priceTextField.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
priceTextField.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        priceTextFieldActionPerformed(evt);
    }
});

jLabel2.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
jLabel2.setForeground(new java.awt.Color(255, 255, 255));
jLabel2.setText("Category");

clearButton.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
clearButton.setText("Clear");
clearButton.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        clearButtonActionPerformed(evt);
    }
});
```

```
        javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);
        jPanel2.setLayout(jPanel2Layout);
        jPanel2Layout.setHorizontalGroup(
jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel2Layout.createSequentialGroup()
                .addGap(27, 27, 27)
.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addGroup(jPanel2Layout.createSequentialGroup()
.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
                                .addComponent(jSeparator1)
                                .addComponent(jLabel8, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                                .addComponent(jLabel6, javax.swing.GroupLayout.Alignment.LEADING, javax.swing.GroupLayout.PREFERRED_SIZE, 127, javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
.addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAI
LING)
                        .addGroup(jPanel2Layout.createSequentialGroup()
                          .addComponent(addButton,
javax.swing.GroupLayout.PREFERRED_SIZE,                              69,
javax.swing.GroupLayout.PREFERRED_SIZE)
                          .addGap(62, 62, 62)
                          .addComponent(clearButton,
javax.swing.GroupLayout.PREFERRED_SIZE,                              73,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.TRAILING)
                          .addComponent(jLabel2,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.PREFERRED_SIZE,                             115,
javax.swing.GroupLayout.PREFERRED_SIZE)
                          .addComponent(jComboBox1,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.PREFERRED_SIZE,                             163,
javax.swing.GroupLayout.PREFERRED_SIZE)
                          .addComponent(jLabel3,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.PREFERRED_SIZE,                             132,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAI
LING, false)
                                        .addComponent(accessoriesNameTextField,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, 204, Short.MAX_VALUE)
                                        .addComponent(accessoriesNumberTextField,
javax.swing.GroupLayout.Alignment.LEADING))
                                    .addComponent(jLabel4,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.PREFERRED_SIZE,                              127,
javax.swing.GroupLayout.PREFERRED_SIZE)
                                    .addComponent(jLabel7,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.PREFERRED_SIZE,                              144,
javax.swing.GroupLayout.PREFERRED_SIZE)
                                    .addComponent(priceTextField,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.PREFERRED_SIZE,                              204,
javax.swing.GroupLayout.PREFERRED_SIZE))))
                          .addGroup(jPanel2Layout.createSequentialGroup()
                                 .addComponent(lowRadioButton,
javax.swing.GroupLayout.PREFERRED_SIZE,                               61,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELAT
ED)
                                 .addComponent(mediumRadioButton)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
                        .addComponent(highRadioButton,
javax.swing.GroupLayout.PREFERRED_SIZE,                                   60,
javax.swing.GroupLayout.PREFERRED_SIZE)))
                  .addContainerGap(20, Short.MAX_VALUE))
               .addGroup(jPanel2Layout.createSequentialGroup()

