

```

1  #include <fstream.h>
2  #include <iomanip.h>
3  #include <stdio.h>
4  #include <stdlib.h>
5  #include <conio.h>
6  #include <string.h>
7  #include <ctype.h>
8  #include "design.h"
9  char op;
10 void menu();
11 void delay() {
12     for (unsigned int i = 0; i < 5000; i++)
13         for (unsigned int j = 0; j < 9000; j++) { /*To delay output*/
14             }
15 }
16 void loading(short x, short y, short N = 7)
17 /* (x,y)-Position of Load Animation
18     Loading Running delay(N=1 for 0.4786 s delay approx.)
19 */
20 {
21     const char slash[5] = {92, 45, 47, 124};
22     for (unsigned short i = 0; i < N; i++)
23         for (short j = 0; j < 4; j++) {
24             gotoxy(x, y);
25             cout << slash[j];
26             delay();
27             delay();
28         }
29 }
30 void clear(short x, short y, short n, short l = 1)
31 /* (x,y)-margin; n - no. of characters to clear
32     l - no. of lines to clear
33 */
34 {
35     for (short i = 0; i < l; i++)
36         for (short j = 0; j < n; j++) {
37             gotoxy(x + j, y + i);
38             cout << " ";
39         }
40 }
41 void line(int p, int q, int l, char type = 'v', int type2 = 0) {
42     int i;
43     char hl, vl;
44     /* Where (p,q)-margin (coordinate of top-left vertice of box border);
45         type=v - For Vertical Line (default)
46         type=h - For Horizontal Line
47         l=length(horizontal distance) of border;
48     */
49     if (type2 == 1) {
50         hl = '*';
51         vl = '!';
52     } else {
53         hl = 45;
54         vl = 124;

```

```

55     } // hyphen(-) and or (|)
56     //(WHITE);
57     if (type == 'v')
58         for (i = 0; i < l; i++) // Vertical line of length l
59             {
60                 gotoxy(p, q + i);
61                 printf(
62                     "%c",
63                     vl); // '7C'(hyphen); 'DE'(accented i); '7C' (or line); 'B3'
64 (border);
65             }
66     else
67         for (i = 0; i < l; i++) // Horizontal Line of length l
68             {
69                 gotoxy(p + i, q);
70                 printf("%c", hl);
71             }
72 }
73 void border(short p, short q, short l, short w,
74             short type = 0) // Use to make borders as the name suggests
75 { /* Where (p,q)-margin (coordinate of top-left vertice of box border);
76     w-width(vertical distance) of border;
77     l-length(horizontal distance) of border;
78     type => 2(double border); type => 0,1(single border); type =>
79     3(slowmotion)
80     */
81     char cc, hl, vl;
82     if (type == 2) {
83         cc = '/';
84         hl = '/';
85         vl = '/';
86     } else if (type == 1) {
87         cc = '*';
88         hl = '+';
89         vl = '!';
90     } else if (type == 4 || type == 0) {
91         cc = '+';
92         hl = '*';
93         vl = '!';
94     } else {
95         cc = '*';
96         hl = '*';
97         vl = '*';
98     }
99     gotoxy(p, q);
100    printf("%c", cc); // top-left
101    gotoxy(p + l, q);
102    printf("%c", cc); // top-right
103    gotoxy(p, q + w);
104    printf("%c", cc); // bottom-left
105    gotoxy(p + l, q + w);
106    printf("%c", cc); // bottom-right
107    for (int i = 1, j = 1, m = (w - 1), n = (l - 1); i++, j++, m--, n--) {
108        if (i < w) {

```

```

109     gotoxy(p, q + i);
110     if (type == 2) cout << vl;
111     printf("%c", vl);
112 }
113 if (j < l) {
114     gotoxy(p + i, q);
115     printf("%c", hl);
116     if (type == 2) cout << hl;
117 }
118 if (type == 4) {
119     delay();
120 } // for delay
121 if (m > 0) {
122     gotoxy(p + l, q + m);
123     if (type == 2) cout << '\b' << vl;
124     printf("%c", vl);
125 }
126 if (n > 0) {
127     gotoxy(p + n, q + w);
128     printf("%c", hl);
129     if (type == 2) cout << hl;
130 }
131 if (type == 4) {
132     delay();
133 } // for delay
134 if (j >= l && i >= w) break;
135 }
136 }
137 void cursor() // To get position of pointer in console window(for output
138               // purpose)
139 {
140     int x = 1, y = 1;
141     char key;
142     do {
143         gotoxy(x, y);
144         cout << " ";
145         gotoxy(x, y);
146         key = getch();
147         if (key == ' ')
148             x++;
149         else if (key == '\b')
150             x--;
151         else if (key == '\r')
152             y++;
153         else if (key == 'p') {
154             gotoxy(1, 1);
155             cout << x << " " << y;
156         } else if (key == 'r') {
157             x = 1;
158             y = 1;
159         }
160     } while (key != 'N');
161 }
162 void ventry(char t[], short code = 4, short N = 1)

```

```

163  /* code=0 >> for inputing alphabets(a-Z), space, comma(,) & dot(.)
164      code=1 >> for inputing numbers(0-9), dot(.) & '/'
165      code=2 >> for inputing username / ID
166      code=3 >> for normal input without space, ':' & ';'
167      code=4 >> for menu input (only numbers 0-9)
168  */
169  {
170      unsigned short i = 0;
171      if (code == 0) {
172          while (i < N && (t[i] = getch()) != '\r')
173              if ((t[i] >= 97 && t[i] <= 122) || (t[i] >= 65 && t[i] <= 90) ||
174                  t[i] == 32 || t[i] == '.' || t[i] == ',') {
175                  if (t[0] != ' ' &&
176                      t[0] != '_') // To prevent space or '_' at first input
177                  {
178                      cout << t[i];
179                      i++;
180                  }
181              } else if (t[i] == 8 && i > 0) {
182                  cout << "\b \b";
183                  i--;
184              }
185      } else if (code == 1) {
186          while (i < N && (t[i] = getch()) != '\r')
187              if ((t[i] >= 48 && t[i] <= 57) || t[i] == 46 || t[i] == '/') {
188                  cout << t[i];
189                  i++;
190              } else if (t[i] == 8 && i > 0) {
191                  cout << "\b \b";
192                  i--;
193              }
194      }
195      if (code == 2) {
196          while (i < N && (t[i] = getch()) != '\r')
197              if ((t[i] >= 95 && t[i] <= 122) || (t[i] >= 46 && t[i] <= 57) ||
198                  (t[i] >= 65 && t[i] <= 90) || t[i] == '@') {
199                  cout << t[i];
200                  i++;
201              } else if (t[i] == 8 && i > 0) {
202                  cout << "\b \b";
203                  i--;
204              }
205      }
206      if (code == 3) {
207          while (i < N && (t[i] = getch()) != '\r')
208              if ((t[i] >= 33 && t[i] <= 126) && t[i] != ':' && t[i] != ';') {
209                  cout << t[i];
210                  i++;
211              } else if (t[i] == 8 && i > 0) {
212                  cout << "\b \b";
213                  i--;
214              }
215      } else if (code == 4) {
216          while (i < N && (t[i] = getch()) != '\r')

```

```

217         if (t[i] >= 48 &&
218             t[i] <= 57) { // cout<<t[i]; //Uncomment to display input no.
219             i++;
220         } else if (t[i] == 8 && i > 0) {
221             cout << "\b \b";
222             i--;
223         }
224     }
225     t[i] = '\0';
226 }
227 void voutput(short N, char t[], char o[] = NULL)
228 /* N >> for outputing N characters of the concatenated string
229 */
230 {
231     unsigned short i = 0;
232     strcat(t, " ");
233     strcat(t, o);
234     if (N < strlen(t)) {
235         for (i = N; i < N + 3; i++) t[i] = '.';
236         strcat(t, "..");
237         N += 2;
238     }
239     for (i = 0; i < N && i < strlen(t); i++) cout << t[i];
240 }
241 void About() {
242     border(3, 2, 74, 22, 2);
243     gotoxy(17, 4);
244     cout << "A B O U T   I N F I N I T Y   C A R   R E N T A L";
245     gotoxy(17, 5);
246     cout << "-----";
247     gotoxy(11, 6);
248     cout << "Our company started in 2001 in a small town in Joerdhailun";
249     gotoxy(14, 8);
250     cout << "which has now expanded globally having locations in";
251     gotoxy(12, 10);
252     cout << "Asia, Europe, North America. We have received numerous";
253     gotoxy(12, 12);
254     cout << "accolades for our service and the quality of our cars.";
255     gotoxy(15, 14);
256     cout << "We strive to provide our clients with quality our";
257     gotoxy(17, 16);
258     cout << "clients with quality cars at affordable prices.";
259     gotoxy(13, 18);
260     cout << "We have a wide selection of cars including high range";
261     gotoxy(18, 20);
262     cout << "of luxury cars like Rolls Royce, Bentley";
263     gotoxy(22, 22);
264     cout << "and even a few exotic sports cars. ";
265 }
266 class Login {
267     char usr[13];
268     char pss[13];
269
270     public:

