Contenido

1	Pr	ueba de la materialización de los TRIGGERS con la sentencia Eliminar	3
	1.1	Trigger que llama a la función que registra en la Tabla Bitácora Pais	3
	1.2	Eliminacion y materialización del Trigger en la bitácora pais	5
2	Tr	igger que llama a la función que registra en la Tabla Bitácora Provincia	6
	2.1	Eliminacion y materialización del Trigger en la bitácora provincia	7
3	Tr	igger que llama a la función que registra en la Tabla Bitácora Cantón	8
	3.1	Eliminacion y materialización del Trigger en la bitácora Canton	9
4	Tr	igger que llama a la función que registra en la Tabla Bitácora Distrito	10
	4.1	Eliminación y materialización del Trigger en la bitácora Distrito	12
5	Tr	igger que llama a la función que registra en la Tabla Bitácora Barrio	13
	5.1	Eliminación y materialización del Trigger en la bitácora Barrio	14
6	Tr	igger que llama a la función que registra en la Tabla Bitácora Empleado	16
	6.1	Eliminación y materialización del Trigger en la bitácora Empleado	18
7	Tr	igger que llama a la función que registra en la Tabla Bitácora Contacto	19
	7.1	Eliminación y materialización del Trigger en la bitácora Contacto	20
8	Tr	igger que llama a la función que registra en la Tabla Bitácora Guia	22
	8.1	Eliminación y materialización del Trigger en la bitácora Contacto	23
9	Tr	igger que llama a la función que registra en la Tabla Bitácora Duración recorrido	25
	9.1	Eliminación y materialización del Trigger en la bitácora Duración recorrido	26
10)	Trigger que llama a la función que registra en la Tabla Bitácora Itinerarios	28
	10.1	Eliminación y materialización del Trigger en la bitácora Itinerarios	29
1:	L	Trigger que llama a la función que registra en la Tabla Bitácora Asignaciones	31
	11.1	Eliminación y materialización del Trigger en la bitácora Asignaciones	32
12	2	eliminar_continente	34
	12.1	Desplegando tabla continente	35
	12.2	Seleccionando la Bitácora Continente	35
13	3	eliminar_cuidador	36
	13.1	Desplegando tabla cuidadores	37
	13.2	Seleccionando la bitácora cuidadores	37
14	1	eliminar_cuidador_especie	38
	14.1	Desplegando tabla cuidador especie	39

	<i>14.2</i> (elin	? Seleccionando la bitácora cuidador especie, se puede ver como en la última acción ninar) fue guardado en la bitácora. Y esta acción es representada con el carácter D	40
15	5	eliminar_descripcion_especie	. 41
	15.1	Desplegando la tabla descripcion_especie	42
	15.2	Seleccionando la bitácora descripción especie	42
16	5	eliminar_especie	43
	16.1	Desplegando la tabla especie	. 44
	16.2 Dele	Desplegando bitácora especie, en la última acción se puede observar el carácter D parete, luego de haber eliminado una entrada de la tabla especie	
17	7	eliminar_especie_habitat	46
	17.1	Desplegando la tabla	47
	17.2	Desplegando bitácora especie_habitat	47
18	3	eliminar_habitat	. 48
	18.1	Desplegando la tabla habitat	49
			49
	18.2	Desplegando la bitacora habitat	49
19	9	eliminar_lista_dezonas_visitadas	50
	19.1	Desplegando la tabla lista de zonas visitadas	51
	19.2	Desplegando la bitácora lista de zonas visitadas	52
2()	eliminar_tipo_habitat	53
	20.1	Desplegando la tabla tipo hábitat	. 54
	20.2	Desplegando la bitácora tipo habitat	. 54
2:	L	eliminar_zona	55
	21.1	Desplegando la tabla zona	56
	21.2	Desplegando la bitácora zona	57

- 1 Prueba de la materialización de los TRIGGERS con la sentencia Eliminar
 - 1.1 Trigger que llama a la función que registra en la Tabla Bitácora Pais

```
CREATE FUNCTION ControlEliminarPais()
returns TRIGGER
LANGUAGE PLPGSQL
AS $$

DECLARE

vusuario varchar(15);
vid_pais integer;
vnombre character(30);

BEGIN
vid_pais = OLD.id_pais;
vnombre=OLD.nombre;

SELECT user INTO vusuario;
```

```
INSERT INTO public."BitacoraPais"
-- 'D' Hace referencia a la sentencia DELETE
VALUES (vid_pais,vnombre,0,' ',vusuario,'D',now());

RETURN NEW;
END

$$;

CREATE TRIGGER Trigger_TablaPaisELIMINAR
AFTER DELETE
ON public.pais
--CALL EL TRIGGER
FOR EACH ROW EXECUTE PROCEDURE ControlEliminarPais();
```

1.2 Eliminacion y materialización del Trigger en la bitácora pais

id_pais_old_ integer	nombre_old character varying (30)	id_pais_new_ integer	nombre_new character varying (30)	usuario character varying (15)	accion character (1)	fechaRegistro date
1	USA	1	Estados Unidos	postgres	A	2021-06-23
52	Mexico	0		postgres	D	2021-06-24
55	Brasil	0		postgres	D	2021-06-24

2 Trigger que llama a la función que registra en la Tabla Bitácora Provincia

```
CREATE FUNCTION ControlEliminarProvincia()
returns TRIGGER
LANGUAGE PLPGSQL
AS $$
DECLARE
vusuario varchar(15);
vid_provincia integer;
vnombre character(30);
vid_pais integer;
vid_provincia =OLD.id_provincia;
vnombre=OLD.nombre;
vid_pais=OLD.id_pais;
SELECT user INTO vusuario;
 INSERT INTO public. "BitacoraProvincia"
 -- 'D' Hace referencia a la sentencia DELETE
 VALUES (vid_provincia, vnombre, vid_pais, 0 , '', 0, vusuario, 'D', now());
 RETURN NEW;
 END
 $$;
 CREATE TRIGGER Trigger_TablaProvinciaELIMINAR
 AFTER DELETE
```

```
ON public.provincia
--CALL EL TRIGGER

FOR EACH ROW EXECUTE PROCEDURE ControlEliminarProvincia();
```

