

```

1  package App;
2
3  import java.io.BufferedReader;
4  import java.io.FileReader;
5  import java.io.IOException;
6  import java.util.Set;
7  import java.util.HashSet;
8
9  public class Main{
10     public static void main(String[] args) {
11         new Menu();
12     }
13
14 }
15
16 public class Menu {
17     int x;
18
19     public Menu() {
20         System.out.println("\t=====");
21         System.out.println("\t\t\tMenu");
22         System.out.println("\t=====");
23         System.out.println("\t'→Navigate by inserting index←'");
24         System.out.println("\t1. Browse parts");
25         System.out.println("\t2. Find nearest dealership");
26         System.out.println("\t3. Contact");
27         System.out.println("\t4. Admin login");
28         System.out.println("\t=====");
29         System.out.print("\tInsert number between 1 to 4: ");
30
31         x = AppScanner.nextInt();
32
33         while (x < 1 || x > 4) {
34             System.out.println("\tInvalid range. \n\tPlease insert a
number between 1 to 4.");
35             System.out.print("\tSelect range: ");
36             x = AppScanner.nextInt();
37         }
38
39         System.out.println("\t-----");
40
41         switch (x) {
42             case 1:
43                 // browse parts
44
45                 System.out.println("\t=====");
46                 System.out.println("\t\tBrowsing Window");
47
48                 System.out.println("\t=====");
49                 System.out.println("\t'→Navigate by inserting
index←'");
50                 System.out.println("\t1. Browse all ");
51                 System.out.println("\t2. Browse by type ");

```

```

50         System.out.println("\t3. Browse by brand ");
51         System.out.println("\t4. Search by name");
52         System.out.println("\t5. Go back to menu");
53
54         System.out.println("\t=====");
55         System.out.print("\tInsert number between 1 to 5: ");
56
57         int x;
58
59         while (true) {
60             try {
61                 x = AppScanner.nextInt();
62                 AppScanner.nextLine(); // Consume the newline
63                 character
64                 break;
65             } catch (Exception e) {
66                 System.out.println("Invalid input. Please enter a
67                 number.");
68                 AppScanner.nextLine(); // Consume the invalid
69                 input
70             }
71         }
72
73         System.out.println("\t-----");
74
75         switch (x) {
76             case 4:
77                 // search using user input
78                 System.out.print("\tInput parts name(must be 1:1):
79                 ");
80                 String s = AppScanner.nextLine();
81
82                 System.out.println("\t-----");
83                 // Call the printPartName method from Selector
84                 class
85                 Selector.printPartName(s);
86                 break;
87             case 5:
88                 // creates new Menu instance
89                 new Menu();
90                 break;
91             default:
92                 // creates new constructor from Selector class
93                 // works when input 1 to 3
94                 new Selector(x);
95                 break;
96         }
97     }
98     break;
99 case 2:
100     new LocationFinder();
101
102     break;
103 case 3:
104     System.out.println("\tEmail : fake@fake.com");

```

```

97         System.out.println("\tCall : fake number");
98
99         new Menu();
100        break;
101        case 4:
102            // admin panel
103            new LoginVerification();
104            break;
105
106        default:
107            break;
108    }
109 }
110 }
111
112
113
114 public class Selector extends Parts {
115     int x;
116
117     public Selector(int x) {
118         super(0, "", "", "", 0, 0);
119         this.x = x;
120         switch (x) {
121             case 1:
122                 printPartAll();
123                 break;
124             case 2:
125                 printPartType();
126                 break;
127             case 3:
128                 printPartBrand();
129                 break;
130             default:
131                 break;
132         }
133         new CallBack();
134     }
135
136     public static void printPartAll() {
137         String csvFilePath = "Database/PartsData.csv";
138
139         try (BufferedReader br = new BufferedReader(new
140 FileReader(csvFilePath))) {
141             String line;
142             boolean isFirstLine = true;
143
144             while ((line = br.readLine()) != null) {
145                 if (isFirstLine) {
146                     isFirstLine = false;
147                     continue;
148                 }
149
150                 Parts part = Parts.createFromCSVLine(line);