```

```

271     int check();
272 } L;
273 int Login::check() {
274     xx:
275     clrscr();
276     border(3, 2, 74, 21, 2);
277     int i = 0;
278     gotoxy(23, 4);
279     printf("W E L C O M E   T O   .   .   .");
280     gotoxy(23, 5);
281     cout << "-----";
282     gotoxy(23, 6);
283     cout << "I N F I N I T Y   C A R   R E N T A L";
284     gotoxy(23, 7);
285     printf("-----");
286     gotoxy(34, 10);
287     printf("SYSTEM LOGIN");
288     gotoxy(34, 11);
289     printf("-----");
290     border(25, 12, 29, 7, 1);
291     gotoxy(28, 14);
292     cout << "Username : ";
293     gotoxy(28, 17);
294     cout << "Password : ";
295     gotoxy(39, 14);
296     ventry(usr, 2, 13);
297     gotoxy(39, 17);
298     while (i < 12 && (pss[i] = getch()) != '\r') {
299         if (pss[i] == '\b' && i > 0) {
300             cout << '\b' << ' ' << '\b';
301             i--;
302         } else if (i != 0 || pss[i] != '\b') {
303             cout << "*";
304             i++;
305         }
306     }
307     pss[i] = '\0';
308     if (strcmp(usr, "admin") == 0 && strcmp(pss, "pass123") == 0) {
309         cout << '\a';
310         clrscr();
311         return (1);
312     } else {
313         gotoxy(24, 21);
314         printf("Invalid Username Or Password !!!");
315         getch();
316         goto xx;
317     }
318 }
319 short namecmp(char *name, char *word)
320 // To search for a 'word' in a 'name'
321 {
322     if (strlen(name) < strlen(word)) {
323         clrscr();
324         cout << "Error exit function! String size less than word size.... ";

```

```

325     return (-1);
326 }
327 short sc, count = 0;
328 for (short i = 0; i <= (strlen(name) - strlen(word)); i++) {
329     sc = 0;
330     for (short j = 0; j < strlen(word); j++)
331         if (tolower(name[i + j]) != tolower(word[j])) {
332             sc = 1;
333             break;
334         }
335     if (sc == 0) count++;
336 }
337 return count;
338 }
339 //*****##### class Cars
340 //*****#####
341 //## stored in cardata
342 class Cars {
343     char carid[6];
344     char make[20];
345     char model[20];
346     char year[5];
347     char category[20];
348     char passengers;
349     float price;
350     short status; // 1=avail. , 0=booked
351 public:
352     void getcar();
353     void putcar();
354     void makecarid();
355     void CarTable();
356     void AvailTable(char *);
357     void switch_status(int op = 2);
358     char *retcarid() { return (carid); }
359     char *retmodel() { return (model); }
360     char *retmake() { return (make); }
361     float retprice() { return (price); }
362     ~Cars() { /*Destructor to distroy this class when the scope is over*/
363     }
364 } C;
365 void Cars::getcar() {
366     cout << "\n\tMAKE => ";
367     gets(make);
368     cout << "\n\tModel => ";
369     gets(model);
370     cout << "\n\tYEAR => ";
371     ventry(year, 1, 5);
372     cout << "\n\tCATEGORY => ";
373     ventry(category, 0, 16);
374     cout << "\n\tPassengers => ";
375     cin >> passengers;
376     cout << "\n\tPrice per day of car => ";
377     cin >> price;
378     status = 1; // status=1 means available;

```

```

379     makecarid();
380     cout << "\n\tCarID => " << carid;
381     getch();
382 }
383 void Cars::putcar() {
384     cout << "\n\tCarID : " << carid; // Displayed on Table
385     cout << "\n\n\tMAKE : " << make; // MAKE+model Displayed on table
386     cout << "\t\tModel : " << model;
387     cout << "\n\n\tYEAR : " << year;
388     cout << "\t\tCATEGORY : " << category;
389     cout << "\n\n\tPassengers : " << passengers;
390 }
391 void Cars::makecarid() {
392     short lm = strlen(make);
393     short ly = strlen(year);
394     int pos = 0, i, j;
395     for (j = 0; j < 20; j++) i = random(4);
396     if (i == 0) {
397         carid[pos++] = tolower(make[0]);
398         carid[pos++] = tolower(make[1]);
399     }
400     if (i == 1) {
401         carid[pos++] = make[0];
402         carid[pos++] = make[lm - 1];
403     }
404     if (i == 2) {
405         carid[pos++] = make[0];
406         carid[pos++] = make[1];
407         carid[pos++] = make[2];
408     }
409     if (i == 3) {
410         carid[pos++] = year[ly - 2];
411         carid[pos++] = year[ly - 1];
412         carid[pos++] = model[0];
413         carid[pos++] = model[2];
414     }
415     randomize();
416     for (j = 0; j < 20; j++) i = random(4);
417     if (i == 0) {
418         carid[pos++] = year[ly - 2];
419         carid[pos++] = year[ly - 1];
420     }
421     if (i == 1) {
422         carid[pos++] = model[0];
423         carid[pos++] = model[1];
424     }
425     if (i == 2 && pos < 3) {
426         carid[pos++] = model[0];
427         carid[pos++] = model[1];
428         carid[pos++] = model[2];
429     }
430     if (i == 3) {
431         carid[pos++] = tolower(make[0]);
432         carid[pos++] = '#';

```



```

433     }
434     for (i = 0; i < 5; i++)
435         if (carid[i] == NULL || carid[i] == ' ') carid[i] = passengers;
436     carid[5] = '\0';
437 }
438 void Cars::CarTable() {
439     border(4, 3, 73, 20);
440     line(8, 4, 19); // Vertical line After No.
441     line(16, 4, 19); // Vertical line After Carid
442     line(37, 4, 19); // Vertical line After Car
443     line(55, 4, 19); // After Category
444     line(65, 4, 19); // After Price
445     line(5, 5, 72, 'h'); // Horizontal line
446     gotoxy(5, 4);
447     cout << " No";
448     gotoxy(9, 4);
449     cout << " CarID";
450     gotoxy(25, 4);
451     cout << "CAR";
452     gotoxy(43, 4);
453     cout << "CATEGORY";
454     gotoxy(56, 4);
455     cout << "Price/day";
456     gotoxy(69, 4);
457     cout << "STATUS";
458     gotoxy(1, 1);
459     int i = 0, r = 7;
460     ifstream carfile("carsdata", ios::in | ios::binary);
461     gotoxy(20, 2);
462     cout << "A L L   C A R S   R E P O R T   T A B L E";
463     // Read From File
464     while (carfile.read((char *)&C, sizeof(Cars))) {
465         gotoxy(6, r);
466         (i > 8) ? cout << (i + 1) : cout << " " << (i + 1);
467         gotoxy(10, r);
468         cout << carid;
469         gotoxy(18, r);
470         voutput(17, make, model);
471         gotoxy(39, r);
472         cout << category;
473         gotoxy(59, r);
474         cout << C.price;
475         gotoxy(67, r);
476         (status == 1) ? cout << "Available" : cout << "BOOKED!";
477         i++;
478         r++;
479     }
480 }
481 void Cars::AvailTable(char *rentcarid) {
482     clrscr();
483     border(4, 4, 73, 17);
484     line(9, 5, 16, 'v'); // Vertical line After No.
485     line(16, 5, 16, 'v'); // Vertical line After Carid
486     line(39, 5, 16, 'v'); // Vertical line After Car