2.1 Eliminacion y materialización del Trigger en la bitácora provincia

```
DELETE FROM public.provincia
WHERE id_provincia = '8';

SELECT id_provincia, nombre, id_pais
```

FROM public.provincia;

id_provincia_old_ integer	nombre_old character varying (30)	id_pais_old_ integer	id_provincia_new_n. integer		id_pais_new_ integer	usuario character varying (15)	accion character (1)	fechaRegistro date
1	San Jose	506	1	Chepe	506	postgres	A	2021-06-23
8	Limon	506	0		0	postgres	D	2021-06-24

3 Trigger que llama a la función que registra en la Tabla Bitácora Cantón

```
CREATE FUNCTION ControlEliminarCanton()
returns TRIGGER
LANGUAGE PLPGSQL
AS $$
DECLARE
vusuario varchar(15);
vid_canton integer;
vnombre character(30);
vid_provincia integer;
vid_pais integer;
BEGIN
vid_canton = OLD.id_canton;
vnombre=OLD.nombre;
vid_provincia = OLD.id_provincia;
vid_pais=OLD.id_pais;
SELECT user INTO vusuario;
```

```
INSERT INTO public."BitacoraCanton"

VALUES (vid_canton,vnombre,vid_provincia,vid_pais,0 ,'',0,0,vusuario,'D',now());
RETURN NEW;
END

$$;

CREATE TRIGGER Trigger_TablaCantonELIMINAR
AFTER DELETE

ON public.canton
--CALL EL TRIGGER
FOR EACH ROW EXECUTE PROCEDURE ControlEliminarCanton();
```

3.1 Eliminacion y materialización del Trigger en la bitácora Canton

```
DELETE FROM public.canton
WHERE id_canton= '3';
```

DELETE FROM public.canton

WHERE id_canton = '4';

id_canton_old integer	nombre_old character varying (30)	id_provincia_old integer	id_canton_new_ integer	nombre_new character varying (30)	id_provincia_new_ integer	usuario character varying (15	accion character (1)	fechaRegist date
1	San Jose	1	1	Sabana	1	postgres	A	2021-06-23
3	Paxton	3	0		0	postgres	D	2021-06-24
4	Liva	4	0		0	postgres	D	2021-06-24

4 Trigger que llama a la función que registra en la Tabla Bitácora Distrito

```
CREATE FUNCTION ControlEliminarDistrito()
returns TRIGGER
LANGUAGE PLPGSQL
AS $$
DECLARE
vid_pais integer;
vusuario varchar(15);
vid_distrito integer;
vid_canton integer;
vnombre character(30);
vid_provincia integer;
BEGIN
vid_canton = OLD.id_canton;
vnombre=OLD.nombre;
vid_provincia = OLD.id_provincia;
vid_distrito=OLD.id_distrito;
```

```
vid_pais =OLD.id_pais;
SELECT user INTO vusuario;
INSERT INTO public."BitacoraDistrito"

VALUES (vid_distrito,vnombre,vid_canton ,vid_provincia,vid_pais,0,' ',0,0,0,vusuario,'I',now());
RETURN NEW;
END
$$;
CREATE TRIGGER Trigger_TablaDistritoELIMINAR
AFTER DELETE
ON public.distrito
```

4.1 Eliminación y materialización del Trigger en la bitácora Distrito

	id_distrito_old_ integer	nombre_old character varying (30)	id_canton_old integer	id_provincia_old_ integer	id_distrito_new_integer	nombre_new character varying (30)		id_provincia_new_integer	usuario character varying (15)	accion character (1)	fechaRegistron date
1	5	Ambrosius	5	5	0		0	0	postgres		2021-06-24

5 Trigger que llama a la función que registra en la Tabla Bitácora Barrio

```
CREATE FUNCTION ControlEliminarBarrio()
returns TRIGGER
LANGUAGE PLPGSQL
AS $$
DECLARE
vusuario varchar(15);
vid_distrito integer;
vid_canton integer;
vnombre character(30);
vid_provincia integer;
vid_barrio integer;
vid_pais integer;
BEGIN
vid_canton = OLD.id_canton;
vnombre=OLD.nombre;
vid_provincia = OLD.id_provincia;
vid_distrito=OLD.id_distrito;
```

```
vid_distrito=OLD.id_barrio;
vid_barrio=OLD.id_barrio;
vid_pais=OLD.id_pais;
SELECT user INTO vusuario;

INSERT INTO public."BitacoraBarrio"

VALUES (vid_barrio,vnombre,vid_canton ,vid_distrito,vid_provincia,vid_pais,0,' ',0,0,0,0,vusuario,'D',now());
RETURN NEW;
END

$$;
CREATE TRIGGER Trigger_TablaBarrioELIMINAR
AFTER DELETE
ON public.barrio
```

5.1 Eliminación y materialización del Trigger en la bitácora Barrio

```
DELETE FROM public.barrio
WHERE id_barrio = '2';

SELECT id_barrio, nombre, id_distrito, id_canton, id_provincia, id_pais
FROM public.barrio;
```

id_barrio_old integer	nombre_old character varying (30)	id_distrito_old integer	id_canton_old integer	id_provincia_old integer	id_barrio_new integer	nombre_new character varying (30)	id_distrito_new integer	id_canton_new integer	id_provincia_new integer
500	La plata	100	56	7	500	BUEN PASO	100	56	
2	Cañas	2	2	2	0		0	0	

