



ESTRUCTURAS DE DATOS

DESARROLLAR METODOS FALTANTES

Programación I ITDSC3-003-SDN

NOMBRE: *Sabrina Amparo Valdez |2025-1320*

FACILITADOR: *Starling Alfredo Germosén Reynoso*

SECCIÓN: *#1 lunes*

CARRERA: *Desarrollo de Software*



complete los métodos de la clase "contactes" o desarrolle su propia solución que cumpla con el requerimiento 1 del proyecto "Agenda Personal".

```

4 //sabrina amparo valdez 20251326
5 0 referencias
6 class Program
7 {
8     0 referencias
9     static void Main(string[] args)
10    {
11        Console.WriteLine("Mi Agenda Perrón");
12        Console.WriteLine("Bienvenido a tu lista de contactes");
13
14        Dictionary<int, string> names = new Dictionary<int, string>();
15        Dictionary<int, string> phones = new Dictionary<int, string>();
16        Dictionary<int, string> emails = new Dictionary<int, string>();
17        List<string> addresses = new List<string>();
18        List<int> ids = new List<int>();
19
20        bool running = true;
21        while (running)
22        {
23            Console.WriteLine("\n--- MENÚ PRINCIPAL ---");
24            Console.Write("1. Agregar Contacto  ");
25            Console.Write("2. Ver Contactos    ");
26            Console.Write("3. Buscar Contactos  ");
27            Console.Write("4. Modificar Contacto ");
28            Console.Write("5. Eliminar Contacto  ");
29            Console.WriteLine("6. Salir");
30            Console.Write("Elige una opción: ");
31
32            try
33            {
34                int choice = Convert.ToInt32(Console.ReadLine());
35
36                switch (choice)
37                {
38                    case 1:
39                        AddContact(ref names, ref phones, ref emails, ref addresses, ref ids);
40                        break;
41                    case 2:
42                        ViewContacts(ref names, ref phones, ref emails, ref addresses, ref ids);
43                        break;
44                    case 3:
45                        SearchContact(ref names, ref phones, ref emails, ref addresses, ref ids);
46                        break;
47                    case 4:
48                        EditContact(ref names, ref phones, ref emails, ref addresses, ref ids);
49                        break;
50                    case 5:
51                        DeleteContact(ref names, ref phones, ref emails, ref addresses, ref ids);
52                        break;
53                    case 6:
54                        running = false;
55                        Console.WriteLine("¡Hasta luego!");
56                        break;
57                    default:
58                        Console.WriteLine("Opción no válida");
59                        break;
60                }
61            }
62        }
63    }
64 }
```

```

59 }
60 catch (FormatException)
61 {
62     Console.WriteLine("Error: Por favor ingresa un número válido.");
63 }
64 }
65 }
66
67 1 referencia
68 static void AddContact(ref Dictionary<int, string> names, ref Dictionary<int, string> phones, ref Dictionary<int, string> emails, ref List<string> addresses, ref List<int> ids)
69 {
70     Console.WriteLine("\n--- AGREGAR CONTACTO ---");
71     Console.WriteLine("Vamos a agregar ese contacte que te trae loco.");
72
73     try
74     {
75         int id = ids.Count > 0 ? ids.Max() + 1 : 1;
76         ids.Add(id);
77
78         Console.Write("Digite el Nombre: ");
79         var name = Console.ReadLine();
80         if (string.IsNullOrEmpty(name))
81         {
82             Console.WriteLine("Error: El nombre no puede estar vacío.");
83             ids.Remove(id);
84             return;
85         }
86         names.Add(id, name);
87
88         Console.Write("Digite el Teléfono: ");
89         var telefono = Console.ReadLine();
90         phones.Add(id, telefono);
91
92         Console.Write("Digite el Email: ");
93         var email = Console.ReadLine();
94         emails.Add(id, email);
95
96         Console.Write("Digite la dirección: ");
97         var address = Console.ReadLine();
98         addresses.Add(address);
99
100        Console.WriteLine("¡Contacto agregado exitosamente!");
101    }
102    catch (Exception ex)
103    {
104        Console.WriteLine($"Error al agregar contacto: {ex.Message}");
105    }
106 }
107
108 4 referencias
109 static void ViewContacts(ref Dictionary<int, string> names, ref Dictionary<int, string> phones, ref Dictionary<int, string> emails, ref List<string> addresses, ref List<int> ids)
110 {
111     Console.WriteLine("\n--- LISTA DE CONTACTOS ---");
112 }
```



```

108 Console.WriteLine("\n--- LISTA DE CONTACTOS ---");
109
110
111 if (ids.Count == 0)
112 {
113     Console.WriteLine("No hay contactos en la agenda.");
114     return;
115 }
116
117 Console.WriteLine("Id\tNombre\t\tTeléfono\t\tEmail\t\t\tDirección");
118 Console.WriteLine("-----");
119
120 foreach (var id in ids)
121 {
122     if (names.ContainsKey(id) && phones.ContainsKey(id) && emails.ContainsKey(id))
123     {
124         string direccion = id <= addresses.Count ? addresses[id - 1] : "No especificada";
125         Console.WriteLine($"{id}\t{names[id]}\t\t{phones[id]}\t\t{emails[id]}\t\t{direccion}");
126     }
127 }
128
129
130 1 referencia
131 static void EditContact(ref Dictionary<int, string> names, ref Dictionary<int, string> phones, ref Dictionary<int, string> emails, ref List<string> addresses, ref List<int> ids)
132 {
133     Console.WriteLine("\n--- MODIFICAR CONTACTO ---");
134
135     if (ids.Count == 0)
136     {
137         Console.WriteLine("No hay contactos para modificar.");
138         return;
139     }
140
141     ViewContacts(ref names, ref phones, ref emails, ref addresses, ref ids);
142
143     try
144     {
145         Console.Write("Digite el Id de Contacto Para Editar: ");
146         int idSeleccionado = Convert.ToInt32(Console.ReadLine());
147
148         if (!ids.Contains(idSeleccionado))
149         {
150             Console.WriteLine("Error: El ID especificado no existe.");
151             return;
152         }
153
154         var nombreSeleccionado = names[idSeleccionado];
155         var telefonoSeleccionado = phones[idSeleccionado];
156         var emailSeleccionado = emails[idSeleccionado];
157         string direccionSeleccionada = addresses[idSeleccionado - 1];
158
159         Console.Write($"El nombre actual es: {nombreSeleccionado}, Digite el Nuevo Nombre: ");
160         var nuevoNombre = Console.ReadLine();
161         if (!string.IsNullOrEmpty(nuevoNombre))
162         {
163             names[idSeleccionado] = nuevoNombre;
164         }
165     }
166 }