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.LEADING)
                     .addComponent(jLabel5,
javax.swing.GroupLayout.PREFERRED_SIZE,                                   64,
javax.swing.GroupLayout.PREFERRED_SIZE)
                     .addGroup(jPanel2Layout.createSequentialGroup()
                        .addComponent(communityCheckBox)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                        .addComponent(companyCheckBox,
javax.swing.GroupLayout.PREFERRED_SIZE,                                   85,
javax.swing.GroupLayout.PREFERRED_SIZE)))
                  .addGap(0, 0, Short.MAX_VALUE))))
         .addGroup(jPanel2Layout.createSequentialGroup()
            .addGap(39, 39, 39)
            .addComponent(noRecommendationCheckBox)
            .addGap(0, 0, Short.MAX_VALUE))
      );
      jPanel2Layout.setVerticalGroup(
jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
          .addGroup(jPanel2Layout.createSequentialGroup()
             .addContainerGap()
             .addComponent(jLabel8,
javax.swing.GroupLayout.PREFERRED_SIZE,                          31,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
             .addComponent(jSeparator1,
javax.swing.GroupLayout.PREFERRED_SIZE,                           6,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
             .addComponent(jLabel7)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
             .addComponent(accessoriesNumberTextField,
javax.swing.GroupLayout.PREFERRED_SIZE,                          34,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELAT
ED)
             .addComponent(jLabel3,
javax.swing.GroupLayout.PREFERRED_SIZE,                          19,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
            .addComponent(accessoriesNameTextField,
javax.swing.GroupLayout.PREFERRED_SIZE, 34,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELAT
ED)
            .addComponent(jLabel2)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELAT
ED)
            .addComponent(jComboBox1,
javax.swing.GroupLayout.PREFERRED_SIZE, 31,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELAT
ED)
            .addComponent(jLabel4)
            .addGap(16, 16, 16)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.BASELINE)
               .addComponent(lowRadioButton)
               .addComponent(mediumRadioButton)
               .addComponent(highRadioButton))
            .addGap(14, 14, 14)
            .addComponent(jLabel5,
javax.swing.GroupLayout.PREFERRED_SIZE, 18,
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(priceTextField,
javax.swing.GroupLayout.PREFERRED_SIZE,                    34,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELAT
ED)
        .addComponent(jLabel6)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELAT
ED)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.BASELINE)
          .addComponent(communityCheckBox)
          .addComponent(companyCheckBox))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(noRecommendationCheckBox)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.BASELINE)
          .addComponent(clearButton,
javax.swing.GroupLayout.PREFERRED_SIZE,                    34,
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
                .addComponent(addButton,
javax.swing.GroupLayout.PREFERRED_SIZE,                                    35,
javax.swing.GroupLayout.PREFERRED_SIZE))
             .addGap(34, 34, 34))
      );


      jPanel3.setBackground(new java.awt.Color(0, 140, 153));
      jPanel3.setBorder(new
javax.swing.border.SoftBevelBorder(javax.swing.border.BevelBorder.RAISED
));


      jLabel1.setBackground(new java.awt.Color(255, 255, 255));
      jLabel1.setFont(new java.awt.Font("Arial Black", 0, 24)); // NOI18N
      jLabel1.setForeground(new java.awt.Color(255, 255, 255));
      jLabel1.setText("Car Accessories Information System");


      javax.swing.GroupLayout          jPanel3Layout          =          new
javax.swing.GroupLayout(jPanel3);
      jPanel3.setLayout(jPanel3Layout);
      jPanel3Layout.setHorizontalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
          .addGroup(jPanel3Layout.createSequentialGroup()
             .addContainerGap()
```

```
            .addComponent(jLabel1,
javax.swing.GroupLayout.PREFERRED_SIZE,                                516,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jLabel15,
javax.swing.GroupLayout.PREFERRED_SIZE,                                 70,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
      );
      jPanel3Layout.setVerticalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
         .addComponent(jLabel1,   javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
         .addComponent(jLabel15,  javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
      );

      jPanel1.setBackground(new java.awt.Color(0, 140, 153));

      jTable1.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
      jTable1.setModel(new javax.swing.table.DefaultTableModel(
         new Object [][] {
         },
         new String [] {
            "ID", "Category", " Name", " Number", "Range Level", "Price ", "Recommendation"
         }
      ));
      jScrollPane1.setViewportView(jTable1);
      if (jTable1.getColumnModel().getColumnCount() > 0) {
         jTable1.getColumnModel().getColumn(0).setMaxWidth(60);
         jTable1.getColumnModel().getColumn(1).setMinWidth(130);
         jTable1.getColumnModel().getColumn(1).setMaxWidth(130);
         jTable1.getColumnModel().getColumn(2).setMinWidth(200);
         jTable1.getColumnModel().getColumn(2).setMaxWidth(300);
         jTable1.getColumnModel().getColumn(3).setMinWidth(80);
         jTable1.getColumnModel().getColumn(3).setMaxWidth(90);
         jTable1.getColumnModel().getColumn(4).setMinWidth(90);
         jTable1.getColumnModel().getColumn(4).setMaxWidth(100);
         jTable1.getColumnModel().getColumn(5).setMinWidth(80);
         jTable1.getColumnModel().getColumn(5).setMaxWidth(90);
         jTable1.getColumnModel().getColumn(6).setMinWidth(150);
         jTable1.getColumnModel().getColumn(6).setMaxWidth(170);
      }

      javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
```

```
        jPanel1.setLayout(jPanel1Layout);
        jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jScrollPane1,
javax.swing.GroupLayout.DEFAULT_SIZE, 792, Short.MAX_VALUE)
                .addContainerGap())
        );
        jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jScrollPane1)
                .addContainerGap())
        );

        jPanel4.setBackground(new java.awt.Color(0, 140, 153));

        jLabel9.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
        jLabel9.setForeground(new java.awt.Color(255, 255, 255));
        jLabel9.setText("Price");


        priceSearchTextField.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                priceSearchTextFieldActionPerformed(evt);
            }
        });


        priceSearchButton.setFont(new java.awt.Font("Tahoma", 0, 14)); //
NOI18N
        priceSearchButton.setText("Search");
        priceSearchButton.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                priceSearchButtonActionPerformed(evt);
            }
        });


        jLabel10.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
        jLabel10.setForeground(new java.awt.Color(255, 255, 255));
        jLabel10.setText("Category");


        categoryComboBox.setFont(new java.awt.Font("Tahoma", 0, 14)); //
NOI18N
```

```
        categoryComboBox.setModel(new
javax.swing.DefaultComboBoxModel<>(new String[] { "Comfort", "Body Kit",
"Media", "Sensors", "Gauge and meters" }));


        categorySearchButton.setFont(new java.awt.Font("Arial", 0, 14)); //
NOI18N
        categorySearchButton.setText("Search Category");
        categorySearchButton.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                categorySearchButtonActionPerformed(evt);
            }
        });


        jLabel11.setFont(new java.awt.Font("Arial", 0, 18)); // NOI18N
        jLabel11.setForeground(new java.awt.Color(255, 255, 255));
        jLabel11.setText("Clear Details From Table");


        priceSearchClearButton.setFont(new java.awt.Font("Tahoma", 0, 14)); //
NOI18N
        priceSearchClearButton.setText("Clear");
        priceSearchClearButton.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                priceSearchClearButtonActionPerformed(evt);
            }
```

```
    });

    clearSelectedButton.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
    clearSelectedButton.setText("Clear Selected");
    clearSelectedButton.addActionListener(new
java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            clearSelectedButtonActionPerformed(evt);
        }
    });

    jLabel12.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
    jLabel12.setForeground(new java.awt.Color(255, 255, 255));
    jLabel12.setText("Search From Price");

    jLabel13.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
    jLabel13.setForeground(new java.awt.Color(255, 255, 255));
    jLabel13.setText("Search From Category");

    jLabel14.setFont(new java.awt.Font("Arial", 0, 18)); // NOI18N
    jLabel14.setForeground(new java.awt.Color(255, 255, 255));
    jLabel14.setText("Exit the System");

    exitButton.setFont(new java.awt.Font("Arial", 0, 14)); // NOI18N
    exitButton.setText("Exit");
```

```
        exitButton.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                exitButtonActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout jPanel4Layout = new javax.swing.GroupLayout(jPanel4);
        jPanel4.setLayout(jPanel4Layout);
        jPanel4Layout.setHorizontalGroup(
jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel4Layout.createSequentialGroup()
                .addContainerGap()
.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addComponent(jSeparator3, javax.swing.GroupLayout.Alignment.TRAILING)
                    .addGroup(jPanel4Layout.createSequentialGroup()
.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                        .addComponent(jSeparator2)
                        .addComponent(jLabel11,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                        .addComponent(jLabel12)
                        .addComponent(jLabel9,
javax.swing.GroupLayout.PREFERRED_SIZE,                              64,
javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addComponent(categorySearchButton,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                        .addComponent(categoryComboBox,                       0,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                        .addComponent(jLabel13)
                        .addComponent(jSeparator4)
                        .addGroup(jPanel4Layout.createSequentialGroup()
                            .addComponent(priceSearchButton,
javax.swing.GroupLayout.PREFERRED_SIZE,                              94,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                            .addComponent(priceSearchClearButton,
javax.swing.GroupLayout.PREFERRED_SIZE,                              91,
javax.swing.GroupLayout.PREFERRED_SIZE))
                        .addComponent(priceSearchTextField,
javax.swing.GroupLayout.PREFERRED_SIZE,                             196,
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
                                .addComponent(jLabel10,
javax.swing.GroupLayout.PREFERRED_SIZE, 67,
javax.swing.GroupLayout.PREFERRED_SIZE)
                                .addComponent(jLabel14)
                                .addComponent(jSeparator5)
                                .addComponent(clearSelectedButton))
                            .addComponent(exitButton,
javax.swing.GroupLayout.PREFERRED_SIZE, 83,
javax.swing.GroupLayout.PREFERRED_SIZE))
                        .addGap(0, 5, Short.MAX_VALUE)))
                .addContainerGap())
        );
        jPanel4Layout.setVerticalGroup(

jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
            .addGroup(jPanel4Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jLabel12)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jSeparator2,
javax.swing.GroupLayout.PREFERRED_SIZE, 10,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jLabel9)
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(priceSearchTextField, javax.swing.GroupLayout.PREFERRED_SIZE, 33, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
               .addComponent(priceSearchButton, javax.swing.GroupLayout.PREFERRED_SIZE, 34, javax.swing.GroupLayout.PREFERRED_SIZE)
               .addComponent(priceSearchClearButton, javax.swing.GroupLayout.PREFERRED_SIZE, 35, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(25, 25, 25)
            .addComponent(jLabel13, javax.swing.GroupLayout.PREFERRED_SIZE, 24, javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jSeparator3, javax.swing.GroupLayout.PREFERRED_SIZE, 11, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(4, 4, 4)
            .addComponent(jLabel10)
            .addGap(18, 18, 18)
            .addComponent(categoryComboBox,
javax.swing.GroupLayout.PREFERRED_SIZE,                                  36,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addComponent(categorySearchButton,
javax.swing.GroupLayout.PREFERRED_SIZE,                                  34,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addComponent(jLabel11)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jSeparator4,
javax.swing.GroupLayout.PREFERRED_SIZE,                                  11,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELAT
ED)
            .addComponent(clearSelectedButton,
javax.swing.GroupLayout.PREFERRED_SIZE,                                  34,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addComponent(jLabel14)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jSeparator5,
javax.swing.GroupLayout.PREFERRED_SIZE,                                  10,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
            .addComponent(exitButton,
javax.swing.GroupLayout.PREFERRED_SIZE, 35,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(23, Short.MAX_VALUE))
      );


      javax.swing.GroupLayout jPanel5Layout = new
javax.swing.GroupLayout(jPanel5);
      jPanel5.setLayout(jPanel5Layout);
      jPanel5Layout.setHorizontalGroup(

jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
         .addGroup(jPanel5Layout.createSequentialGroup()
            .addContainerGap()

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
               .addComponent(jPanel3,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
               .addGroup(jPanel5Layout.createSequentialGroup()
                  .addComponent(jPanel4,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                  .addGap(14, 14, 14)
                  .addComponent(jPanel1,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELAT
ED)
                  .addComponent(jPanel2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
            .addContainerGap())
      );
      jPanel5Layout.setVerticalGroup(

jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
         .addGroup(jPanel5Layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jPanel3,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELAT
ED)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.LEADING)
```

```
                .addComponent(jPanel1,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jPanel2,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jPanel4,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
            .addContainerGap())
      );


      fileMenu.setText("File");



openMenuItem.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.
event.KeyEvent.VK_O, java.awt.event.InputEvent.CTRL_MASK));
      openMenuItem.setText("Open");
      openMenuItem.addActionListener(new java.awt.event.ActionListener() {
         public void actionPerformed(java.awt.event.ActionEvent evt) {
            openMenuItemActionPerformed(evt);
         }
      });
      fileMenu.add(openMenuItem);
```

```
exitMenuItem.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.e
vent.KeyEvent.VK_E, java.awt.event.InputEvent.CTRL_MASK));
        exitMenuItem.setText("Exit");
        exitMenuItem.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                exitMenuItemActionPerformed(evt);
            }
        });
        fileMenu.add(exitMenuItem);

        jMenuBar1.add(fileMenu);

        helpMenu.setText("Help");


viewHelpMenuItem.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.
awt.event.KeyEvent.VK_H, java.awt.event.InputEvent.CTRL_MASK));
        viewHelpMenuItem.setText("View Help");
        viewHelpMenuItem.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                viewHelpMenuItemActionPerformed(evt);
            }
        });
        helpMenu.add(viewHelpMenuItem);
```

