

# GESTIÓN DE DATOS DE ESTUDIANTES



Asignatura: Big Data  
Integrantes: Ignacio Arancibia  
Eduardo Silva  
Profesor: Edio Mardones

11/09/2024

Models

# CREACIÓN DE COLECCIONES

```

1  ## Lógica de conexión para la BD
2
3  from connexion import *
4
5  def InsertarPersona(p):
6      db = client.ejemplo1
7      collection = db['Person']
8      collection.insert_one(p)

```



## Colección para alumnos

```

15 def FilterAlumno(search):
16     db = client.ejemplo1
17     collection = db['Person']
18     show_data_count = len(list(collection.find(search)))
19     if show_data_count > 0 :
20         show_data = collection.find(search)
21         for i in show_data:
22             print(i)
23     else:
24         print("No existe el rut ingresado")

```

```

30 def VizualizarAlumno():
31     db = client.ejemplo1
32     collection = db['Person']
33     show_data = collection.find()
34     for data in show_data:
35         print(data)
36 def EliminarAlumno(rut):
37     db = client.ejemplo1
38     collection = db['Person']
39     collection.delete_many({"Rut": rut})
40
41 def ActualizarAlumno(rut, update):
42     db = client.ejemplo1
43     collection = db['Person']
44     collection.update_one({"Rut": rut}, {'$set': update})

```



## Colección para cursos

```

48 def VizualizarCurso():
49     db = client.ejemplo1
50     collection = db['Curso']
51     show_data = collection.find()
52     for data in show_data:
53         print(data)

55 def FilterCurso(search):
56     db = client.ejemplo1
57     collection = db['Curso']
58     show_data_count = len(list(collection.find(search)))
59     if show_data_count > 0 :
60         show_data = collection.find(search)
61         for i in show_data:
62             print(i)
63     else:
64         print("No existe el código de curso ingresado")

```

```

70 def ActualizarCurso(rut, update):
71     db = client.ejemplo1
72     collection = db['Curso']
73     collection.update_one({"Rut_estudiante": rut}, {'$set': update})
74
75 def EliminarCurso(rut):
76     db = client.ejemplo1
77     collection = db['Curso']
78     collection.delete_one({"Rut_estudiante": rut})
79
79 def InsertCurso(p):
80     db = client.ejemplo1
81     collection = db['Curso']
82     collection.insert_one(p)

```



# CREACIÓN DE FUNCIONES

Funciones

```
from models import *

def Ingresar_alumno():
    while True:
        nombre = input("Ingrese el nombre\n")
        if len(nombre) > 0:
            break
        else:
            print("No ingresó un nombre válido. Intente nuevamente.")
    while True:
        try:
            edad = int(input("Ingresa edad\n"))
            if edad < 0 or edad > 90:
                print("No ingresó una edad válida. Intente nuevamente.")
            else:
                break
        except ValueError:
            print("No ingresó un número válido. Intente nuevamente.")
```

```
22 while True:
23     rut = input("Ingrese el rut\n")
24     if len(rut) == 8 or len(rut) == 9:
25         break
26     else:
27         print("No ingresó un rut válido. Intente nuevamente.")
28 while True:
29     carrera = input("Ingrese la carrera\n")
30     if len(carrera) > 0:
31         break
32     else:
33         print("No ingresó una carrera válida. Intente nuevamente.")
34 while True:
35     try:
36         año_ingreso = int(input("Ingrese el año de ingreso\n"))
37         if año_ingreso < 1900 or año_ingreso > 2022:
38             print("No ingresó un año de ingreso válido. Intente nuevamente.")
39         else:
40             break
41     except ValueError:
42         print("No ingresó un número válido. Intente nuevamente.")
```

```
48 p = {}
49 if len(nombre) > 0:
50     p["Nombre"] = nombre
51 if len(str(edad)) > 0:
52     p["Edad"] = edad
53 if len(str(rut)) > 0:
54     p["Rut"] = rut
55 if len(carrera) > 0:
56     p["Carrera"] = carrera
57 if len(str(año_ingreso)) > 0:
58     p["Año_ingreso"] = año_ingreso
59 InsertarPersona(p)
```



```

65 def Filtro_alumno():
66     while True:
67         try:
68             opcion = int(input("""
69                 1 Buscar por nombre
70                 2 Buscar por edad
71                 3 Buscar por rut
72                 4 Buscar por carrera
73                 5 Buscar por año de ingreso
74                 6 Salir
75                 --> """))
76             search_criteria = {}
77             if opcion == 1:
78                 search = input("Ingresa nombre\n")
79                 search_criteria["Nombre"] = search
80             elif opcion == 2:
81                 search = int(input("Ingresa edad\n"))
82                 search_criteria["Edad"] = search
83             elif opcion == 3:
84                 search = input("Ingresa rut\n")
85                 search_criteria["Rut"] = search
86             elif opcion == 4:
87                 search = input("Ingresa carrera\n")
88                 search_criteria["Carrera"] = search
89             elif opcion == 5:
90                 search = int(input("Ingresa año de ingreso\n"))
91                 search_criteria["Año_ingreso"] = search
92             elif opcion == 6:
93                 break
94             else:
95                 print("Opción no válida. Intente nuevamente.")
96                 continue
97
98             FilterAlumno(search_criteria)
99         except ValueError:
100             print("\nVALOR INCORRECTO\n")