<u></u>	usuario character varying (15)	accion character (1)	fechaRegistro ate
7	postgres	A	2021-06-23
0	postgres	D	2021-06-24

6 Trigger que llama a la función que registra en la Tabla Bitácora Empleado

```
CREATE FUNCTION ControlEliminarEmpleado()
returns TRIGGER
LANGUAGE PLPGSQL
AS $$
DECLARE
vid_empleado integer;
vprimer_nombre character(30);
vsegundo_nombre character(30);
vprimer_apellido character(30);
vsegundo_apellido character(30);
vusuario varchar(15);
vid_distrito integer;
vid_canton integer;
vid_provincia integer;
vid_barrio integer;
vid_pais integer;
```

```
BEGIN

vid_empleado=OLD.id_empleado;

vprimer_nombre=OLD.primer_nombre;

vsegundo_nombre =OLD.segundo_nombre;

vprimer_apellido =OLD.primer_apellido;

vsegundo_apellido=OLD.segundo_apellido;

vid_canton =OLD.id_canton;

vid_provincia =OLD.id_provincia;

vid_distrito=OLD.id_distrito;

vid_barrio=OLD.id_barrio;

vid_pais=OLD.id_pais;

SELECT user INTO vusuario;

INSERT INTO public."BitacoraEmpleados"
```

```
VALUES (vid_empleado,vprimer_nombre,vsegundo_nombre,vprimer_apellido,vsegundo_apellido,vid_provincia,vid_canton ,vid_distrito,v
```

```
, vid\_barrio, vid\_pais, 0, ' ', ' ', ' ', ' ', 0, 0, 0, 0, 0, vusuario, 'D', now());
```

```
RETURN NEW;
```

END

\$\$;

CREATE TRIGGER Trigger_TablaEmpleadoEliminar
AFTER DELETE

ON public.empleados
--CALL EL TRIGGER

FOR EACH ROW EXECUTE PROCEDURE ControlEliminarEmpleado();

6.1 Eliminación y materialización del Trigger en la bitácora Empleado

DELETE FROM public.empleados WHERE id_empleado = '1234';

2077	dorian	[null]	Rodriguez	fdfsdf	7	56	100	500	
2077	dorian	[null]	sjdfbsjfbsdf	fdfsdf	7	56	100	500	
2077	dorian	[null]	Rodriguez	fdfsdf	7	56	100	500	
2077	dorian	[null]	Rodriguez	fdfsdf	7	56	100	500	
1234	Julian	Daniel	ordonez	Guzman	1	1	1	1	

7 Trigger que llama a la función que registra en la Tabla Bitácora Contacto

```
CREATE FUNCTION ControlEliminarContacto()
returns TRIGGER
LANGUAGE PLPGSQL
AS $$
DECLARE
vusuario varchar(15);
vid_contacto integer;
vid_empleado integer;
vnumero_de_telefono character(30);
vnumero_secundario character(30);
vcorreo_electronico character(60);
vcorreo_electronico_secundario character(60);
BEGIN
vid_contacto = OLD.id_contacto;
vid_empleado =OLD.id_empleado;
vnumero_de_telefono=OLD.numero_de_telefono;
vnumero_secundario =OLD.numero_secundario;
vcorreo_electronico=OLD.correo_electronico;
vcorreo_electronico_secundario=OLD.correo_electronico_secundario;
SELECT user INTO vusuario;
INSERT INTO public."BitacoraContacto"
VALUES (vid_contacto ,vid_empleado ,vnumero_de_telefono,vnumero_secundario,vcorreo_electronico,vcorreo_electronico_secundario,0,0,'','','',
```

vusuario,'D',now());

```
RETURN NEW;

END

$$;

CREATE TRIGGER Trigger_TablaContactoEliminar
AFTER DELETE

ON public.contacto
--CALL EL TRIGGER

FOR EACH ROW EXECUTE PROCEDURE ControlEliminarContacto();
--trigger de insercion de la tabla guia
```

7.1 Eliminación y materialización del Trigger en la bitácora Contacto

```
DELETE FROM public.contacto
WHERE id_contacto= '15';
```

4	id_contacto_old integer	id_empleado_old integer	numero_de_telefono_old character (30)	numero_secundario_old character (30)	correo_electronico_old character (60)	correo_electronico_secundario_old character (60)	id_contacto_new integer	id_empleado_new integer
	3	7882	245544	85222	sdfdd@mail.com	sdfsd@mail2.com	3	7{
	0	0					4	5(
	0	0					5	20
	0	0					10	5(
	0	0					15	20
	15	2001	131 387 4436	492-215-0061	jdowzell2@newyorker.com	cpehrsson2@amazon.co.uk	0	

id_empleado_new integer	numero_de_telefono_new character (30)	numero_secundario_new character varying (60)	correo_electronico_new character varying (60)	correo_electronico_secundario_new character (60)	usuario character varying (15)	accion character (1)	fechaRegistro date
7882	1554564565465	85222	sdfsd@mail2.com	sdfsd@mail2.com	postgres	A	2021-06-23
5001	401 964 5175	564-568-9916	knibley3@bloomberg.com	hziemen3@ebay.co.uk	postgres	I	2021-06-23
2004	250 223 5071	149-554-0359	prubury4@china.com.cn	syushankin4@hc360.com	postgres	I	2021-06-23
5003	221 346 4883	494-934-3852	tduckham1@ustream.tv	bshuttlewood1@mayoclinic.com	postgres	I	2021-06-23
2001	131 387 4436	492-215-0061	jdowzell2@newyorker.com	cpehrsson2@amazon.co.uk	postgres	I	2021-06-23
0					postgres	D	2021-06-24