```

```

165 Console.Write($"El Teléfono actual es: {telefonoSeleccionado}, Digite el Nuevo Teléfono: ");
166 var nuevoTelefono = Console.ReadLine();
167 phones[idSeleccionado] = nuevoTelefono;
168
169 Console.Write($"El Email actual es: {emailSeleccionado}, Digite el Nuevo Email: ");
170 var nuevoEmail = Console.ReadLine();
171 emails[idSeleccionado] = nuevoEmail;
172
173 Console.Write($"La dirección actual es: {direccionSeleccionada}, Digite la nueva dirección: ");
174 var nuevaDireccion = Console.ReadLine();
175 addresses[idSeleccionado - 1] = nuevaDireccion;
176
177 Console.WriteLine("Contacto modificado exitosamente!");
178 }
179 catch (FormatException)
180 {
181     Console.WriteLine("Error: Por favor ingresa un ID válido.");
182 }
183 catch (Exception ex)
184 {
185     Console.WriteLine($"Error al modificar contacto: {ex.Message}");
186 }
187
188
189 1 referencia
190 static void DeleteContact(ref Dictionary<int, string> names, ref Dictionary<int, string> phones, ref Dictionary<int, string> emails, ref List<string> addresses, ref List<int> ids)
191 {
192     Console.WriteLine("\n--- ELIMINAR CONTACTO ---");
193
194     if (ids.Count == 0)
195     {
196         Console.WriteLine("No hay contactos para eliminar.");
197         return;
198     }
199
200     ViewContacts(ref names, ref phones, ref emails, ref addresses, ref ids);
201
202     try
203     {
204         Console.Write("Digite el Id de Contacto Para Eliminar: ");
205         int idSeleccionado = Convert.ToInt32(Console.ReadLine());
206
207         if (!ids.Contains(idSeleccionado))
208         {
209             Console.WriteLine("Error: El ID especificado no existe.");
210             return;
211         }
212
213         Console.WriteLine($"¿Está seguro que desea eliminar el contacto '{names[idSeleccionado]}'?");
214         Console.Write("1. Sí, 2. No: ");
215         int opcion = Convert.ToInt32(Console.ReadLine());
216     }
217 }