```

```

487 line(58, 5, 16, 'v'); // After Category
488 line(70, 5, 16, 'v'); // After Price
489 line(5, 6, 72, 'h'); // Horizontal line
490 gotoxy(5, 5);
491 cout << " No";
492 gotoxy(10, 5);
493 cout << " Year";
494 gotoxy(26, 5);
495 cout << "CAR";
496 gotoxy(45, 5);
497 cout << "CATEGORY";
498 gotoxy(60, 5);
499 cout << "Price/day";
500 gotoxy(71, 5);
501 cout << "Seats";
502 gotoxy(1, 2);
503 ifstream carfile("carsdata", ios::in | ios::binary);
504 // Read From File
505 int i, r;
506 cout << "\t\t\tDISPLAY AVAILABLE CARS";
507 for (i = 1, r = 8; carfile.read((char *)&C, sizeof(C));)
508     if (status == 1) {
509         gotoxy(5, r);
510         cout << "[" << i << "]";
511         gotoxy(11, r);
512         cout << year;
513         gotoxy(18, r);
514         voutput(19, make, model);
515         gotoxy(41, r);
516         cout << category;
517         gotoxy(63, r);
518         cout << C.price;
519         gotoxy(73, r);
520         cout << passengers;
521         i++, r++;
522     }
523 carfile.clear();
524 carfile.seekg(0);
525 gotoxy(5, 24);
526 cout << "Select Car No. >> ";
527 cin >> r;
528 i = 1;
529 while (carfile.read((char *)&C, sizeof(C))) {
530     if (i == r && status == 1) {
531         strcpy(rentcarid, carid);
532         break;
533     }
534     if (status == 1) {
535         i++;
536     }
537 }
538 }
539 void Cars::switch_status(int op) {
540     if (op == 1)

```

```

541     status = 1;
542     else if (op == 0)
543         status = 0;
544     else
545         (status == 0) ? status = 1 : status = 0;
546 }
547 void status_edit(char *op = "#Nothing") {
548     fstream carfile("carsdata", ios::in | ios::out | ios::binary);
549     while (carfile.read((char *)&C, sizeof(C))) {
550         if (*op == '1' && *(op + 1) == '\\0')
551             C.switch_status(1);
552         else if (*op == '0' && *(op + 1) == '\\0')
553             C.switch_status(0);
554         else if (strcmp(C.retcarid(), op) == 0) {
555             C.switch_status();
556             carfile.seekg(carfile.tellg() - sizeof(C));
557             carfile.write((char *)&C, sizeof(C));
558             break;
559         }
560         carfile.seekg(carfile.tellg() - sizeof(C));
561         carfile.write((char *)&C, sizeof(C));
562     }
563 }
564 void AddCar() {
565     Cars C;
566     int i, N;
567     cout << "\\nEnter no. of cars to be added => ";
568     cin >> N;
569     clrscr();
570     ofstream carfile("carsdata", ios::out | ios::binary | ios::app);
571     // Write TO File
572     cout << "\\n\\t\\t\\tADD NEW CAR";
573     for (i = 0; i < N; i++) {
574         clrscr();
575         cout << "\\n\\n\\tAdd Car No. " << (i + 1) << endl;
576         C.getcar();
577         carfile.write((char *)&C, sizeof(Cars));
578     }
579     carfile.close();
580 }
581 short SearchCar(Cars &D, char *objective = NULL)
582 /* Can only search for make, model, carid and price
583 */
584 {
585     short count = 0, searchstatus = 0, i = 0;
586     Cars M[10];
587     ifstream carfile("carsdata", ios::in | ios::out | ios::binary);
588     if (*objective == NULL) return 0;
589     // Read from File and Write to File
590     while (carfile.read((char *)&C, sizeof(Cars))) {
591         if (strcmp(objective, C.retcarid()) == 0 ||
592             strcmpi(objective, C.retmodel()) == 0) {
593             D = C;
594             searchstatus++;

```

```

595     break;
596 } else if (strcmpi(objective, C.retmake()) == 0)
597     count++;
598 }
599 carfile.clear();
600 carfile.seekg(0);
601 if (count == 1)
602     while (carfile.read((char *)&C, sizeof(Cars)))
603         if (strcmpi(objective, C.retmake()) == 0) {
604             D = C;
605             searchstatus++;
606             break;
607         }
608 if (count > 1) { // clrscr();
609     gotoxy(13, 13);
610     cout << "More than one record found !\a";
611     while (carfile.read((char *)&C, sizeof(Cars)))
612         if (strcmpi(objective, C.retmake()) == 0) {
613             M[i] = C;
614             i++;
615         }
616     for (i = 0; i < count; i++) {
617         gotoxy(9, 15 + i);
618         cout << "[" << (i + 1) << "]"
619             << "\tCarID: " << M[i].retcarid();
620         gotoxy(36, 15 + i);
621         cout << "CAR: " << M[i].retmake() << " " << M[i].retmodel();
622     }
623     int op;
624     gotoxy(6, 18);
625     cout << " => ";
626     cin >> op;
627     carfile.clear();
628     D = M[op - 1];
629     searchstatus++;
630 }
631 carfile.close();
632 if (searchstatus)
633     return 1;
634 else
635     return 0;
636 }
637 void UpdateCar() {
638     short updatestatus = 0;
639     Cars D;
640     char update[20];
641     fstream carfile("carsdata", ios::in | ios::out | ios::binary);
642     // Read from File and Write to File
643     clrscr();
644     cout << "\nEnter Carid/Model/Make of Car to Update => ";
645     gets(update);
646     SearchCar(D, update);
647     D.putcar();
648     getch();

```

```

649 while (carfile.read((char *)&C, sizeof(Cars)))
650     if (strcmp(update, C.retcarid()) == 0 ||
651         strcmpi(update, C.retmodel()) == 0 ||
652         strcmp(D.retcarid(), C.retcarid()) == 0) {
653         cout << "\nCar Found!";
654         updatestatus = 1;
655         carfile.seekg(carfile.tellg() - sizeof(Cars));
656         C.putcar();
657         C.getcar();
658         carfile.write((char *)&C, sizeof(Cars));
659         break;
660     }
661     if (updatestatus)
662         cout << "\n\t\tCar Details Updated ! ";
663     else
664         cout << "\n\t\tCar Not Found !";
665     carfile.close();
666     getch();
667 }
668 void SearchCar_UI() {
669     char info[30];
670     search:
671     clrscr();
672     border(3, 2, 74, 21, 2);
673     gotoxy(30, 4);
674     cout << "SEARCH AND DISPLAY CAR";
675     gotoxy(30, 5);
676     cout << "-----";
677     gotoxy(8, 7);
678     cout << "Carid: ";
679     gotoxy(8, 9);
680     cout << "Model: ";
681     gotoxy(8, 11);
682     cout << "MAKE: ";
683     gotoxy(36, 7);
684     cout << "# Press ENTER to skip";
685     gotoxy(36, 8);
686     cout << "# Atleast one field should be entered";
687     gotoxy(36, 9);
688     cout << "# Model=Car Model(eg. Corolla,Innova)";
689     gotoxy(36, 10);
690     cout << "# MAKE=Company(eg. BMW,TOYOTA)";
691     line(6, 12, 45, 'h');
692     gotoxy(15, 7);
693     gets(info);
694     if (info[0] == NULL || SearchCar(C, info) == 0) {
695         gotoxy(15, 9);
696         gets(info);
697         if (info[0] == NULL || SearchCar(C, info) == 0) {
698             gotoxy(14, 11);
699             gets(info);
700             if (SearchCar(C, info) == 0) {
701                 gotoxy(8, 13);
702                 cout << "RECORD NOT FOUND !";

```

```

703         gotoxy(8, 15);
704         cout << "Do you wish to continue (y/n)....";
705         if (tolower(::op = getch()) == 'n') {
706             flushall();
707             return;
708         }
709         goto search;
710     }
711 }
712 }
713 flushall();
714 clear(7, 13, 65, 7);
715 gotoxy(13, 12);
716 C.putcar();
717 cout << "\n\n\t\tDo you wish to search again ? (y/n).....";
718 border(3, 2, 74, 21, 2);
719 choose:
720     if (tolower(::op = getch()) == 'y')
721         goto search;
722     else if (::op == 'n') {
723         flushall();
724         return;
725     } else
726         goto choose;
727 }
728 void DeleteCar() {
729     char info[30];
730     search:
731     clrscr();
732     border(3, 2, 74, 21, 2);
733     gotoxy(35, 4);
734     cout << "DELETE CAR";
735     gotoxy(35, 5);
736     cout << "-----";
737     gotoxy(8, 7);
738     cout << "Carid: ";
739     gotoxy(8, 9);
740     cout << "Model: ";
741     gotoxy(8, 11);
742     cout << "MAKE: ";
743     gotoxy(36, 7);
744     cout << "# Press ENTER to skip";
745     gotoxy(36, 8);
746     cout << "# Atleast one field should be entered";
747     gotoxy(36, 9);
748     cout << "# Model=Car Model(eg. Corolla,Innova)";
749     gotoxy(36, 10);
750     cout << "# MAKE=Company(eg. BMW,TOYOTA)";
751     line(6, 12, 45, 'h');
752     gotoxy(15, 7);
753     gets(info);
754     if (info[0] == NULL || SearchCar(C, info) == 0) {
755         gotoxy(15, 9);
756         gets(info);