8 Trigger que llama a la función que registra en la Tabla Bitácora Guia

```
CREATE FUNCTION ControlEliminarGuia()
returns TRIGGER
LANGUAGE PLPGSQL
AS $$

DECLARE

vusuario varchar(15);
vid_guia integer;
vid_empleado integer;
vfecha_deingreso DATE;

BEGIN
vid_guia = OLD.id_guia;
vid_empleado = OLD.id_empleado;
vfecha_deingreso=OLD.fecha_deingreso_laboral;

SELECT user INTO vusuario;
```

```
VALUES (vid_guia,vid_empleado,vfecha_deingreso,0,0,null,vusuario,'D',now());

RETURN NEW;
END

$$;

CREATE TRIGGER Trigger_TablaGuiaEliminar
AFTER DELETE

ON public.guia
--CALL EL TRIGGER
FOR EACH ROW EXECUTE PROCEDURE ControlEliminarGuia();
```

8.1 Eliminación y materialización del Trigger en la bitácora Contacto

```
DELETE FROM public.guia
WHERE id_guia= '4' ;
```

	id_guia_old integer △	id_empleado_old integer	fecha_deingreso_laboral_old date	id_guia_new integer	id_empleado_new integer	fecha_deingreso_laboral_new date	usuario character varying (15)	accion character (1)	fechaRegistro date
1	900	2077	2015-07-23	900	2077	2005-07-23	postgres	A	2021-06-23
2	0	0	[null]	1	5030	2020-07-09	postgres		2021-06-23
3	0	0	[null]	2	2001	2020-08-19	postgres		2021-06-23
4	0	0	[null]	4	5001	2021-02-04	postgres		2021-06-23
5	4	5001	2021-02-04	0	0	[null]	postgres	D	2021-06-24

9 Trigger que llama a la función que registra en la Tabla Bitácora Duración recorrido

```
CREATE FUNCTION ControlEliminarDuracionRecorrido()
returns TRIGGER
LANGUAGE PLPGSQL
AS $$

DECLARE

vusuario varchar(15);
vid_duracion_recorrido integer;
vhora integer;
vminutos integer;
vsegundos integer;
vsegundos integer;

BEGIN
vid_duracion_recorrido =OLD.id_duracion_recorrido;
vhora =OLD.hora;
```

```
vminutos=OLD.minutos;
vsegundos=OLD.segundos;

SELECT user INTO vusuario;

INSERT INTO public."BitacoraDuracion_recorrido"

VALUES (vid_duracion_recorrido,vhora,vminutos,vsegundos,0,0,0,vusuario,'D',now());

RETURN NEW;
END

$$;

CREATE TRIGGER Trigger_TablaDuracion_Recorrido_Eliminar
AFTER DELETE
```

FOR EACH ROW EXECUTE PROCEDURE ControlEliminarDuracionRecorrido();

9.1 Eliminación y materialización del Trigger en la bitácora Duración recorrido

id_duracion_recorrido_old integer	hora_old integer	•	minutos_old integer	segundos_old integer	id_duracion_recorrido_new integer		hora_new integer		minutos_new integer	segundos_new integer	ш.	usuario character varying (15)	accion character (1)	
90		1	0	0		90	1	١	30		0	postgres	A	
0		0	0	0		1	11	١	1		0	postgres	I	
0		0	0	0		2	5	5	14		0	postgres	I	
0		0	0	0		3	5	5	47		0	postgres	I	
3		5	47	0		0	0)	0		0	postgres	D	

fechaRegistro date	_
2021-06-23	
2021-06-23	
2021-06-23	
2021-06-23	
2021-06-24	

10Trigger que llama a la función que registra en la Tabla Bitácora <u>Itinerarios</u>

```
CREATE FUNCTION ControlEliminarItinerarios()
returns TRIGGER
LANGUAGE PLPGSQL
AS $$

DECLARE

vusuario varchar(15);
vcodigo varchar(30);
vid_duracion_recorrido integer;
vlongitud_recorrido double precision;
vnumero_maximodevisitante_autorizados integer;
vnumero_dedistintasespecies_quevisita integer;
vsegundos integer;
```

BEGIN

```
RETURN NEW;
END

$$;

CREATE TRIGGER Trigger_TablaItinerarios_Eliminar
AFTER DELETE

ON public.itinerarios
--CALL EL TRIGGER

FOR EACH ROW EXECUTE PROCEDURE ControlEliminarItinerarios();
```

10.1 Eliminación y materialización del Trigger en la bitácora Itinerarios

```
DELETE FROM public.itinerarios
WHERE codigo= 'GH30' ;
```

codigo_old character varying (30)	id_duracion_recorrido_old integer	longitud_recorrido_old_ double precision	${\color{red} \textbf{numero_maximodevisitante_autorizados_old_integer}}$	numero_dedistintasespecies_quevisita_old_integer
3000	90	100.31	10	12
0	0	0	0	(
0	0	0	0	(
0	0	0	0	(
0	0	0	0	(
GH30	1	300.4	33	32
GH30	1	300.4	33	33

usuario character varying (15	accion character (1)	fechaRegistro date
postgres	A	2021-06-23
postgres	1	2021-06-23
postgres	1	2021-06-23
postgres	I	2021-06-23
postgres	I	2021-06-23
postgres	I	2021-06-24
postgres	D	2021-06-24

11 Trigger que llama a la función que registra en la Tabla Bitácora Asignaciones

```
CREATE FUNCTION ControlEliminarAsignaciones()
returns TRIGGER
LANGUAGE PLPGSQL
AS $$

DECLARE

vusuario varchar(15);
vid_asignaciones integer;
vid_guia integer;

vhora_comienzo TIME;
vcodigo varchar(30);

BEGIN
```

```
vid_guia =OLD.id_guia;
vhora_comienzo =OLD.hora_comienzo;
vcodigo =OLD.codigo ;

SELECT user INTO vusuario;

INSERT INTO public."BitacoraAsignaciones"

VALUES (vid_asignaciones,vhora_comienzo ,vid_guia,vcodigo,0,null,0,'',vusuario,'D|',now());
RETURN NEW;
END

$$;
```

```
CREATE TRIGGER Trigger_TablaEliminarAsignaciones

AFTER DELETE

ON public.asignaciones

--CALL EL TRIGGER

FOR EACH ROW EXECUTE PROCEDURE ControlEliminarAsignaciones();

--fin del trigger de control de eliminacion de asignaciones
```

