

Calculo IMC*

Emerson Aldair Pérez Rivera, 201902852^{1, **}

¹*Facultad de Ingeniería, Universidad de San Carlos, Edificio T1, Ciudad Universitaria, Zona 12, Guatemala.*

I. OBJETIVOS

- Poder Agregar,Actualizar y Borrar datos desde oc-tave
- Guardar y Leer informacion en un archivo .txt

II. CODIGOS UTILIZADOS

A. Codigo Octave

```

1 pkg load database
2
3 conn = pq_connect(setdbopts("dbname", "imc", "
    host", "localhost", "port", "5432", "user",
    "postgres", "password", "kuto"));
4
5
6 %Ingresar
7 %nombre = 'Juan';
8 %peso = 75;
9 %altura = 180;
10
11 %insert_query = "INSERT INTO imc (nombre, peso,
    altura) VALUES ($1, $2, $3);";
12 %pq_exec_params(conn, insert_query, {nombre,
    peso, altura});
13
14
15 % Borrar un registro
16 %nombre = 'Juan';
17
18 %delete_query = "DELETE FROM imc WHERE nombre =
    $1;";
19 %pq_exec_params(conn, delete_query, {nombre});
20
21 %N = pq_exec_params(conn, 'select * from imc;');
22
23
24
25 % Actualizar el peso y altura de un registro
26 %nuevo_peso = 80;
27 %nueva_altura = 182;
28 %nombre = 'Juan';
29
30 %update_query = "UPDATE imc SET peso = $1,
    altura = $2 WHERE nombre = $3;";
31 %pq_exec_params(conn, update_query, {nuevo_peso,
    nueva_altura, nombre});
32
33 bajoPeso = "Bajo Peso";

```

```

32 pesoNormal = "Peso Normal";
33 sobrePeso = "Sobrepeso";
34 filename = 'E:/perez/Documentos/Cursos 2do.
Semestre 2024/PROYECTOS DE COMPUTACION
APLICADA A I.E. Secci n P/Tareas Proyectos/
imc.txt';
35
36
37
38 opcion = 0;
39 opcionu = 0;
40 while opcion ~= 5
41     % Men de Opciones
42     disp('Seleccione una Opci n: ')
43     disp('1. Usuario')
44     disp('2. Calcular IMC y Mostrar Resultados')
45     disp('3. Leer Archivo')
46     disp('4. Borrar Informaci n')
47     disp('5. Salir')
48     opcion = input('Ingrese su Elecci n: ');
49
50 switch opcion
51     case 1
52         try
53             % Men de Usuario
54             while opcionu ~= 4
55                 disp('Seleccione una Opci n: ')
56                 disp('1. Aadir Usuario')
57                 disp('2. Actualizar Usuario')
58                 disp('3. Borrar Usuario')
59                 disp('4. Salir')
60                 opcionu = input('Ingrese la Opci n: ');
61             );
62
63             switch opcionu
64                 case 1
65                     disp('Aadiendo usuario...');
66                     nombre = input('Ingrese el Nombre
del Usuario: ', 's');
67                     altura = input('Ingrese la Altura
del Usuario en CM: ');
68                     peso = input('Ingrese el Peso del
Usuario en Libras: ');
69
70                     insert_query = "INSERT INTO imc (
nombre, peso, altura) VALUES ($1, $2, $3);";
71                     pq_exec_params(conn, insert_query,
{nombre, peso, altura});
72                     disp('Usuario a adido...');
73
74                 case 2
75                     nombre = input('Ingrese el Nombre
del Usuario: ', 's');
76                     altura = input('Actualizar Altura
del Usuario en CM: ');
77                     peso = input('Actualizar el Peso
del Usuario en Libras: ');
78                     update_query = "UPDATE imc SET
peso = $1, altura = $2 WHERE nombre = $3;";
79                     pq_exec_params(conn, update_query,
{peso, altura, nombre});
80                     disp('Usuario actualizado...');