```

```

757     if (info[0] == NULL || SearchCar(C, info) == 0) {
758         gotoxy(14, 11);
759         gets(info);
760         if (SearchCar(C, info) == 0) {
761             gotoxy(8, 13);
762             cout << "RECORD NOT FOUND !";
763             gotoxy(8, 15);
764             cout << "Do you wish to continue (y/n)....";
765             if (tolower(::op = getch()) == 'n') {
766                 flushall();
767                 return;
768             }
769             goto search;
770         }
771     }
772 }
773 flushall();
774 clear(7, 13, 65, 7);
775 gotoxy(13, 12);
776 C.putcar();
777 border(3, 2, 74, 21, 2);
778 gotoxy(17, 21);
779 cout << '_';
780 gotoxy(17, 22);
781 cout << "Delete";
782 gotoxy(38, 21);
783 cout << '_';
784 gotoxy(38, 22);
785 cout << "Retry";
786 gotoxy(58, 21);
787 cout << '_';
788 gotoxy(58, 22);
789 cout << "Cancel";
790 choose:
791 if (tolower(::op = getch()) == 'r')
792     goto search;
793 else if (::op == 'c') {
794     flushall();
795     return;
796 } else if (::op == 'd') {
797     del:
798     clrscr();
799     border(3, 2, 74, 21, 2);
800     gotoxy(8, 4);
801     cout << "\aAre you sure you want to DELETE RECORD !! (Y/N): ";
802     ventry(&::op, 3, 2);
803     if (::op == 'Y') {
804         Cars D;
805         ifstream carfile;
806         ofstream temp;
807         carfile.open("carsdata", ios::binary);
808         temp.open("temp", ios::binary);
809         while (carfile.read((char *)&D, sizeof(Cars)))
810             if (strcmp(C.retcarid(), D.retcarid()) != 0)

```

```

811     temp.write((char *)&D, sizeof(D));
812     temp.close();
813     carfile.close();
814     flushall();
815     remove("carsdata");
816     rename("temp", "carsdata");
817     gotoxy(8, 6);
818     cout << "RECORD Successfully DELETED....";
819     getch();
820 } else if (::op == 'N')
821     return;
822 else
823     goto del;
824 } else
825     goto choose;
826 }
827 void DisplayCar() {
828     Cars D;
829     int i = 0;
830     ifstream carfile("carsdata", ios::in | ios::binary);
831     // Read From File
832     clrscr();
833     cout << "\n\t\t\tDISPLAY AVAILABLE CARS";
834     while (carfile.read((char *)&D, sizeof(Cars))) {
835         cout << "\n\n\tCar No. " << (i + 1) << endl;
836         D.putcar();
837         getch();
838         i++;
839     }
840     carfile.close();
841 }
842 void rentcar_design() {
843     border(3, 2, 74, 21, 2);
844     gotoxy(9, 7);
845     cout << "Drv. Lic. #: ";
846     gotoxy(46, 7);
847     cout << "Name: ";
848     line(8, 8, 60, 'h');
849     gotoxy(9, 10);
850     cout << "PhoneNo. & or Address: ";
851     gotoxy(9, 12);
852     cout << "City: ";
853     gotoxy(46, 12);
854     cout << "ZIPCode: ";
855     gotoxy(9, 14);
856     cout << "Email: ";
857     gotoxy(46, 14);
858     cout << "Rent Car ID: ";
859     line(8, 16, 60, 'h');
860     gotoxy(9, 17);
861     cout << "FROM DATE (dd/mm/yyyy): ";
862     gotoxy(45, 17);
863     cout << "TO DATE (dd/mm/yyyy): ";
864 }

```



```

865 //#####***** CLASS rentDate *****#####
866 // TO Calculate difference in dates
867 struct date {
868     unsigned short day, month, year;
869 };
870 void getdate(date &, short, short);
871 void putdate(date &, short, short);
872 class rentDate {
873     public:
874     date from, to;
875     unsigned int convert_datetodays(date D, unsigned short y = 0);
876     unsigned int calc_diff() {
877         return (convert_datetodays(to, from.year) - convert_datetodays(from));
878     }
879     short leap_year(unsigned short year) {
880         if ((year % 4 == 0) && !(year % 100 == 0) || year % 400 == 0)
881             return 1;
882         else
883             return 0;
884     }
885     short verifydate(short, short code = 1);
886     void Get_fromdate(short p, short q) { getdate(from, p, q); }
887     void Get_todate(short p, short q) { getdate(to, p, q); }
888     void Put_fromdate(short p, short q) { putdate(from, p, q); }
889     void Put_todate(short p, short q) { putdate(to, p, q); }
890     ~rentDate() { /*Destructor, the destroyer*/
891     }
892 } A;
893 void getdate(date &d, short p, short q) {
894     in:
895     clear(p, q, 12);
896     gotoxy(p, q);
897     cout << " / / ";
898     gotoxy(p, q);
899     cin >> d.day;
900     if (!A.verifydate(d.day, 1)) {
901         cin.clear();
902         cin.ignore(1000, '\n');
903         goto in;
904     }
905     if (d.day < 10) {
906         gotoxy(p, q);
907         cout << 0 << d.day;
908     }
909     gotoxy(p + 3, q);
910     cin >> d.month;
911     if (!A.verifydate(d.month, 2)) {
912         cin.clear();
913         cin.ignore(100, '\n');
914         goto in;
915     }
916     if (d.month < 10) {
917         gotoxy(p + 3, q);
918         cout << 0 << d.month;

```

```

919     }
920     gotoxy(p + 6, q);
921     cin >> d.year;
922     if (!A.verifydate(d.year, 3)) {
923         cin.clear();
924         cin.ignore(100, '\n');
925         goto in;
926     }
927 }
928 void putdate(date &d, short p, short q) {
929     gotoxy(p, q);
930     cout << setw(2) << setfill('0') << d.day << '/';
931     cout << setw(2) << setfill('0') << d.month << '/' << d.year;
932 }
933 short rentDate::verifydate(short d, short code) {
934     if (code == 2) {
935         if (d >= 1 && d <= 12)
936             return 1;
937         else
938             return 0;
939     } else if (code == 3) {
940         if (d >= 2016 && d <= 2070)
941             return 1;
942         else
943             return 0;
944     } else {
945         if (d >= 1 && d <= 31)
946             return 1;
947         else
948             return 0;
949     }
950 }
951 unsigned int rentDate::convert_datetodays(date D, unsigned short y) {
952     unsigned int days = 0;
953     short i;
954     for (unsigned short k = y; k < D.year; k++) {
955         if (y == 0) break;
956         if (leap_year(k))
957             days += 366;
958         else
959             days += 365;
960     }
961     for (i = 1; i < D.month; i++) {
962         if (i == 4 || i == 6 || i == 9 || i == 11)
963             days += 30;
964         else if (i == 2) {
965             if (leap_year(D.year))
966                 days += 29;
967             else
968                 days += 28;
969         } else
970             days += 31;
971     }
972     days += D.day;

```

```

973     return days;
974 }
975 //*****//*****//***** class Customer
976 //*****//*****//*****//*****
977 /* stored in CUST_record */
978 class Customer {
979     char lic_no[12];
980     char name[30], email[30];
981     char address[100], city[30];
982     char rentcarid[10];
983     char zipcode[10];
984
985 public:
986     rentDate T;
987     void getcustomer();
988     void putcustomer();
989     void CustomerCar_table();
990     void get_rentcarid() { C.AvailTable(rentcarid); }
991     void customer_report();
992     char *ret_rentcarid() { return rentcarid; }
993     char *ret_licno() { return lic_no; }
994     char *ret_name() { return name; }
995     char *ret_email() { return email; }
996     char *ret_address() { return address; }
997     ~Customer() { /*Destructor the destroyer*/
998     }
999 } N;
1000 void Customer::getcustomer() {
1001     lic:
1002     gotoxy(22, 7);
1003     ventry(lic_no, 1, 10);
1004     if (lic_no[0] == NULL) goto lic;
1005     name:
1006     gotoxy(52, 7);
1007     ventry(name, 0, 23);
1008     if (name[0] == NULL) goto name;
1009     gotoxy(33, 10);
1010     gets(address);
1011     gotoxy(15, 12);
1012     gets(city);
1013     gotoxy(55, 12);
1014     ventry(zipcode, 1, 6);
1015     gotoxy(16, 14);
1016     ventry(email, 3, 26);
1017     date:
1018     T.Get_fromdate(13, 18);
1019     T.Get_todate(49, 18);
1020     if (T.calc_diff() > 365) {
1021         gotoxy(13, 20);
1022         cout << "Sorry, maximum rent period is 365 days !.....";
1023         getch();
1024         clear(13, 18, 60, 4);
1025         goto date;
1026     }