```

```

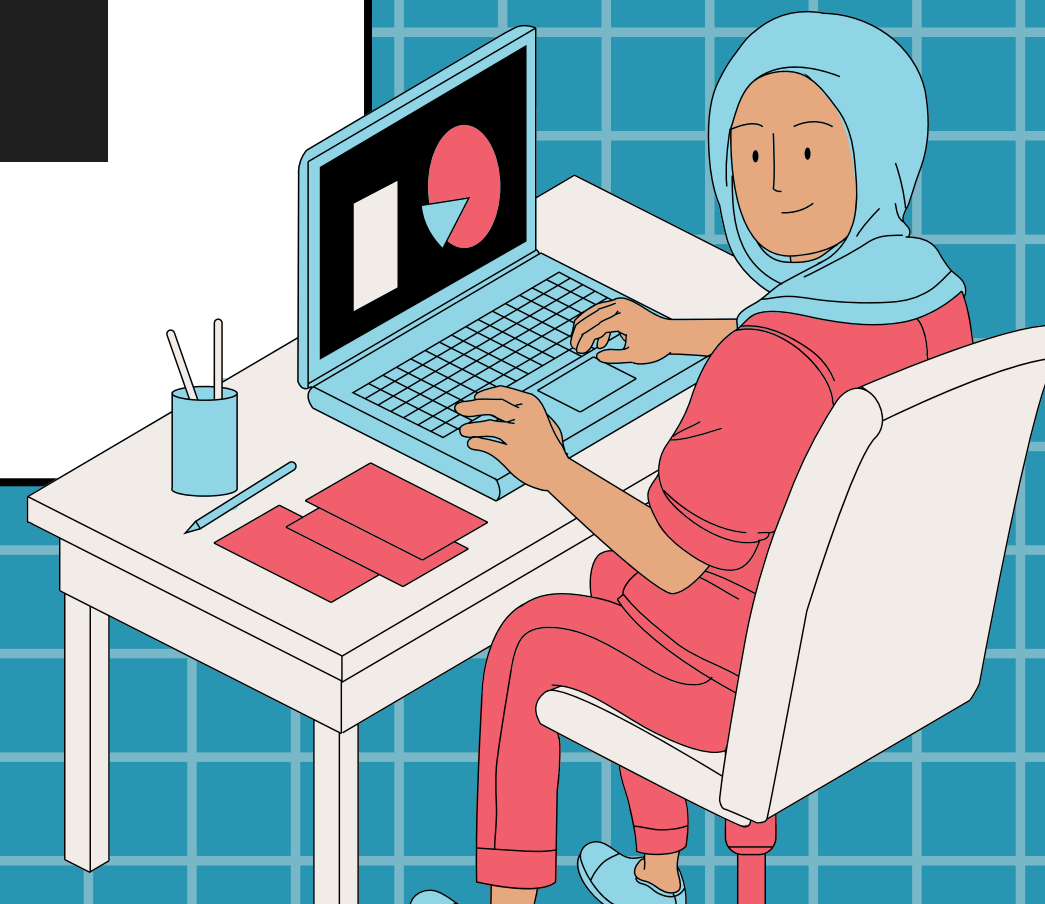
102 def Actualizar_alumno():
103     while True:
104         rut = input("Ingrese el rut del alumno a actualizar\n")
105         if len(rut) == 8 or len(rut) == 9:
106             break
107         else:
108             print("No ingresó un rut válido. Intente nuevamente.")
109     update_fields = {}
110     while True:
111         nombre = input("Ingrese el nuevo nombre (deje en blanco para no cambiar)\n")
112         if len(nombre) > 0:
113             update_fields["Nombre"] = nombre
114             break
115     while True:
116         edad = input("Ingrese la nueva edad (deje en blanco para no cambiar)\n")
117         if len(edad) > 0:
118             try:
119                 edad = int(edad)
120                 if edad < 0 or edad > 90:
121                     print("No ingresó una edad válida. Intente nuevamente.")
122                 else:
123                     update_fields["Edad"] = edad
124             except ValueError:
125                 print("No ingresó un número válido. Intente nuevamente.")
126             break
127     while True:
128         carrera = input("Ingrese la nueva carrera (deje en blanco para no cambiar)\n")
129         if len(carrera) > 0:
130             update_fields["Carrera"] = carrera
131             break
132     while True:
133         año_ingreso = input("Ingrese el nuevo año de ingreso (deje en blanco para no cambiar)\n")
134         if len(año_ingreso) > 0:
135             try:
136                 año_ingreso = int(año_ingreso)
137                 if año_ingreso < 1900 or año_ingreso > 2022:
138                     print("No ingresó un año de ingreso válido. Intente nuevamente.")
139                 else:
140                     update_fields["Año_ingreso"] = año_ingreso
141             except ValueError:
142                 print("No ingresó un número válido. Intente nuevamente.")
143             break
144
145     if update_fields:
146         ActualizarAlumno(rut, update_fields)
147         print("Alumno actualizado correctamente.")