11.1 Eliminación y materialización del Trigger en la bitácora Asignaciones

```
DELETE FROM public.asignaciones
WHERE id_asignaciones= '10';
```

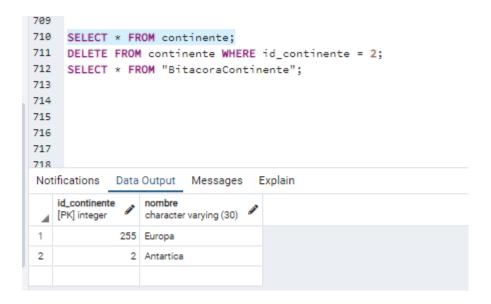
id_asignaciones_old integer	hora_comienzo_old time without time zone	id_guia_old integer	codigo_old character varying (30)	id_asignaciones_new integer	hora_comienzo_new time without time zone	id_guia_new integer	codigo_new character varying (30)	usuario character varying
1	07:00:00	900	3000	1	07:30:00	900	3000	postgres
0	[null]	0		4	11:23:02	1	GH22	postgres
0	[null]	0		5	10:52:01	2	BH26	postgres
0	[null]	0		10	10:50:01	2	BH26	postgres
10	10:50:01	2	BH26	0	[null]	0		postgres

accion character (1)	fechaRegistro date
_	2021-06-23
The second secon	2021-06-23
1	2021-06-23
1	2021-06-23
D	2021-06-24

12 eliminar_continente

```
26
  27 -- insertar continente
  28 CREATE FUNCTION insertar_continente() RETURNS TRIGGER AS
  29 $$
  30
          DECLARE
  31
  32
            accion character(1) := 'I';
  33
             usuario varchar(30) := (select user);
  34
             fecha_registro date := now();
  35
             id_nulo integer := 0;
  36
             nombre_nulo character(1) := ' ';
  37
  38
         BEGIN
  39
  40
             INSERT INTO public. "BitacoraContinente"
  41
                 VALUES (id_nulo, nombre_nulo,
  42
  43
                                 NEW.id_continente, NEW.nombre,
  44
  45
                                 usuario, accion, fecha_registro);
  46
  47
              RETURN NEW;
  48
  49
          END
  50 $$
  51 LANGUAGE PLPGSQL;
```

12.1 Desplegando tabla continente



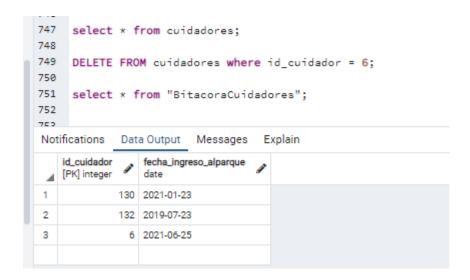
12.2 Seleccionando la Bitácora Continente



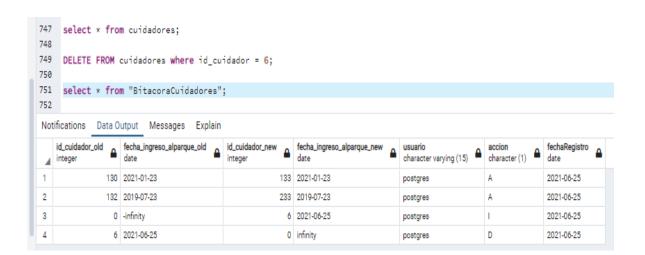
13 eliminar_cuidador

```
1/2
173 -- eliminar cuidador
174 CREATE FUNCTION eliminar_cuidador() RETURNS TRIGGER AS
175 $$
176
         DECLARE
177
178
            accion character(1) := 'D';
179
            usuario varchar(30) := (select user);
180
            fecha_registro date := now();
181
            id_nulo integer := 0;
182
             fecha_de_ingreso_alparque_nulo date := 'infinity';
183
184
185
186
         BEGIN
187
             INSERT INTO public. "BitacoraCuidadores"
188
189
                 VALUES (OLD.id_cuidador, OLD.fecha_ingreso_alparque,
190
191
                                id_nulo, fecha_de_ingreso_alparque_nulo,
192
193
                                usuario, accion, fecha_registro);
194
195
             RETURN NEW;
196
197
         END
198 $$
199 LANGUAGE PLPGSQL;
```

13.1 Desplegando tabla cuidadores



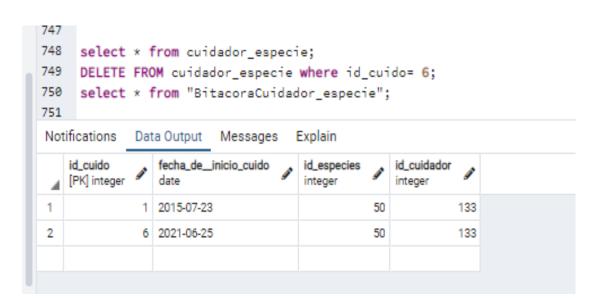
13.2 Seleccionando la bitácora cuidadores