```

* Proyectos

** g-mail: 3741958620101@ingenieria.usac.edu.gt

```

81         case 3
82             nombre = input('Ingrese el Nombre
del Usuario a Borrar: ', 's');
83             delete_query = "DELETE FROM imc
WHERE nombre = $1;";
84             pq_exec_params(conn, delete_query,
{nombre});
85             disp('Usuario borrado...');
86
87         case 4
88             disp('Saliendo');
89
90         otherwise
91             disp('Opci n no v lida.');
```

```

92     end
93 end
94 catch
95     disp('Error al gestionar el usuario');
96 end
97 case 2
98     try
99         nombre_usuario = input('Ingrese el
Nombre: ', 's');
100         query = "SELECT peso, altura FROM imc
WHERE nombre = $1;";
101         result = pq_exec_params(conn, query, {
nombre_usuario});
102
103         % Mensajes de depuraci n
104         disp('Resultado de la consulta:');
105         disp(result);
106         disp(['Tipo de resultado: ', class(
result)]);
107
108         if isempty(result) || isempty(result.
data)
109             disp('Usuario no encontrado o datos
vac os');
110         else
111             % Extraer los valores de peso y altura
de la estructura
112             try
113                 peso = result.data{1,1}; % El
primer valor en la primera fila
114                 altura = result.data{1,2}; % El
segundo valor en la primera fila
115
116                 if peso < 80
117                     estado = bajoPeso;
118                     disp(estado);
119                 elseif peso > 81 && peso < 150
120                     estado = pesoNormal;
121                     disp(estado);
122                 elseif peso > 151
123                     estado = sobrePeso;
124                     disp(estado);
125                 endif
126
127                 disp(['Peso extra do: ', num2str(
peso)]);
128                 disp(['Altura extra da: ', num2str(
altura)]);
129
130                 if isempty(peso) || isempty(altura)
|| peso <= 0 || altura <= 0
131                     disp('Error: Los valores de peso y
altura son inv lidos.');
```

```

132             else
133                 imc = (peso * 0.453592) / ((altura
/ 100) ^ 2);
134
135                 disp(['Peso: ', num2str(peso)]);
136                 disp(['Altura: ', num2str(altura)
]);
137                 disp(['IMC: ', num2str(imc)]);
138
139                 % Abrir archivo para escritura (
vaciar archivo si ya existe)
140                 fileID = fopen(filename, 'w');
141                 if fileID == -1
142                     error('No se pudo abrir el
archivo para escritura.');
```

```

143             end
144             fprintf(fileID, 'Usuario %s\n',
nombre_usuario);
145             fprintf(fileID, 'Peso: %.2f Libras
\n', peso);
146             fprintf(fileID, 'Altura: %.2f CM\n',
altura);
147             fprintf(fileID, 'IMC: %.2f\n', imc
);
148             fprintf(fileID, 'Estado: %s\n',
estado);
149             fclose(fileID);
150         end
151     catch
152         disp('Error al extraer o convertir
los datos');
```

```

153     end
154 end
155 catch
156     disp('Error al calcular y mostrar el IMC
');
```

```

157 case 3
158     try
159         % Leer archivo
160         fileIA = fopen(filename, 'r');
161         if fileIA == -1
162             error('No se pudo abrir el archivo
para lectura.');
```

```

163         endif
164         while ~feof(fileIA)
165             line = fgetl(fileIA);
166             if ischar(line)
167                 disp(line);
168             else
169                 disp('Error al leer una l nea del
archivo.');
```

```

170             endif
171         endwhile
172         fclose(fileIA);
173     catch
174         disp('Error al leer el archivo');
```

```

175     end
176 case 4
177     try
178         % Borrar informaci n del archivo
179         fileID = fopen(filename, 'w');
180         if fileID == -1
181             error('No se pudo abrir el archivo
para escritura.');
```

```

182         end
183         fclose(fileID);
184         disp('Informaci n borrada del archivo.
');
```

```

185     catch
186         disp('Error al borrar la informaci n
del archivo');
```

```

187     end
188 case 5

```

```

189     disp('Saliendo del Programa');
190     disp('Gracias por usar el programa');
191     otherwise
192         disp('Opci n no v lida');
193     end
194 end

```

B. Codigo Python

```

1     import psycopg2
2
3     # Conexi n a la base de datos
4     conn = psycopg2.connect(
5         dbname="imc",
6         user="postgres",
7         password="kuto",
8         host="localhost",
9         port="5432"
10    )
11
12    cursor = conn.cursor()
13
14    pajoPeso = "Bajo Peso"
15    pesoNormal = "Peso Normal"
16    sobrePeso = "Sobrepeso"
17    filename = 'E:/perez/Documentos/Cursos 2do.
18                Semestre 2024/PROYECTOS DE COMPUTACION
19                APLICADA A I.E. Secci n P/Tareas Proyectos/
20                imcp.txt'
21
22    opcion = 0
23    opcionu = 0
24    while opcion != 5:
25        print('Seleccione una Opci n: ')
26        print('1. Usuario')
27        print('2. Calcular IMC y Mostrar Resultados')
28        print('3. Leer Archivo')
29        print('4. Borrar Informaci n')
30        print('5. Salir')
31        opcion = int(input('Ingrese su Elecci n: '))
32
33        if opcion == 1:
34            try:
35                while opcionu != 4:
36                    print('Seleccione una Opci n: ')
37
38                    print('1. Aadir Usuario')
39                    print('2. Actualizar Usuario')
40                    print('3. Borrar Usuario')
41                    print('4. Salir')
42                    opcionu = int(input('Ingrese la
43                    Opci n: '))
44
45                    if opcionu == 1:
46                        print('Aadiendo usuario...')
47
48                        nombre = input('Ingrese el
49                        Nombre del Usuario: ')
50                        altura = float(input('
51                        Ingrese la Altura del Usuario en CM: '))
52                        peso = float(input('Ingrese
53                        el Peso del Usuario en Libras: '))
54
55                        cursor.execute("INSERT INTO
56                        imc (nombre, peso, altura) VALUES (%s, %s, %