```

```

151 def Eliminar_alumno():
152     while True:
153         rut = input("Ingrese el rut del alumno a eliminar\n")
154         if len(rut) == 8 or len(rut) == 9:
155             break
156         else:
157             print("No ingresó un rut válido. Intente nuevamente.")
158     EliminarAlumno(rut)
159     print("Alumno eliminado correctamente.")
160

```







```

163 def Ingresar_curso():
164     while True:
165         curso = input("Ingrese el curso\n")
166         if len(curso) > 0:
167             break
168         else:
169             print("No ingresó un curso válido. Intente nuevamente.")
170
171     while True:
172         codigo_curso = input("Ingrese el código del curso\n")
173         if len(codigo_curso) > 0:
174             break
175         else:
176             print("No ingresó un código de curso válido. Intente nuevamente.")
177
178     while True:
179         rut_estudiante = input("Ingrese el rut del estudiante\n")
180         if len(rut_estudiante) == 8 or len(rut_estudiante) == 9:
181             break
182         else:
183             print("No ingresó un rut válido. Intente nuevamente.")
184
185     while True:
186         try:
187             nota_final = float(input("Ingrese la nota final\n"))
188             if nota_final < 1 or nota_final > 7:
189                 print("No ingresó una nota final válida. Intente nuevamente.")
190             else:
191                 break
192         except ValueError:
193             print("No ingresó un número válido. Intente nuevamente.")

```

```

p = {}
if len(curso) > 0:
    p["Curso"] = curso
if len(codigo_curso) > 0:
    p["Codigo_curso"] = codigo_curso
if len(rut_estudiante) > 0:
    p["Rut_estudiante"] = rut_estudiante
if len(str(nota_final)) > 0:
    p["Nota_final"] = nota_final

```

InsertCurso(p)



```

213 def Filtro_curso():
214     while True:
215         try:
216             opcion = int(input("""
217                 1 Buscar por curso
218                 2 Buscar por código de curso
219                 3 Buscar por rut
220                 4 Salir
221                 --> """))
222             search_criteria = {}
223             if opcion == 1:
224                 search = input("Ingresa curso\n")
225                 search_criteria["Curso"] = search
226             elif opcion == 2:
227                 search = input("Ingresa código de curso\n")
228                 search_criteria["Codigo_curso"] = search
229             elif opcion == 3:
230                 search = input("Ingresa rut\n")
231                 search_criteria["Rut_estudiante"] = search
232             elif opcion == 4:
233                 break
234             else:
235                 print("Opción no válida. Intente nuevamente.")
236                 continue
237
238             FilterCurso(search_criteria)
239         except ValueError:
240             print("\nVALOR INCORRECTO\n")