```

```

1027     flushall();
1028 }
1029 void Customer::putcustomer() {
1030     gotoxy(22, 7);
1031     cout << lic_no;
1032     gotoxy(52, 7);
1033     cout << name;
1034     gotoxy(33, 10);
1035     cout << address;
1036     gotoxy(15, 12);
1037     cout << city;
1038     gotoxy(55, 12);
1039     puts(zipcode);
1040     gotoxy(16, 14);
1041     puts(email);
1042     gotoxy(59, 14);
1043     puts(rentcarid);
1044     T.Put_fromdate(13, 18);
1045     T.Put_todate(50, 18);
1046 }
1047 void Customer::CustomerCar_table() {
1048     clrscr();
1049     gotoxy(20, 2);
1050     cout << "C U S T O M E R - C A R   R E P O R T   T A B L E";
1051     ifstream customer("CUST_record", ios::in | ios::binary);
1052     if (!customer) {
1053         cout << "\n\n\t\aFile Does Not Exist / Unable to Open File ";
1054         getch();
1055     } else {
1056         Cars D;
1057         border(4, 3, 73, 20, 1);
1058         line(16, 4, 19); // Vertical line After Lic. #
1059         line(38, 4, 19); // Vertical line After Customer
1060         line(46, 4, 19); // After CarID
1061         line(67, 4, 19); // After CAR
1062         line(5, 5, 72, 'h'); // Horizontal line
1063         gotoxy(5, 4);
1064         cout << " Licence #";
1065         gotoxy(23, 4);
1066         cout << " CUSTOMER ";
1067         gotoxy(39, 4);
1068         cout << " CarID";
1069         gotoxy(54, 4);
1070         cout << " CAR";
1071         gotoxy(68, 4);
1072         cout << "Price/day";
1073         short i = 0, r = 7;
1074         // Read From File
1075         while (customer.read((char *) &N, sizeof(N))) {
1076             SearchCar(D, rentcarid);
1077             gotoxy(40, r);
1078             cout << rentcarid;
1079             gotoxy(6, r);
1080             cout << lic_no;

```

```

1081     gotoxy(48, r);
1082     voutput(17, D.retmake(), D.retmodel());
1083     gotoxy(18, r);
1084     cout << name;
1085     gotoxy(71, r);
1086     cout << D.retprice();
1087     // getch();
1088     i++;
1089     r += 2;
1090 }
1091 }
1092 }
1093 void Customer::customer_report() {
1094     clrscr();
1095     gotoxy(22, 2);
1096     cout << "A L L   C U S T O M E R   R E P O R T";
1097     ifstream customer("CUST_record", ios::in | ios::binary);
1098     if (!customer) {
1099         gotoxy(8, 4);
1100         cout << "\aFile Does Not Exist / Unable to Open File ";
1101     } else {
1102         Cars D;
1103         border(4, 3, 73, 20, 1);
1104         gotoxy(2, 24);
1105         cout << "* Price may vary depending on the rented car and returning day";
1106         gotoxy(4, 25);
1107         cout << "and offers included";
1108         line(16, 4, 19); // Vertical line After Lic. #
1109         line(38, 4, 19); // Vertical line After Customer
1110         line(49, 4, 19); // After fromdate
1111         line(60, 4, 19); // After todate
1112         line(67, 4, 19); // After days
1113         line(5, 5, 72, 'h'); // Horizontal line
1114         gotoxy(5, 4);
1115         cout << " Licence #";
1116         gotoxy(23, 4);
1117         cout << " CUSTOMER ";
1118         gotoxy(40, 4);
1119         cout << "From Date";
1120         gotoxy(51, 4);
1121         cout << "To Date";
1122         gotoxy(61, 4);
1123         cout << " Days";
1124         gotoxy(69, 4);
1125         cout << "*Price";
1126         short i = 0, r = 7;
1127         // Read From File
1128         while (customer.read((char *)this, sizeof(Customer))) {
1129             SearchCar(D, rentcarid);
1130             gotoxy(6, r);
1131             cout << lic_no;
1132             gotoxy(18, r);
1133             cout << name;
1134             T.Put_fromdate(39, r);

```

```

1135     T.Put_todate(50, r);
1136     gotoxy(62, r);
1137     cout << T.calc_diff();
1138     gotoxy(69, r);
1139     cout << (T.calc_diff() * D.retprice());
1140     i++;
1141     r += 2;    // getch();
1142 }
1143 }
1144 }
1145 void AddCustomer() {
1146     clrscr();
1147     gotoxy(32, 4);
1148     cout << "RENT A CAR SYSTEM";
1149     gotoxy(32, 5);
1150     cout << "-----";
1151     rentcar_design();
1152     N.getcustomer();
1153     N.get_rentcarid();
1154     clrscr();
1155     rentcar_design();
1156     gotoxy(32, 4);
1157     cout << "RENT A CAR SYSTEM";
1158     gotoxy(32, 5);
1159     cout << "-----";
1160     N.putcustomer();
1161     gotoxy(23, 21);
1162     cout << '_';
1163     gotoxy(23, 22);
1164     cout << "Save";
1165     gotoxy(51, 21);
1166     cout << '_';
1167     gotoxy(51, 22);
1168     cout << "Cancel";
1169     save:
1170     ::op = tolower(getch());
1171     if (::op == 's') {    // Write TO File
1172         ofstream cust;
1173         cust.open("CUST_record", ios::out | ios::binary | ios::app);
1174         cust.write((char *)&N, sizeof(class Customer));
1175         cust.close();
1176         flushall();
1177         status_edit(N.ret_rentcarid());
1178         clrscr();
1179         gotoxy(30, 12);
1180         cout << " PLEASE WAIT ! ";
1181         loading(52, 12);
1182         gotoxy(30, 12);
1183         cout << " CUSTOMER SAVED ! ";
1184         for (short i = 0; i < 5; i++) delay();
1185     } else if (::op == 'c') {
1186         flushall();
1187         return;
1188     } else

```

```

1189     goto save;
1190 }
1191 void DisplayCust() {
1192     clrscr();
1193     gotoxy(32, 4);
1194     cout << "RENT A CAR SYSTEM";
1195     gotoxy(32, 5);
1196     cout << "-----";
1197     ifstream cust;
1198     cust.open("CUST_record", ios::in | ios::binary);
1199     cust.clear();
1200     cust.seekg(0);
1201     // Read From File
1202     while (cust.read((char *)&N, sizeof(class Customer))) {
1203         clear(4, 6, 73, 18);
1204         rentcar_design();
1205         N.putcustomer();
1206         flushall();
1207         getch();
1208     }
1209     cust.close();
1210 }
1211 short SearchCust(Customer &D, char *objective = "#Nothing", short name = 0)
1212 /* Can only search for Drv_Lic.#, *Name, email_id and rentcarid
1213    * more than one record;
1214 */ {
1215     short count = 0, searchstatus = 0, i = 0;
1216     Customer M[10];
1217     ifstream customer("CUST_record", ios::in | ios::out | ios::binary);
1218     if (*objective == NULL) return 0;
1219     // Read from File and Write to File
1220     while (customer.read((char *)&N, sizeof(Customer))) {
1221         if (strcmp(objective, N.ret_rentcarid()) == 0 ||
1222             strcmp(objective, N.ret_email()) == 0 ||
1223             strcmp(objective, N.ret_licno()) == 0) {
1224             D = N;
1225             searchstatus++;
1226             break;
1227         } else if (name == 1 && strcmp(N.ret_name(), objective))
1228             count++;
1229     }
1230     // gotoxy(1,1);cout<<"{ "<<count<<" }"; //For Testing
1231     customer.clear();
1232     customer.seekg(0);
1233     if (count == 1)
1234         while (customer.read((char *)&N, sizeof(Customer)))
1235             if (strcmp(N.ret_name(), objective)) {
1236                 D = N;
1237                 searchstatus++;
1238                 break;
1239             }
1240     if (count > 1) {
1241         gotoxy(8, 16);
1242         cout << "More than one record found !\a";