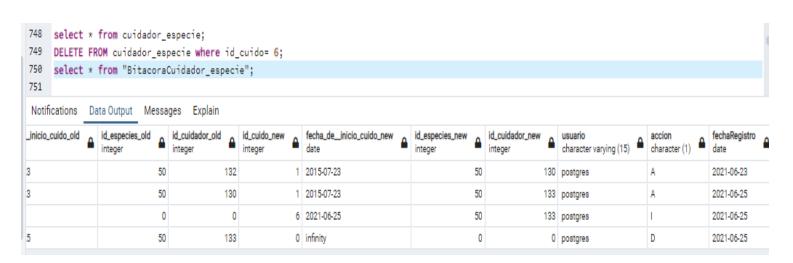
14 eliminar_cuidador_especie

```
-- eliminar cuidador especie
110 CREATE FUNCTION eliminar_cuidador_especie() RETURNS TRIGGER AS
111 $$
112
         DECLARE
113
114
             accion character(1) := 'D';
115
             usuario varchar(30) := (select user);
116
             fecha_registro date := now();
117
             id_nulo integer := θ;
118
             fecha_de_inicio_cuido_nulo date := 'infinity';
119
             id_especies_nulo integer := 0;
120
             id_cuidador_nulo integer := 0;
121
122
123
         BEGIN
124
125
             INSERT INTO public. "BitacoraCuidador_especie"
126
                 VALUES (OLD.id_cuido, OLD.fecha_de__inicio_cuido,
127
                                 OLD.id_especies, OLD.id_cuidador ,
128
129
                                 id_nulo, fecha_de_inicio_cuido_nulo,
130
                                 id_especies_nulo, id_cuidador_nulo,
131
132
                                 usuario, accion, fecha_registro);
133
134
             RETURN NEW:
135
136
         END
137
138 LANGUAGE PLPGSQL;
```

14.1 Desplegando tabla cuidador_especie



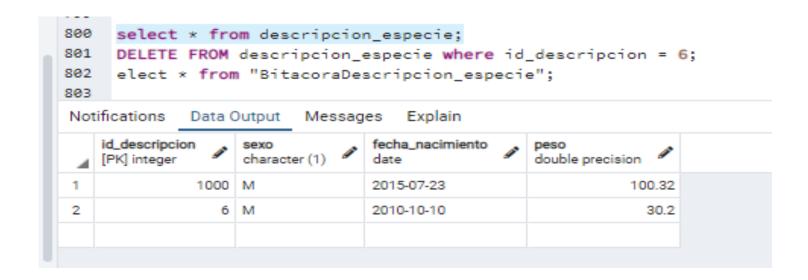
14.2 Seleccionando la bitácora cuidador especie, se puede ver como en la última acción (eliminar) fue guardado en la bitácora. Y esta acción es representada con el carácter *D*



15 eliminar_descripcion_especie

```
231 -- eliminar descripcion especie
232 CREATE FUNCTION eliminar_descripcion_especie() RETURNS TRIGGER AS
233 $$
234
         DECLARE
235
236
            accion character(1) := 'D';
237
            usuario varchar(30) := (select user);
            fecha_registro date := now();
238
239
            id_nulo integer := Θ;
240
            sexo_nulo character(1) := ' ';
            fecha_de_nacimiento_nulo date := 'infinity';
241
242
           peso_nulo double precision := 0.0;
243
244
245
246
         BEGIN
247
248
             INSERT INTO public. "BitacoraDescripcion_especie"
249
                VALUES (OLD.id_descripcion, OLD.sexo, OLD.fecha_nacimiento, OLD.peso,
250
251
                                id_nulo, sexo_nulo, fecha_de_nacimiento_nulo, peso_nulo,
252
253
                                usuario, accion, fecha_registro);
254
255
             RETURN NEW;
256
257
258 $$
259 LANGUAGE PLPGSQL;
```

15.1 Desplegando la tabla descripcion_especie



15.2 Seleccionando la bitácora descripción especie

	FROM descripcion_e								
802 selec	t * from "BitacoraDe	scripcion_espec	ie";						
803									
Notifications	Data Output Message	es Explain							
	fache projectorie ald	peso_old	id_descripcion_new _	sexo_new o	fecha_nacimiento_new	peso_new o	usuario	accion	fechaRegistr
	fecha_nacimiento_old date	double precision	integer	character (1)	date	double precision	character varying (15)	character (1)	date
character (1)	•			character (1)	date 2015-07-23	double precision	character varying (15)	character (1)	date 2021-06-23
sexo_old character (1) M	date	double precision	integer 1000	character (1)	date	double precision 100.32	character varying (15)	character (1)	

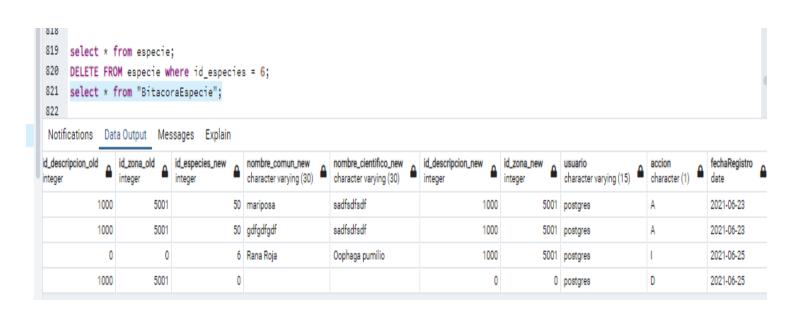
16 eliminar_especie

```
294
    -- eliminar especie
295 CREATE FUNCTION eliminar_especie() RETURNS TRIGGER AS
296 $$
297
         DECLARE
298
299
           accion character(1) := 'D';
300
            usuario varchar(30) := (select user);
            fecha_registro date := now();
301
302
            id_nulo integer := 0;
303
           nombre_comun_nulo character(1) := ' ';
304
            nombre_cientifico_nulo character(1) := ' ';
            id_descripcion_nulo integer := 0;
306
            id_zona_nulo integer := 0;
307
308
309
310
         BEGIN
311
312
             INSERT INTO public. "BitacoraEspecie"
313
                 VALUES (OLD.id_especies, OLD.nombre_comun, OLD.nombre_cientifico,
314
                                OLD.id_descripcion, OLD.id_zona,
315
316
                                 id_nulo, nombre_comun_nulo, nombre_cientifico_nulo,
317
                                 id_descripcion_nulo, id_zona_nulo,
318
319
                                usuario, accion, fecha_registro);
320
321
             RETURN NEW;
322
323
         END
324 $$
325 LANGUAGE PLPGSQL;
```