```





```

242 def Actualizar_curso():
243     while True:
244         rut = input("Ingrese el rut del alumno a actualizar\n")
245         if len(rut) == 8 or len(rut) == 9:
246             break
247         else:
248             print("No ingresó un rut válido. Intente nuevamente.")
249
250     update_fields = {}
251     while True:
252         curso = input("Ingrese el nuevo curso (deje en blanco para no cambiar)\n")
253         if len(curso) > 0:
254             update_fields["Curso"] = curso
255             break
256     while True:
257         codigo_curso = input("Ingrese el código del nuevo curso (deje en blanco para no cambiar)\n")
258         if len(codigo_curso) > 0:
259             update_fields["Codigo_curso"] = codigo_curso
260             break
261
262     while True:
263         nota = input("Ingrese la nueva nota (deje en blanco para no cambiar)\n")
264         if len(nota) > 0:
265             try:
266                 nota = float(nota)
267                 if nota < 1 or nota > 7:
268                     print("No ingresó una nota válida. Intente nuevamente.")
269                 else:
270                     update_fields["Nota_final"] = nota
271             except ValueError:
272                 print("No ingresó un número válido. Intente nuevamente.")
273             break
274
275     if update_fields:
276         ActualizarCurso(rut, update_fields)
277         print("Curso actualizado correctamente.")
278     else:
279         print("No se realizaron cambios.")

```

```

281 def Eliminar_curso():
282     while True:
283         rut = input("Ingrese el rut del alumno a eliminar curso\n")
284         if len(rut) == 8 or len(rut) == 9:
285             break
286         else:
287             print("No ingresó un rut válido. Intente nuevamente.")
288     EliminarCurso(rut)
289     print("Curso eliminado correctamente.")