```

```

1243 while (customer.read((char *)&N, sizeof(Customer)))
1244     if (strcmp(N.ret_name(), objective)) {
1245         M[i] = N;
1246         i++;
1247     }
1248 gotoxy(9, 17);
1249 cout << "Option\t\t Drv_Lic.#"
1250     << "\t\t\tFull Name";
1251 gotoxy(7, 18);
1252 cout << "-----";
1253 for (i = 0; i < count; i++) {
1254     gotoxy(10, 19 + i);
1255     cout << "[" << (i + 1) << "]"
1256         << "\t\t" << setw(10) << setfill(' ') << M[i].ret_licno();
1257     cout << "\t\t" << setw(17) << M[i].ret_name();
1258 }
1259 int op;
1260 opt:
1261 gotoxy(8, 22);
1262 cout << "Enter Option / 0 to exit => ";
1263 cin >> op;
1264 if (op < i + 2) {
1265     if (op == 0) {
1266         clear(6, 16, 65, 7);
1267         return 0;
1268     }
1269     D = M[op - 1];
1270     searchstatus++;
1271 } else
1272     goto opt;
1273 }
1274 customer.close();
1275 if (searchstatus)
1276     return 1;
1277 else
1278     return 0;
1279 }
1280 void SearchCust_UI() {
1281     char info[30];
1282 search:
1283     clrscr();
1284     border(3, 2, 74, 21, 2);
1285     gotoxy(26, 4);
1286     cout << "SEARCH AND DISPLAY CUSTOMER";
1287     gotoxy(26, 5);
1288     cout << "-----";
1289     gotoxy(8, 8);
1290     cout << "Drv. Lic. #: ";
1291     gotoxy(8, 10);
1292     cout << "Rent Car ID: ";
1293     gotoxy(8, 12);
1294     cout << "Name: ";
1295     gotoxy(8, 14);
1296     cout << "Email: ";

```



```

1297 gotoxy(37, 8);
1298 cout << "# Press ENTER to skip";
1299 gotoxy(37, 10);
1300 cout << "# Atleast one field should be entered";
1301 gotoxy(37, 12);
1302 cout << "# Drv. Lic. # = Licence No.of Customer";
1303 gotoxy(37, 14);
1304 cout << "# Rent Car ID = CarID of rented Car";
1305 line(6, 15, 45, 'h');
1306 gotoxy(21, 8);
1307 gets(info); // input lic_no.
1308 if (info[0] == NULL || SearchCust(N, info) == 0) // search lic_no.
1309 {
1310     gotoxy(21, 10);
1311     gets(info); // input rentcarid
1312     if (info[0] == NULL || SearchCust(N, info) == 0) // search rentcarid
1313     {
1314         gotoxy(14, 12);
1315         gets(info); // input name
1316         if (info[0] == NULL || SearchCust(N, info, 1) == 0) // search name
1317         {
1318             gotoxy(15, 14);
1319             gets(info); // input email
1320             if (SearchCust(N, info) == 0) // search email
1321             {
1322                 gotoxy(8, 16);
1323                 cout << "RECORD NOT FOUND !";
1324                 gotoxy(8, 18);
1325                 cout << "Do you wish to continue (y/n)....";
1326                 if (tolower(::op = getch()) == 'n') {
1327                     flushall();
1328                     return;
1329                 }
1330                 goto search;
1331             }
1332         }
1333     }
1334 }
1335 flushall();
1336 clrscr();
1337 rentcar_design();
1338 N.putcustomer();
1339 gotoxy(29, 4);
1340 cout << "RENTAL INFORMATION";
1341 gotoxy(29, 5);
1342 cout << "-----";
1343 gotoxy(20, 21);
1344 cout << "Do you wish to search again ? (y/n).....";
1345 choose:
1346 if (tolower(::op = getch()) == 'y')
1347     goto search;
1348 else if (::op == 'n') {
1349     flushall();
1350     return;

```

```

1351     } else
1352         goto choose;
1353 }
1354 void deletecust(Customer D) {
1355     ifstream customer;
1356     ofstream temp;
1357     customer.open("CUST_record", ios::binary);
1358     temp.open("temp", ios::binary);
1359     while (customer.read((char *)&N, sizeof(Customer)))
1360         if (strcmp(D.ret_licno(), N.ret_licno()) != 0)
1361             temp.write((char *)&N, sizeof(Customer));
1362     temp.close();
1363     customer.close();
1364     flushall();
1365     remove("Cust_record");
1366     rename("temp", "Cust_record");
1367 }
1368 void DeleteCust() {
1369     char info[30];
1370 search:
1371     clrscr();
1372     border(3, 2, 74, 21, 2);
1373     gotoxy(33, 4);
1374     cout << "DELETE CUSTOMER";
1375     gotoxy(33, 5);
1376     cout << "-----";
1377     gotoxy(8, 8);
1378     cout << "Drv. Lic. #: ";
1379     gotoxy(8, 12);
1380     cout << "Name: ";
1381     gotoxy(8, 14);
1382     cout << "Email: ";
1383     gotoxy(8, 10);
1384     cout << "Rent Car ID: ";
1385     gotoxy(37, 8);
1386     cout << "# Press ENTER to skip";
1387     gotoxy(37, 10);
1388     cout << "# Atleast one field should be entered";
1389     gotoxy(37, 12);
1390     cout << "# Drv. Lic. # = Licence No.of Customer";
1391     gotoxy(37, 14);
1392     cout << "# Rent Car ID = CarID of rented Car";
1393     line(6, 15, 45, 'h');
1394     gotoxy(21, 8);
1395     gets(info); // input lic_no.
1396     if (info[0] == NULL || SearchCust(N, info) == 0) // search lic_no.
1397     {
1398         gotoxy(21, 10);
1399         gets(info); // input rentcarid
1400         if (info[0] == NULL || SearchCust(N, info) == 0) // search rentcarid
1401         {
1402             gotoxy(14, 12);
1403             gets(info); // input name
1404             if (info[0] == NULL || SearchCust(N, info, 1) == 0) // search name

```

```

1405     {
1406         gotoxy(15, 14);
1407         gets(info); // input email
1408         if (SearchCust(N, info) == 0) // search email
1409         {
1410             gotoxy(8, 16);
1411             cout << "RECORD NOT FOUND !";
1412             gotoxy(8, 18);
1413             cout << "Do you wish to continue (y/n)....";
1414             if (tolower(::op = getch()) == 'n') {
1415                 flushall();
1416                 return;
1417             }
1418             goto search;
1419         }
1420     }
1421 }
1422 }
1423 flushall();
1424 clrscr();
1425 rentcar_design();
1426 N.putcustomer();
1427 gotoxy(29, 4);
1428 cout << "RENTAL INFORMATION";
1429 gotoxy(29, 5);
1430 cout << "-----";
1431 gotoxy(20, 21);
1432 gotoxy(17, 21);
1433 cout << '_';
1434 gotoxy(17, 22);
1435 cout << "Delete";
1436 gotoxy(38, 21);
1437 cout << '_';
1438 gotoxy(38, 22);
1439 cout << "Retry";
1440 gotoxy(58, 21);
1441 cout << '_';
1442 gotoxy(58, 22);
1443 cout << "Cancel";
1444 choose:
1445 if (tolower(::op = getch()) == 'r')
1446     goto search;
1447 else if (::op == 'c') {
1448     flushall();
1449     return;
1450 } else if (::op == 'd') {
1451     del:
1452     clrscr();
1453     border(3, 2, 74, 21, 2);
1454     gotoxy(8, 4);
1455     cout << "\aAre you sure you want to DELETE RECORD !! (Y/N): ";
1456     ventry(&::op, 3, 2);
1457     if (::op == 'Y') {
1458         status_edit(N.ret_rentcarid());

```

```

1459     deletecust(N);
1460     gotoxy(8, 6);
1461     cout << "RECORD Successfully DELETED....";
1462     getch();
1463     } else if (::op == 'N')
1464         return;
1465     else
1466         goto del;
1467     } else
1468         goto choose;
1469 }
1470 class Bill { // Customer details
1471     char lic_no[12];
1472     char name[30];
1473     char contact[100];
1474     // Car details
1475     char carid[10];
1476     char car[40];
1477     float carprice;
1478     rentDate D;
1479     date currentdate;
1480     float totalprice;
1481
1482     public:
1483     void makeBill(Customer);
1484     void getcar(char *);
1485     // void getdate(date&,short,short);
1486     void calc_totalprice();
1487     unsigned int calc_returndaydiff();
1488     void editdate();
1489     void test() {
1490         gotoxy(8, 1);
1491         cout << totalprice;
1492     }
1493     void editprice();
1494     void printbill();
1495     void savebill(); // Saves Bill in binaryfile"bills.dat"
1496 };
1497 void Bill::makeBill(Customer N) {
1498     border(3, 2, 74, 21, 2);
1499     gotoxy(10, 4);
1500     cout << "Enter Current Date: ";
1501     getdate(currentdate, 30, 4);
1502     strcpy(lic_no, N.ret_licno());
1503     strcpy(name, N.ret_name());
1504     strcpy(contact, N.ret_address());
1505     getcar(N.ret_rentcarid());
1506     D = N.T;
1507     calc_totalprice();
1508 }
1509 unsigned int Bill::calc_returndaydiff() {
1510     signed int days;
1511     unsigned int a = A.convert_datetodays(currentdate, 2016);
1512     unsigned int b = A.convert_datetodays(D.to, 2016);