```
aboutMenuItem.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK_A, java.awt.event.InputEvent.ALT_MASK));
        aboutMenuItem.setText("About");
        aboutMenuItem.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                aboutMenuItemActionPerformed(evt);
            }
        });
        helpMenu.add(aboutMenuItem);

        jMenuBar1.add(helpMenu);

        setJMenuBar(jMenuBar1);

        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel5, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        );
        layout.setVerticalGroup(
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel5,   javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        );


        pack();
     }// </editor-fold>


     private void jComboBox1ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
     }


     private                                                            void
communityCheckBoxActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        //When community check box is selected, no recommendation button will
be set to false.
        noRecommendationCheckBox.setSelected(false);
     }


     private void exitButtonActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
            this.dispose(); //closes the information system application.
     }
```

```
    private void lowRadioButtonActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }

    //This method will add items in the table.
    private void addButtonActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        String range = "";
        String recommendation = "";
        String category = jComboBox1.getSelectedItem().toString();
        String accessoriesName = accessoriesNameTextField.getText();
        String accessoriesNumber = accessoriesNumberTextField.getText();
        String price = priceTextField.getText();

        if (lowRadioButton.isSelected())
        {
            range = lowRadioButton.getText().toString();
        }
        if (mediumRadioButton.isSelected())
        {
            range = mediumRadioButton.getText().toString();
        }
        if (highRadioButton.isSelected())
```

```
{
    range = highRadioButton.getText().toString();
}
if (companyCheckBox.isSelected())
{
    recommendation = String.valueOf("Company");
}
if (communityCheckBox.isSelected())
{
    recommendation = String.valueOf("Community");
}
if (noRecommendationCheckBox.isSelected())
{
    recommendation = String.valueOf("No Recommendation");
}
if (companyCheckBox.isSelected() && communityCheckBox.isSelected())
{
    recommendation = String.valueOf("Company, Community");
}
try
{
    if(accessoriesNumber.equals("") && accessoriesName.equals("") &&
range.equals("") && price.equals("") && recommendation.equals(""))
    {
        Toolkit.getDefaultToolkit().beep();
```

```
            JOptionPane.showMessageDialog(this, "Please enter all the value.", "Error!", JOptionPane.ERROR_MESSAGE);
        }
        else if(accessoriesNumber.equals(""))
        {
            Toolkit.getDefaultToolkit().beep();
            JOptionPane.showMessageDialog(this, "Please enter the accessories number.", "Error!", JOptionPane.ERROR_MESSAGE);
        }
        else if(accessoriesName.equals(""))
        {
            Toolkit.getDefaultToolkit().beep();
            JOptionPane.showMessageDialog(this, "Please enter the accessories name.", "Error!", JOptionPane.ERROR_MESSAGE);
        }
        else if (category.equals(""))
        {
            Toolkit.getDefaultToolkit().beep();
            JOptionPane.showMessageDialog(this, "Please select a accessories category.", "Error!", JOptionPane.ERROR_MESSAGE);
        }
        else if(range.equals(""))
        {
            Toolkit.getDefaultToolkit().beep();
            JOptionPane.showMessageDialog(this, "Please select a accessories prize range.", "Error!", JOptionPane.ERROR_MESSAGE);
```

```
}
else if(price.equals(""))
{
   Toolkit.getDefaultToolkit().beep();
   JOptionPane.showMessageDialog(this, "Please enter the accessories price.", "Error!", JOptionPane.ERROR_MESSAGE);
}
else if(Integer.parseInt(price)<0)
{
   Toolkit.getDefaultToolkit().beep();
   JOptionPane.showMessageDialog(this, "Please enter the correct accessories price.", "Error!", JOptionPane.ERROR_MESSAGE);
}
else if(recommendation.equals(""))
{
   Toolkit.getDefaultToolkit().beep();
   JOptionPane.showMessageDialog(this, "Please select the accessories recommendations.", "Error!", JOptionPane.ERROR_MESSAGE);
}
else
{  //else condition will run when all the values are added and selected in text fields, combox boxes and radio buttons.
   int id = idNumberArrayList.size()+1;   //It stores the id number which will be added by 1, every time a new accessory is added.
   String idNumber = String.valueOf(id);
```