```

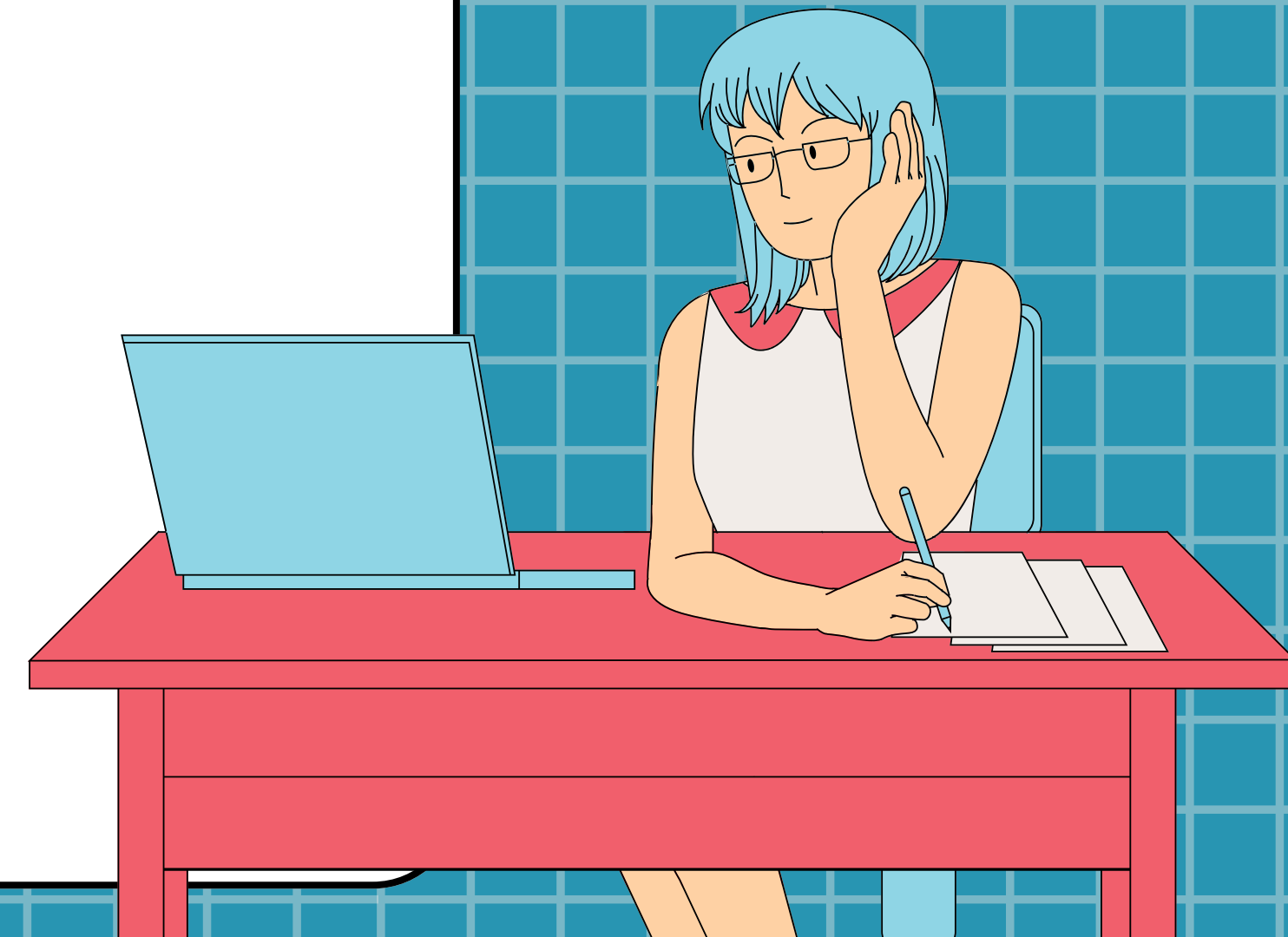
# CREACIÓN DE MENÚ

Main

```

3  from funciones import *
4  def Seleccionar_opcion_alumno():
5      while True:
6          try:
7              opcion = int(input("""
8                  1 Para Ingresar
9                  2 Para Vizualizar datos
10                 3 Para Busqueda por filtro
11                 4 Para Actualizar datos
12                 5 Para Eliminar datos
13                 6 Menu Principal
14
15                 --> """))
16
17                 if opcion == 1:
18                     Ingresar_alumno()
19                 elif opcion == 2:
20                     Vizualizar_alumno()
21                 elif opcion == 3:
22                     Filtro_alumno()
23                 elif opcion == 4:
24                     Actualizar_alumno()
25                 elif opcion == 5:
26                     Eliminar_alumno()
27                 elif opcion == 6:
28                     break
29             except:
30                 print ("\nVALOR INCORRECTO\n")

```





```

31  def Seleccionar_opcion_curso():
32      while True:
33          try:
34              opcion = int(input("""
35                  1 Para Ingresar
36                  2 Para Vizualizar datos
37                  3 Para Busqueda por filtro
38                  4 Para Actualizar datos
39                  5 Para Eliminar datos
40                  6 Menu Principal
41
42                  --> """))
43          if opcion == 1:
44              Ingresar_curso()
45          elif opcion == 2:
46              Vizualizar_curso()
47          elif opcion == 3:
48              Filtro_curso()
49          elif opcion == 4:
50              Actualizar_curso()
51          elif opcion == 5:
52              Eliminar_curso()
53          elif opcion == 6:
54              break
55      except:
56          print ("\nVALOR INCORRECTO\n")

```





```

58  ✓ def Main():
59  ✓     while True:
60  ✓         try:
61  ✓             opcion = int(input("""
62  ✓                 1 Para Estudiantes
63  ✓                 2 Para Cursos
64  ✓                 3 Para salir
65  ✓
66  ✓                 --> """))
67  ✓             if opcion == 1:
68  ✓                 Seleccionar_opcion_alumno()
69  ✓             elif opcion == 2:
70  ✓                 Seleccionar_opcion_curso()
71  ✓             elif opcion == 3:
72  ✓                 break
73  ✓         except:
74  ✓             print ("\nVALOR INCORRECTO\n")
75  Main()

```

# CONEXIÓN A MONGODB

conexión

```
1 import pymongo as pym
2 client = pymongo.MongoClient('mongodb+srv://edupanpan.sa270.mongodb.net/',
3                               27017, connect = False,
4                               username = , password = )
5
6
```



# BIBLIOGRAFÍA

Collection Methods — MongoDB Manual. (s. f.). <https://www.mongodb.com/docs/v5.3/reference/method/js-collection/>



**GRACIAS POR SU  
ATENCIÓN**