```

```

1513     days = a - b;
1514     if (days > 2)
1515         return days;
1516     else
1517         return 0;
1518 }
1519 void Bill::editdate() {
1520     clear(32, 15, 12);
1521     getdate(D.to, 32, 15);
1522     calc_totalprice();
1523 }
1524 void Bill::editprice() {
1525     clear(10, 20, 60);
1526     gotoxy(10, 20);
1527     cout << "New Amount ( Rs " << totalprice << " )= Rs ";
1528     cin >> totalprice;
1529 }
1530 void Bill::getcar(char *rentcarid) {
1531     Cars T;
1532     SearchCar(T, rentcarid);
1533     strcpy(carid, T.retcid());
1534     strcpy(car, T.retmake());
1535     strcat(car, " ");
1536     strcat(car, T.retmodel());
1537     carprice = T.retprice();
1538 }
1539 void Bill::calc_totalprice() {
1540     totalprice = D.calc_diff() * carprice;
1541     if (calc_returndaydiff())
1542         totalprice += (carprice * 1.5 * calc_returndaydiff());
1543     if (totalprice > 100000) {
1544         // gotoxy(38,20);cout<<"You have recieved 25% discount!!"<<endl;
1545         totalprice *= 0.75;
1546     } else if (totalprice > 50000) {
1547         // gotoxy(38,20);cout<<"You have recieved 20% discount!!"<<endl;
1548         totalprice *= 0.8;
1549     } else if (totalprice > 25000) {
1550         // gotoxy(38,20);cout<<"You have recieved 15% discount!!";
1551         totalprice *= 0.85;
1552     } else if (totalprice > 10000) {
1553         // gotoxy(38,20);cout<<"You have recieved 10% discount!!";
1554         totalprice *= 0.9;
1555     }
1556 }
1557 void Bill::printbill() {
1558     gotoxy(2, 1);
1559     cout << "Current Date: ";
1560     putdate(currentdate, 15, 1);
1561     border(3, 2, 74, 22, 2);
1562     border(8, 5, 64, 16, 0);
1563     gotoxy(27, 4);
1564     cout << "C U S T O M E R   B I L L ";
1565     gotoxy(10, 6);
1566     cout << "CUSTOMER DETAILS";

```

```

1567 line(9, 7, 30, 'h', 1);
1568 gotoxy(10, 8);
1569 cout << "Drv. Lic. #: " << lic_no;
1570 gotoxy(46, 8);
1571 cout << "NAME: " << name;
1572 gotoxy(10, 9);
1573 cout << "Contact INFO: " << contact;
1574 line(9, 10, 30, 'h', 1);
1575 gotoxy(10, 11);
1576 cout << "RENTAL INFO";
1577 line(9, 12, 30, 'h', 1);
1578 gotoxy(10, 13);
1579 cout << "RentedCar_ID: " << carid;
1580 gotoxy(46, 13);
1581 cout << "CAR: " << car;
1582 gotoxy(10, 14);
1583 cout << "FROM DATE (dd/mm/yyyy): ";
1584 D.Put_fromdate(34, 14);
1585 gotoxy(10, 15);
1586 cout << "TO DATE (dd/mm/yyyy): ";
1587 D.Put_todate(32, 15);
1588 line(9, 16, 30, 'h', 1);
1589 gotoxy(10, 17);
1590 cout << "PRICE";
1591 line(9, 18, 30, 'h', 1);
1592 gotoxy(10, 19);
1593 cout << "Price/day for Car = Rs " << carprice;
1594 gotoxy(41, 19);
1595 cout << "No. of Days Rented = " << D.calc_diff();
1596 if (calc_returndaydiff()) cout << "+" << calc_returndaydiff();
1597 if (calc_returndaydiff()) {
1598     gotoxy(41, 20);
1599     cout << "Fine(Late Return) = Rs "
1600         << (1.5 * calc_returndaydiff() * carprice);
1601 }
1602 gotoxy(10, 20);
1603 cout << "Net Amount = Rs. " << totalprice;
1604 }
1605 void Bill::savebill() {
1606     ofstream bill;
1607     bill.open("bills.dat", ios::out | ios::app);
1608     bill.write((char *)this, sizeof(Bill));
1609     bill.close();
1610 }
1611 void ReturnCar() {
1612     char info[30];
1613     search:
1614     clrscr();
1615     border(3, 2, 74, 21, 2);
1616     gotoxy(24, 4);
1617     cout << "R E T U R N   A N D   B I L L I N G";
1618     gotoxy(24, 5);
1619     cout << "-----";
1620     gotoxy(8, 8);

```

```

1621 cout << "Drv. Lic. #: ";
1622 gotoxy(8, 10);
1623 cout << "Rent Car ID: ";
1624 gotoxy(8, 12);
1625 cout << "Name: ";
1626 gotoxy(8, 14);
1627 cout << "Email: ";
1628 gotoxy(37, 8);
1629 cout << "# Press ENTER to skip";
1630 gotoxy(37, 10);
1631 cout << "# Atleast one field should be entered";
1632 gotoxy(37, 12);
1633 cout << "# Drv. Lic. # = Licence No.of Customer";
1634 gotoxy(37, 14);
1635 cout << "# Rent Car ID = CarID of rented Car";
1636 line(6, 15, 45, 'h');
1637 gotoxy(21, 8);
1638 gets(info); // input lic_no.
1639 if (info[0] == NULL || SearchCust(N, info) == 0) // search lic_no.
1640 {
1641     gotoxy(21, 10);
1642     gets(info); // input rentcarid
1643     if (info[0] == NULL || SearchCust(N, info) == 0) // search rentcarid
1644     {
1645         gotoxy(14, 12);
1646         gets(info); // input name
1647         if (info[0] == NULL || SearchCust(N, info, 1) == 0) // search name
1648         {
1649             gotoxy(15, 14);
1650             gets(info); // input email
1651             if (SearchCust(N, info) == 0) // search email
1652             {
1653                 gotoxy(8, 16);
1654                 cout << "RECORD NOT FOUND !";
1655                 gotoxy(8, 18);
1656                 cout << "Do you wish to continue (y/n)....";
1657                 if (tolower(::op = getch()) == 'n') {
1658                     flushall();
1659                     return;
1660                 }
1661                 goto search;
1662             }
1663         }
1664     }
1665 }
1666 flushall();
1667 clrscr();
1668 rentcar_design();
1669 N.putcustomer();
1670 gotoxy(29, 4);
1671 cout << "RENTAL INFORMATION";
1672 gotoxy(29, 5);
1673 cout << "-----";
1674 gotoxy(20, 21);

```

```

1675 gotoxy(17, 21);
1676 cout << "      _";
1677 gotoxy(17, 22);
1678 cout << "View Bill";
1679 gotoxy(38, 21);
1680 cout << ' _';
1681 gotoxy(38, 22);
1682 cout << "Retry";
1683 gotoxy(58, 21);
1684 cout << ' _';
1685 gotoxy(58, 22);
1686 cout << "Cancel";
1687 choose:
1688 if (tolower(::op = getch()) == 'r')
1689     goto search;
1690 else if (::op == 'c') {
1691     flushall();
1692     return;
1693 } else if (::op == 'b') {
1694     clrscr();
1695     Bill B;
1696     B.makeBill(N);
1697 billing:
1698     clrscr();
1699     B.printbill();
1700     gotoxy(11, 22);
1701     cout << ' _';
1702     gotoxy(11, 23);
1703     cout << "Return Car";
1704     gotoxy(30, 22);
1705     cout << "      _";
1706     gotoxy(30, 23);
1707     cout << "Edit Price";
1708     gotoxy(49, 22);
1709     cout << "      _";
1710     gotoxy(49, 23);
1711     cout << "Edit Date";
1712     gotoxy(66, 22);
1713     cout << " _";
1714     gotoxy(66, 23);
1715     cout << "Exit";
1716     if (tolower(::op = getch()) == 'r') {
1717         clrscr();
1718         border(3, 2, 74, 21, 2);
1719         gotoxy(8, 4);
1720         cout << "Confirm Return Car [y/n] > ";
1721         cin >> ::op;
1722         if (tolower(op) == 'y') {
1723             status_edit(N.ret_rentcarid());
1724             deletecust(N);
1725             B.savebill();
1726             gotoxy(8, 6);
1727             cout << "Car Returned And Bill Saved.....";
1728             getch();