```
                String [] arrayAdd = {idNumber, category, accessoriesName, accessoriesNumber, range, price, recommendation};

                model.addRow(arrayAdd);

                idNumberArrayList.add(arrayAdd);

                JOptionPane.showMessageDialog(this, "Accessory is successfully added.", "Result", JOptionPane.INFORMATION_MESSAGE);

            }

        }

        catch(NumberFormatException exceptionVariable) {

            Toolkit.getDefaultToolkit().beep();

                JOptionPane.showMessageDialog(this, "Please enter correct value in price", "Error!", JOptionPane.ERROR_MESSAGE);

        }

    }


    private void noRecommendationCheckBoxActionPerformed(java.awt.event.ActionEvent evt) {

        // TODO add your handling code here:

        //When no recommendation check box is selected, company and community check box will be set to false.

        companyCheckBox.setSelected(false);

        communityCheckBox.setSelected(false);

    }
    private void
companyCheckBoxActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        //When company check box is selected, no recommendation check box is
set to false.
        noRecommendationCheckBox.setSelected(false);
    }

    private void
priceSearchTextFieldActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }

    private void priceSearchButtonActionPerformed(java.awt.event.ActionEvent
evt) {
        // TODO add your handling code here
        //Search by price:
        try
        {
        int rowCount = jTable1.getRowCount();
        String search = priceSearchTextField.getText();
        if(!"".equals(search) && rowCount!=0)
        {
            int searchItem = Integer.parseInt(priceSearchTextField.getText());
//The value that we enter in the price search text field.
            int data[] = new int[rowCount];
```

```
for(int i=0; i<rowCount; i++)

{

   data [i] = Integer.parseInt(jTable1.getValueAt(i, 5).toString()); //Since our price is on 5th position, data [i] will get the value of 5th position.

}

int searchResult = binarySearch(data, searchItem);     //binary search method is called.

if (searchResult != -1)

{

   for (int i = 0; i < rowCount; i++)

   {

      if (Integer.parseInt(jTable1.getValueAt(i, 5).toString()) == searchResult)

      {

         int itemIndex = 2;

         String itemName = jTable1.getValueAt(i, itemIndex).toString(); //Our accessories name is stored at 2nd position, itemIndex value is assigned with 2.

         JOptionPane.showMessageDialog(this, "Search Found!\nPrice: Rs." + searchItem + "\nAccessories: " + itemName, "Result", JOptionPane.INFORMATION_MESSAGE);

         break;

      }

   }

}

else
```

```
            {
                Toolkit.getDefaultToolkit().beep();
                JOptionPane.showMessageDialog(this, "No accessories are
available for this price.", "Result", JOptionPane.INFORMATION_MESSAGE);
            }
        }
        else
        {
            Toolkit.getDefaultToolkit().beep();
            JOptionPane.showMessageDialog(this, "Please enter price.", "Error",
JOptionPane.WARNING_MESSAGE);
        }
        }
        catch(NumberFormatException exception)
        {
            Toolkit.getDefaultToolkit().beep();
            JOptionPane.showMessageDialog(this, "Please enter price amount in
whole number.","Error", JOptionPane.WARNING_MESSAGE);
        }
    }
    private void
categorySearchButtonActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        //search by category.
        String categoryItems = categoryComboBox.getSelectedItem().toString();
    int searchIndex1 = 1;

    int rows1 = jTable1.getRowCount();

    int no = 1;

    String displayName="";

    for (int i = 0; i < rows1; i++)

    {

      if ((jTable1.getValueAt(i,
searchIndex1).toString()).equals(categoryItems))

      {

      //searchIndex1 is assigned with 1 since our category is at 1st position
of the array.

        int itemIndex = 2;

        displayName = displayName+ no + "." + ((String)
jTable1.getValueAt(i, itemIndex)) + "\n";

        no++;

      }

    }

    if (!"".equals(displayName))

    {

      JOptionPane.showMessageDialog(this, displayName, (no-1) + "
accessories found in " + categoryItems+" category.",
JOptionPane.INFORMATION_MESSAGE);