16.1 Desplegando la tabla especie

```
OTO
      select * from especie;
819
820 -- DELETE FROM especie where id_especies = 6;
      -- select * from "BitacoraEspecie";
821
822
               Data Output
                                         Explain
Notifications
                             Messages
    id_especies
                   nombre_comun
                                          nombre_cientifico
                                                                 id_descripcion
                                                                                  id_zona
                   character varying (30)
                                          character varying (30)
                                                                                  integer
    [PK] integer
                                                                 integer
                50 gdfgdfgdf
                                          sadfsdfsdf
 1
                                                                             1000
                                                                                        5001
                                          Oophaga pumilio
 2
                6 Rana Roja
                                                                             1000
                                                                                        5001
```

Desplegando bitácora especie, en la última acción se puede observar el carácter D para *Delete*, luego de haber eliminado una entrada de la tabla especie



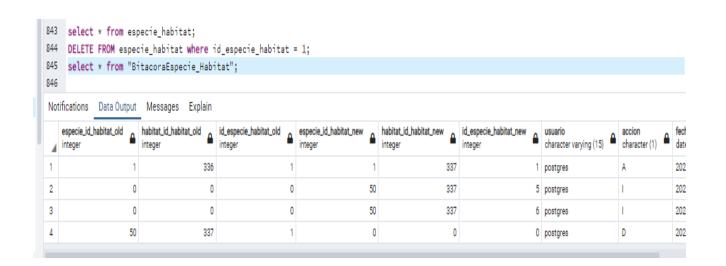
17 eliminar_especie_habitat

```
-- eliminar especie_habitat
362 CREATE FUNCTION eliminar_especie_habitat() RETURNS TRIGGER AS
363 $$
364
        DECLARE
365
366
            accion character(1) := 'D';
367
            usuario varchar(30) := (select user);
368
            fecha_registro date := now();
369
            id_nulo integer := 0;
370
            habitat_id_habitat_nulo integer := 0;
371
             id_especie_habitat_nulo integer := 0;
372
373
374
        BEGIN
375
376
             INSERT INTO public. "BitacoraEspecie_Habitat"
377
                 VALUES (OLD.especie_id_habitat, OLD.habitat_id_habitat, OLD.id_especie_habitat,
378
379
                                 id_nulo, habitat_id_habitat_nulo, id_especie_habitat_nulo,
380
381
                                 usuario, accion, fecha_registro);
382
383
             RETURN NEW;
384
385
         END
386 $$
387
    LANGUAGE PLPGSQL;
```

17.1 Desplegando la tabla

```
842
      select * from especie habitat;
843
      DELETE FROM especie habitat where id especie habitat = 1;
844
      select * from "BitacoraEspecie Habitat";
845
846
Notifications
               Data Output
                             Messages
                                         Explain
                        habitat_id_habitat
                                            id_especie_habitat
    especie_id_habitat
                                            [PK] integer
    integer
                        integer
1
                     50
                                       337
                                                             1
                                                             5
2
                     50
                                       337
3
                     50
                                       337
                                                              6
```

17.2 Desplegando bitácora especie_habitat



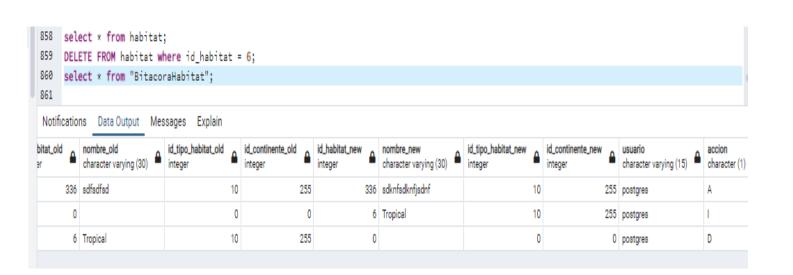
18 eliminar_habitat

```
419
    -- eliminar habitat
420 CREATE FUNCTION eliminar_habitat() RETURNS TRIGGER AS
421
422
         DECLARE
423
424
             accion character(1) := 'D';
425
             usuario varchar(30) := (select user);
426
             fecha_registro date := now();
427
             id_nulo integer := Θ;
428
            nombre_nulo character(1) := ' ';
429
            id_tipo_habitat_nulo integer := 0;
430
             id_continente_nulo integer := 0;
431
432
433
         BEGIN
434
435
             INSERT INTO public. "BitacoraHabitat"
436
                 VALUES (OLD.id_habitat, OLD.nombre, OLD.id_tipo_habitat, OLD.id_continente,
437
438
                                 id_nulo, nombre_nulo, id_tipo_habitat_nulo, id_continente_nulo,
439
440
                                 usuario, accion, fecha_registro);
441
442
             RETURN NEW;
443
444
         END
445 $$
446 LANGUAGE PLPGSQL;
```

18.1 Desplegando la tabla habitat

```
858
      select * from habitat;
859
      DELETE FROM habitat where id_habitat = 6;
      select * from "BitacoraHabitat";
860
861
Notifications
               Data Output
                              Messages
                                           Explain
    id_habitat
                                           id_tipo_habitat
                    nombre
                                                             id_continente
                    character varying (30)
    [PK] integer
                                           integer
                                                             integer
                   sdknfsdknfjsdnf
                                                         10
                                                                         255
1
2
               337
                   sdfsdfsd
                                                         10
                                                                         255
3
                   Tropical
                                                          10
                                                                         255
```