```

```

150         printPart(part);
151     }
152 } catch (IOException e) {
153     e.printStackTrace();
154 }
155 }
156
157 private static void printPartAll(String s) {
158     String csvFilePath = "Database/PartsData.csv";
159     boolean found = false;
160
161     try (BufferedReader br = new BufferedReader(new
162 FileReader(csvFilePath))) {
163         String line;
164         boolean isFirstLine = true;
165
166         while ((line = br.readLine()) != null) {
167             if (isFirstLine) {
168                 isFirstLine = false;
169                 continue;
170             }
171
172             Parts part = Parts.createFromCSVLine(line);
173             if (part.type.equalsIgnoreCase(s) ||
174 part.brand.equalsIgnoreCase(s) || part.name.equalsIgnoreCase(s)) {
175                 printPart(part);
176                 found = true;
177             }
178         } catch (IOException e) {
179             e.printStackTrace();
180         }
181
182         if (!found) {
183             System.out.println("No parts found for the specified criteria:
184 " + s);
185         }
186     }
187
188     public void printPartType() {
189         // Use AppScanner instead of direct Scanner
190         System.out.println("Please select which type of product you want
191 to see by, inserting number(1-6)");
192         System.out.println("'→ 1 for Engine.");
193         System.out.println("'→ 2 for Wheels.");
194         System.out.println("'→ 3 for Turbo.");
195         System.out.println("'→ 4 for ECU.");
196         System.out.println("'→ 5 for Rear Wing.");
197         System.out.println("'→ 6 for Aero Kit.");
198         System.out.print("Select range: ");
199         int y = AppScanner.nextInt();
200
201         while (y < 1 || y > 6) {
202             System.out.println("Invalid range. Please insert a number
203 between 1 to 6: ");
204         }
205     }
206 }

```

```

200         System.out.print("Select range: ");
201         y = AppScanner.nextInt();
202     }
203
204     switch (y) {
205         case 1:
206             printPartAll("Engine");
207             break;
208         case 2:
209             printPartAll("Wheels");
210             break;
211         case 3:
212             printPartAll("Turbo");
213             break;
214         case 4:
215             printPartAll("ECU");
216             break;
217         case 5:
218             printPartAll("Rear Wing");
219             break;
220         case 6:
221             printPartAll("Aero Kit");
222             break;
223         default:
224             break;
225     }
226 }
227
228 private void printPartBrand() {
229     String csvFilePath = "Database/PartsData.csv";
230     Set<String> uniqueStrings =
UniqueStringGenerator.generateUniqueStrings(csvFilePath);
231
232     System.out.println("Unique Strings:");
233     int i = 1;
234     for (String str : uniqueStrings) {
235         System.out.println(i++ + " for " + str);
236     }
237
238     int y = AppScanner.nextInt();
239     while (y < 1 || y > (i - 1)) {
240         System.out.println("Invalid range. Please insert a number
between 1 to " + (i - 1) + "");
241         System.out.print("Select range: ");
242         y = AppScanner.nextInt();
243     }
244
245     i = 1;
246     for (String str : uniqueStrings) {
247         if (i == y) {
248             System.out.println("\t====You chose " + str + "====");
249             System.out.println("====The available parts from " + str
+ " are====");
250             System.out.println();
251             printPartAll(str);

```

```

252         } else {
253             // do nothing
254         }
255         i++;
256     }
257 }
258
259 public static void printPartName(String s) {
260     printPartAll(s);
261     new Callback();
262 }
263
264 private static void printPart(Parts part) {
265     PartPrinter.printPartInfo(part);
266 }
267 }
268
269
270 public class UniqueStringGenerator {
271     //checks the different brand name and act likes a dictionary that
272     //holds unique string
273     public static Set<String> generateUniqueStrings(String csvFilePath) {
274         Set<String> uniqueStrings = new HashSet<>();
275
276         try (BufferedReader br = new BufferedReader(new
277             FileReader(csvFilePath))) {
278             String line;
279             boolean isFirstLine = true;
280
281             while ((line = br.readLine()) != null) {
282                 if (isFirstLine) {
283                     isFirstLine = false;
284                     continue;
285                 }
286
287                 String[] data = line.split(",");
288                 // index 3 contains the string of interest
289                 String stringValue = data.length > 3 ? data[3].trim() :
290                 "";
291
292                 if (!stringValue.isEmpty()) {
293                     uniqueStrings.add(stringValue);
294                 }
295             } catch (IOException e) {
296                 e.printStackTrace();
297             }
298         }
299     }
300 }

```