    }

    else

    {
      JOptionPane.showMessageDialog(this, "No Items Related to Category: 
" + categoryItems, "Message", JOptionPane.INFORMATION_MESSAGE);
    }
}
private void exitMenuItemActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    this.dispose();       //closes the information system.
}
private void openMenuItemActionPerformed(java.awt.event.ActionEvent 
evt) {
    // TODO add your handling code here:
    //Opens any file from Desktop interface.
    JFileChooser filechooser = new JFileChooser();
    int i = filechooser.showOpenDialog(this);
      if(i == JFileChooser.APPROVE_OPTION)
      {
        File file = filechooser.getSelectedFile();
            //Opening file
            //Checks if desktop interaction is available or not from this platform.
            if(Desktop.isDesktopSupported())
            {
                try
                {
                File openFile = new File(file.getAbsolutePath());
```

```
                  Desktop.getDesktop().open(openFile);   //opens the absolute
form of this abstract pathname.
                  }
                  catch (IOException iOException)
                  {
                     //Showing the path in a dialog box.
                  JOptionPane.showMessageDialog(this, "No application
available to open the file you chose.\n"
                        + "Chosen File location: \n" + file.getAbsolutePath(),
"Error!", JOptionPane.PLAIN_MESSAGE);
                  }
                  catch(Exception filenfException)
                  {
                     JOptionPane.showMessageDialog(this, "There was an error
while accessing the file.\n", "Error!", JOptionPane.PLAIN_MESSAGE);
                  }
               }
               else
               {
                   Toolkit.getDefaultToolkit().beep();
               JOptionPane.showMessageDialog(this, "Desktop interaction was
denied.", "Information", JOptionPane.INFORMATION_MESSAGE);
               }
         }
         else
         {
            //If user doesn't select files and click exits in the file chooser dialog
box.
            Toolkit.getDefaultToolkit().beep();
            JOptionPane.showMessageDialog(this, "No file was selected.",
"Information", JOptionPane.INFORMATION_MESSAGE);
         }
   }

   private void aboutMenuItemActionPerformed(java.awt.event.ActionEvent
evt) {
      // TODO add your handling code here:
      Toolkit.getDefaultToolkit().beep();
      JOptionPane.showMessageDialog(this, "Car Accessories Information
System version 1.0 \n"
            +"Developed by: \n"
            +"1. Kushal Gurung \n"
            +"2. Sanjit Adhikari \n"
            +"3. Sadikshya Baral \n"
            +"4. Sagar Nepali \n"
            +"5. Hrithik Sunar"
            , "About", JOptionPane.INFORMATION_MESSAGE);
   }

   private void
viewHelpMenuItemActionPerformed(java.awt.event.ActionEvent evt) {
      // TODO add your handling code here:
      JOptionPane.showMessageDialog(this, "Learn to use Car Accessories Information System: \n"

      +"1. To add new accessories items, fill all the text fields, select category, range level and recommendations. Then, click on (Add) Button. \n"

      +"2. To seach accessory of a particular price, enter price amount and click on (Search accessories) button. \n"

      +"3. To search accessories of a particular category, select the category option and then click on (Accessories available in category) button. \n"

      +"4. To open any file from Car Accessories Information System, press Ctrl+O or select (Open) menu item from (File) menu. \n"

      +"5. To clear all the previously entered values in the text fields, click on (Clear) button. \n"

      +"6. To know about the group members(developers) of this information system, press Alt+A or select (About) menu item from (Help) menu. \n"

      +"7. To remove a selected accessory details in the table, select a accessory in the table and then click on (Clear Selected) button. \n"

      +"8. To remove all the accessories details in the table, click on (Clear all) button. \n"

      +"9. To close the application, click on (Exit) button or press Ctrl+E button. You can also exit by selecting (Exit) menu item of (File) menu.",

       "Help", JOptionPane.PLAIN_MESSAGE);

   }


   private void clearButtonActionPerformed(java.awt.event.ActionEvent evt) {

      // TODO add your handling code here:

      //It clears the values entered in text fields and the values selected in check boxes and radio button.
      accessoriesNameTextField.setText("");

      accessoriesNumberTextField.setText("");

      priceTextField.setText("");

      buttonGroup1.clearSelection();

      companyCheckBox.setSelected(false);

      communityCheckBox.setSelected(false);

      noRecommendationCheckBox.setSelected(false);

   }


   private                                                                    void
priceSearchClearButtonActionPerformed(java.awt.event.ActionEvent evt) {