```



```

1729     } else
1730         goto billing;
1731 } else if (::op == 'p') {
1732     B.editprice();
1733     B.test();
1734     getch();
1735     goto billing;
1736 } else if (::op == 'd') {
1737     B.editdate();
1738     goto billing;
1739 } else if (::op == 'x') {
1740     flushall();
1741     clrscr();
1742     return;
1743 } else {
1744     cout << '\a';
1745     goto billing;
1746 }
1747 } else {
1748     cout << '\a';
1749     goto choose;
1750 }
1751 }
1752 void report_menu() {
1753 menu:
1754     clrscr();
1755     border(3, 2, 74, 21, 2);
1756     border(23, 4, 37, 17, 0);
1757     gotoxy(30, 6);
1758     cout << "R E P O R T S   M E N U";
1759     gotoxy(29, 7);
1760     cout << "-----";
1761     gotoxy(27, 9);
1762     cout << " [1] View All Car Reports";
1763     gotoxy(27, 11);
1764     cout << " [2] View All Customer Report";
1765     gotoxy(27, 13);
1766     cout << " [3] View Customer - Car Report";
1767     gotoxy(27, 15);
1768     cout << " [4] GO BACK\n";
1769     gotoxy(1, 1);
1770     ventry(&::op);
1771     switch (::op) {
1772     case '1':
1773         clrscr();
1774         C.CarTable();
1775         getch();
1776         break;
1777     case '2':
1778         clrscr();
1779         N.customer_report();
1780         getch();
1781         break;
1782     case '3':

```

```

1783     clrscr();
1784     N.CustomerCar_table();
1785     getch();
1786     break;
1787     case '4':
1788         clrscr();
1789         return;
1790     default:
1791         cout << '\a';
1792     }
1793     goto menu;
1794 }
1795 void status_editmenu() {
1796 menu:
1797     clrscr();
1798     cout << "\n\t\t\tEDIT CAR STATUS\n\n";
1799     cout << "\n\t[1] EDIT All Status to [ AVAILABLE ] \n ";
1800     cout << "\n\t[2] EDIT All Status to [ BOOKED! ]\n";
1801     cout << "\n\t[3] Car Status Switch\n";
1802     cout << "\n\t[4] GO BACK\n";
1803     cout << "\nEnter your option => ";
1804     ventry(&::op);
1805     char carid[6];
1806     switch (::op) {
1807     case '1':
1808         clrscr();
1809         cout << '\a';
1810         status_edit("1");
1811         getch();
1812         break;
1813     case '2':
1814         clrscr();
1815         cout << '\a';
1816         status_edit("0");
1817         getch();
1818         break;
1819     case '3':
1820         clrscr();
1821         cout << '\a';
1822         cout << "\n\tEnter CarID to Update: ";
1823         gets(carid);
1824         status_edit(carid);
1825         getch();
1826         break;
1827     case '4':
1828         clrscr();
1829         return;
1830     default:
1831         cout << '\a';
1832     }
1833     goto menu;
1834 }
1835 void recordedit_menu() {
1836 menu:

```

```

1837 clrscr();
1838 border(3, 2, 74, 21, 2);
1839 border(23, 4, 37, 17, 0);
1840 gotoxy(27, 5);
1841 cout << "\tEDIT RECORD | MENU |";
1842 gotoxy(27, 8);
1843 cout << " [1] Update Car";
1844 gotoxy(27, 10);
1845 cout << " [2] Update Customer";
1846 gotoxy(27, 12);
1847 cout << " [3] DELETE Car";
1848 gotoxy(27, 14);
1849 cout << " [4] DELETE Customer";
1850 gotoxy(27, 16);
1851 cout << " [5] EDIT Status";
1852 gotoxy(27, 18);
1853 cout << " [6] GO BACK\n";
1854 gotoxy(1, 1);
1855 ventry(&::op);
1856 switch (::op) {
1857     case '1':
1858         clrscr();
1859         UpdateCar();
1860         break;
1861     case '2':
1862         clrscr();
1863         // UpdateCust();
1864         break;
1865     case '3':
1866         clrscr();
1867         DeleteCar();
1868         break;
1869     case '4':
1870         clrscr();
1871         DeleteCust();
1872         break;
1873     case '5':
1874         clrscr();
1875         status_editmenu();
1876         break;
1877     case '6':
1878         clrscr();
1879         return;
1880     default:
1881         cout << "\a";
1882 }
1883 goto menu;
1884 }
1885 void SearchDisplay_menu() {
1886 menu:
1887     clrscr();
1888     border(3, 2, 74, 22, 2);
1889     border(23, 4, 37, 18, 0);
1890     gotoxy(27, 6);

```

```

1891 cout << "\t SEARCH AND DISPLAY";
1892 gotoxy(27, 10);
1893 cout << " [1] Display ALL Cars";
1894 gotoxy(27, 12);
1895 cout << " [2] Display ALL Customers";
1896 gotoxy(27, 14);
1897 cout << " [3] SEARCH Car";
1898 gotoxy(27, 16);
1899 cout << " [4] SEARCH Customer";
1900 gotoxy(27, 18);
1901 cout << " [5] GO BACK\n";
1902 gotoxy(1, 1);
1903 ventry(&::op);
1904 switch (::op) {
1905     case '1':
1906         clrscr();
1907         DisplayCar();
1908         break;
1909     case '2':
1910         clrscr();
1911         DisplayCust();
1912         break;
1913     case '3':
1914         clrscr();
1915         SearchCar_UI();
1916         break;
1917     case '4':
1918         clrscr();
1919         SearchCust_UI();
1920         break;
1921     case '5':
1922         clrscr();
1923         return;
1924     default:
1925         cout << '\a';
1926 }
1927 goto menu;
1928 }
1929 void menu() {
1930     clrscr();
1931     border(3, 2, 74, 22, 2);
1932     // border(23,3,37,20,4);
1933     menu:
1934     clrscr();
1935     // border(1,1,78,24,2);
1936     border(3, 2, 74, 22, 2);
1937     border(23, 3, 37, 20, 0);
1938     gotoxy(32, 4);
1939     cout << "/ M A I N   M E N U /";
1940     gotoxy(32, 5);
1941     cout << "-----";
1942     gotoxy(27, 7);
1943     cout << "\t[1] RENT A CAR";
1944     gotoxy(27, 9);

```

```
1945 cout << "\t[2] ADD CAR";
1946 gotoxy(27, 11);
1947 cout << "\t[3] SEARCH & DISPLAY";
1948 gotoxy(27, 13);
1949 cout << "\t[4] RETURN & BILLING";
1950 gotoxy(27, 15);
1951 cout << "\t[5] REPORTS";
1952 gotoxy(27, 17);
1953 cout << "\t[6] EDIT RECORD";
1954 gotoxy(27, 19);
1955 cout << "\t[7] ABOUT";
1956 gotoxy(27, 21);
1957 cout << "\t[8] EXIT\n";
1958 gotoxy(1, 1);
1959 ventry(&::op, 4, 1);
1960 switch (::op) {
1961     case '2':
1962         clrscr();
1963         AddCar();
1964         break;
1965     case '1':
1966         clrscr();
1967         AddCustomer();
1968         break;
1969     case '3':
1970         clrscr();
1971         SearchDisplay_menu();
1972         break;
1973     case '5':
1974         clrscr();
1975         report_menu();
1976         break;
1977     case '6':
1978         clrscr();
1979         recordedit_menu();
1980         break;
1981     case '4':
1982         clrscr();
1983         ReturnCar();
1984         break;
1985     case '7':
1986         clrscr();
1987         About();
1988         getch();
1989         break;
1990     case '8':
1991         clrscr();
1992         cout << "PROJECT HALTED";
1993         exit(-1);
1994     default:
1995         cout << '\a';
1996 }
1997 goto menu;
1998 }
```

```
1999 void main() {  
2000     welcome();  
2001     L.check();  
2002     menu();  
    }
```