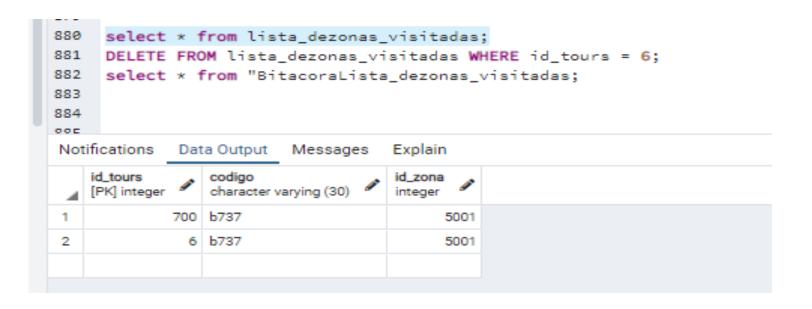
18.2 Desplegando la bitacora habitat



19 eliminar_lista_dezonas_visitadas

```
480
    -- eliminar lista zonas visitadas
481
     CREATE FUNCTION eliminar_lista_dezonas_visitadas() RETURNS TRIGGER AS
482
    $$
483
        DECLARE
484
485
            accion character(1) := 'D';
486
            usuario varchar(30) := (select user);
487
             fecha_registro date := now();
488
             id_nulo integer := Θ;
489
             codigo_nulo varchar(30) := ' ';
490
             id_zonas_nulo integer := Θ;
491
492
493
        BEGIN
494
495
             INSERT INTO public. "BitacoraLista_dezonas_visitadas"
496
                 VALUES (OLD.id_tours, OLD.codigo, OLD.id_zona,
497
498
                                 id_nulo, codigo_nulo, id_zonas_nulo,
499
500
                                 usuario, accion, fecha_registro);
501
502
             RETURN NEW;
503
504
        END
505 $$
506 LANGUAGE PLPGSQL;
507
```

19.1 Desplegando la tabla lista de zonas visitadas



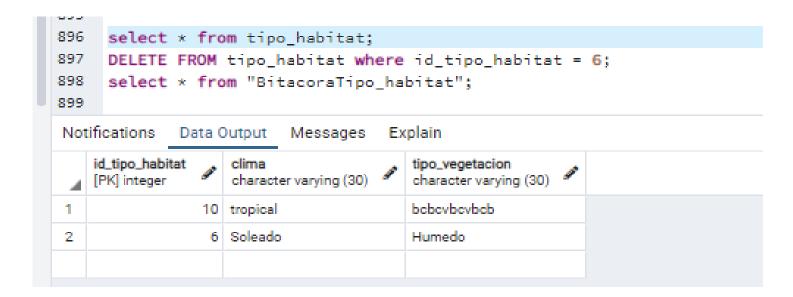
19.2 Desplegando la bitácora lista de zonas visitadas



20 eliminar_tipo_habitat

```
-- eliminar tipo_habitat
540
    CREATE FUNCTION eliminar_tipo_habitat() RETURNS TRIGGER AS
541
542
         DECLARE
543
544
             accion character(1) := 'D';
545
            usuario varchar(30) := (select user);
546
            fecha_registro date := now();
547
             id_nulo integer := θ;
548
             clima_nulo character := ' ';
549
             tipo_vegetacion_nulo character(1) = ' ';
550
551
552
        BEGIN
553
554
             INSERT INTO public. "BitacoraTipo_habitat"
555
                 VALUES (OLD.id_tipo_habitat, OLD.clima, OLD.tipo_vegetacion,
556
557
                                 id_nulo, clima_nulo, tipo_vegetacion_nulo,
558
                                 usuario, accion, fecha_registro);
560
561
             RETURN NEW;
562
         END
563
564 $$
565 LANGUAGE PLPGSQL;
566
```

20.1 Desplegando la tabla tipo hábitat



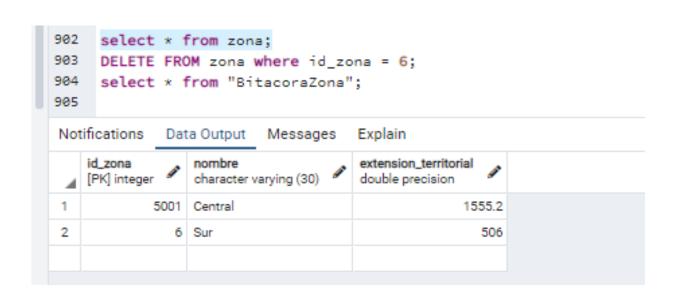
20.2 Desplegando la bitácora tipo habitat



21 eliminar zona

```
-- eliminar zona
   CREATE FUNCTION eliminar_zona() RETURNS TRIGGER AS
56
57
        DECLARE
58
59
          accion character(1) := 'D';
60
           usuario varchar(30) := (select user);
61
           fecha_registro date := now();
62
           id_nulo integer := 0;
63
           nombre_nulo character(1) := ' ';
64
            extension_territorial_nulo double precision := 0.0;
65
66
        BEGIN
67
            INSERT INTO public. "BitacoraZona"
68
69
                VALUES (OLD.id_zona, OLD.nombre, OLD.extension_territorial,
70
71
                                id_nulo, nombre_nulo, extension_territorial_nulo,
72
73
                               usuario, accion, fecha_registro);
74
75
            RETURN NEW;
76
77
        END
78 $$
79 LANGUAGE PLPGSQL;
```

21.1 Desplegando la tabla zona



21.2 Desplegando la bitácora zona