      // TODO add your handling code here:

      //Clears the price entered by user.

     priceSearchTextField.setText("");

   }


   private                                                                    void
clearSelectedButtonActionPerformed(java.awt.event.ActionEvent evt) {

      // TODO add your handling code here:

      //Removes the selected accessories item from the table.

      int [] rows = jTable1.getSelectedRows();

      for(int i=0; i<rows.length; i++)

      {

          model.removeRow(rows[i]);

      }
        JOptionPane.showMessageDialog(this, "Selected row has been deleted.",
"Result", JOptionPane.INFORMATION_MESSAGE);

    }


    private void
accessoriesNumberTextFieldActionPerformed(java.awt.event.ActionEvent
evt) {

        // TODO add your handling code here:

    }


    private void priceTextFieldActionPerformed(java.awt.event.ActionEvent evt)
{

        // TODO add your handling code here:

    }

private void accessoriesNameTextFieldKeyTyped(java.awt.event.KeyEvent
evt) {

        // TODO add your handling code here:

        //This method will not allow to enter integer value in accessories name.

        char obj = evt.getKeyChar();

        if(!(Character.isAlphabetic(obj))) {

            getToolkit().beep();

            evt.consume();

        }

    }


    /**
```

```
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
         * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {
            java.util.logging.Logger.getLogger(CAD_Info.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {
            java.util.logging.Logger.getLogger(CAD_Info.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(CAD_Info.class.getName()).log(java.util.log
ging.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(CAD_Info.class.getName()).log(java.util.log
ging.Level.SEVERE, null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {

            public void run() {
                new CAD_Info().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify
    private javax.swing.JMenuItem aboutMenuItem;
    private javax.swing.JTextField accessoriesNameTextField;
    private javax.swing.JTextField accessoriesNumberTextField;
    private javax.swing.JButton addButton;
    private javax.swing.ButtonGroup buttonGroup1;
    private javax.swing.JComboBox<String> categoryComboBox;
    private javax.swing.JButton categorySearchButton;
    private javax.swing.JButton clearButton;
    private javax.swing.JButton clearSelectedButton;
    private javax.swing.JCheckBox communityCheckBox;
    private javax.swing.JCheckBox companyCheckBox;
    private javax.swing.JButton exitButton;
    private javax.swing.JMenuItem exitMenuItem;
    private javax.swing.JMenu fileMenu;
    private javax.swing.JMenu helpMenu;
    private javax.swing.JRadioButton highRadioButton;
    private javax.swing.JComboBox<String> jComboBox1;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel10;
    private javax.swing.JLabel jLabel11;
    private javax.swing.JLabel jLabel12;
    private javax.swing.JLabel jLabel13;
    private javax.swing.JLabel jLabel14;
    private javax.swing.JLabel jLabel15;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JLabel jLabel5;
    private javax.swing.JLabel jLabel6;

    private javax.swing.JLabel jLabel7;

    private javax.swing.JLabel jLabel8;

    private javax.swing.JLabel jLabel9;

    private javax.swing.JMenuBar jMenuBar1;

    private javax.swing.JPanel jPanel1;

    private javax.swing.JPanel jPanel2;

    private javax.swing.JPanel jPanel3;

    private javax.swing.JPanel jPanel4;

    private javax.swing.JPanel jPanel5;

    private javax.swing.JScrollPane jScrollPane1;

    private javax.swing.JSeparator jSeparator1;

    private javax.swing.JSeparator jSeparator2;

    private javax.swing.JSeparator jSeparator3;

    private javax.swing.JSeparator jSeparator4;

    private javax.swing.JSeparator jSeparator5;

    private javax.swing.JTable jTable1;

    private javax.swing.JRadioButton lowRadioButton;

    private javax.swing.JRadioButton mediumRadioButton;

    private javax.swing.JCheckBox noRecommendationCheckBox;

    private javax.swing.JMenuItem openMenuItem;

    private javax.swing.JButton priceSearchButton;

    private javax.swing.JButton priceSearchClearButton;

    private javax.swing.JTextField priceSearchTextField;
    private javax.swing.JTextField priceTextField;

    private javax.swing.JMenuItem viewHelpMenuItem;

    // End of variables declaration

}
```