```




```

215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
119
```

```

267     MostrarDetalleContacto(idseleccionado, ref names, ref phones, ref emails, ref addresses);
268 }
269 else
270 {
271     Console.WriteLine("Error: El ID especificado no existe.");
272 }
273 }
274 else if (opcionBusqueda == 2)
275 {
276     Console.Write("Digite el nombre o parte del nombre a buscar: ");
277     string nombreBuscado = Console.ReadLine().ToLower();
278
279     bool encontrado = false;
280     Console.WriteLine("\n--- RESULTADOS DE BÚSQUEDA ---");
281
282     foreach (var id in ids)
283     {
284         if (names.ContainsKey(id) && names[id].ToLower().Contains(nombreBuscado))
285         {
286             MostrarDetalleContacto(id, ref names, ref phones, ref emails, ref addresses);
287             encontrado = true;
288         }
289     }
290
291     if (!encontrado)
292     {
293         Console.WriteLine("No se encontraron contactos con ese nombre.");
294     }
295 }
296 else
297 {
298     Console.WriteLine("Opción de búsqueda no válida.");
299 }
300 }
301 catch (FormatException)
302 {
303     Console.WriteLine("Error: Por favor ingresa un número válido.");
304 }
305 }
306
307 2 referencias
308 static void MostrarDetalleContacto(int id, ref Dictionary<int, string> names, ref Dictionary<int, string> phones, ref Dictionary<int, string> emails, ref List<string> addresses)
309 {
310     Console.WriteLine($" \n--- DETALLES DEL CONTACTO ID: {id} ---");
311     Console.WriteLine($"Nombre: {names[id]}");
312     Console.WriteLine($"Teléfono: {phones[id]}");
313     Console.WriteLine($"Email: {emails[id]}");
314     Console.WriteLine($"Dirección: {addresses[id - 1]}");
315     Console.WriteLine("-----");
316 }

```

```
@_C:\oservados\source\temp\ - 1 -
Mi Agenda Perrón
Bienvenido a tu lista de contactes

--- MENÚ PRINCIPAL ---
1. Agregar Contacto      2. Ver Contactos      3. Buscar Contactos      4. Modificar Contacto      5. Eliminar Contacto      6. Salir
Elige una opción: 2

--- LISTA DE CONTACTOS ---
No hay contactos en la agenda.

--- MENÚ PRINCIPAL ---
1. Agregar Contacto      2. Ver Contactos      3. Buscar Contactos      4. Modificar Contacto      5. Eliminar Contacto      6. Salir
Elige una opción: 1

--- AGREGAR CONTACTO ---
Vamos a agregar ese contacte que te trae loco.
Digite el Nombre: sabrina
Digite el Teléfono: 8095555432
Digite el Email: amparovsbrina@gmail.com
Digite la dirección: la loma #12 parte atras
¡Contacto agregado exitosamente!

--- MENÚ PRINCIPAL ---
1. Agregar Contacto      2. Ver Contactos      3. Buscar Contactos      4. Modificar Contacto      5. Eliminar Contacto      6. Salir
Elige una opción: 4

--- MODIFICAR CONTACTO ---

--- LISTA DE CONTACTOS ---
Id      Nombre      Teléfono      Email      Dirección
-----
1      sabrina      8095555432      amparovsbrina@gmail.com      la loma #12 parte atras
Digite el Id de Contacto Para Editar: 1
El nombre actual es: sabrina, Digite el Nuevo Nombre: amparo
El Teléfono actual es: 8095555432, Digite el Nuevo Teléfono:
```

En este caso, agregué mejoras al código para hacerlo más robusto y fácil de usar, incluyendo manejo de errores con try-catch para evitar que el programa se cierre pues inesperadamente, también validaciones que verifican si existen contactos antes de realizar operaciones, una función de búsqueda mejorada que permite encontrar contactos tanto por ID como por nombre, y mensajes más claros que en mi experiencia pues me han ayudado durante todo el proceso.