```

```

57                        s);", (nombre, peso, altura))
58                        conn.commit()
59                        print('Usuario a adido...')
60
61                    elif opcionu == 2:
62                        nombre = input('Ingrese el
63                        Nombre del Usuario: ')
64                        altura = float(input('
65                        Actualizar Altura del Usuario en CM: '))
66                        peso = float(input('
67                        Actualizar el Peso del Usuario en Libras: '))
68
69                        cursor.execute("UPDATE imc
70                        SET peso = %s, altura = %s WHERE nombre = %s
71                        ;", (peso, altura, nombre))
72                        conn.commit()
73                        print('Usuario actualizado
74                        ...')
75
76                    elif opcionu == 3:
77                        nombre = input('Ingrese el
78                        Nombre del Usuario a Borrar: ')
79
80                        cursor.execute("DELETE FROM
81                        imc WHERE nombre = %s;", (nombre,))
82                        conn.commit()
83                        print('Usuario borrado...')
84
85                    elif opcionu == 4:
86                        print('Saliendo')
87
88                    else:
89                        print('Opci n no v lida.')
90            except Exception as e:
91                print('Error al gestionar el usuario
92                ', e)
93
94        elif opcion == 2:
95            try:
96                nombre_usuario = input('Ingrese el
97                Nombre: ')
98                cursor.execute("SELECT peso, altura
99                FROM imc WHERE nombre = %s;", (
100                nombre_usuario,))
101                result = cursor.fetchone()
102
103                if result is None:
104                    print('Usuario no encontrado o
105                    datos vac os')
106                else:
107                    peso, altura = result
108
109                    if peso < 80:
110                        estado = pajoPeso
111                    elif 81 <= peso <= 150:
112                        estado = pesoNormal
113                    else:
114                        estado = sobrePeso
115
116                    if peso <= 0 or altura <= 0:
117                        print('Error: Los valores de
118                        peso y altura son inv lidos.')
119                    else:
120                        imc = (peso * 0.453592) / ((
121                        altura / 100) ** 2)
122                        print(f'Peso: {peso} libras'
123                        )
124                        print(f'Altura: {altura} cm'
125                        )

```

```

98         print(f'IMC: {imc}')
99         print(f'Estado: {estado}')
100
101         with open(filename, 'w') as
102             file:
103                 file.write(f'Usuario {
104                     nombre_usuario}\n')
105                 file.write(f'Peso: {peso
106                     :.2f} Libras\n')
107                 file.write(f'Altura: {
108                     altura:.2f} CM\n')
109                 file.write(f'IMC: {imc
110                     :.2f}\n')
111                 file.write(f'Estado: {
112                     estado}\n')
113
114         except Exception as e:
115             print('Error al calcular y mostrar
116             el IMC', e)
117
118     elif opcion == 3:
119         try:
120             with open(filename, 'r') as file:
121                 for line in file:
122                     print(line.strip())
123             except Exception as e:
124                 print('Error al leer el archivo', e)
125
126     elif opcion == 4:
127         try:
128             with open(filename, 'w') as file:
129                 pass
130             print('Informaci n borrada del
131             archivo.')
132             except Exception as e:
133                 print('Error al borrar la
134                 informaci n del archivo', e)
135
136     elif opcion == 5:
137         print('Saliendo del Programa')
138         print('Gracias por usar el programa')
139
140     else:
141         print('Opci n no v lida')
142
143 cursor.close()
144 conn.close()

```

III. CONCLUSIONES

- La implementación de funciones para agregar, actualizar y borrar datos en Octave demuestra la capacidad del software para interactuar eficazmente con bases de datos, facilitando la gestión dinámica de información.
- La capacidad de guardar y leer información en archivos de texto desde Octave subraya la flexibilidad del entorno en la gestión de datos persistentes. La escritura en archivos permite almacenar resultados y registros de manera estructurada, mientras que la lectura asegura que estos datos puedan ser recuperados y utilizados para análisis futuros, facilitando así la persistencia y la accesibilidad de la información.

IV. REPOSITORIO GITHUB

