Obligatorio I

Base de Datos II

Universidad ORT Uruguay – Facultad de Ingeniería



Matías Gutierrez

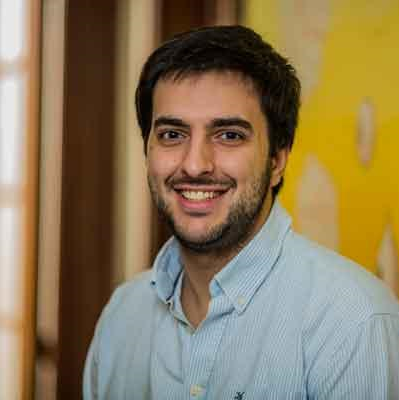
(200244)

Mario Souto

(168772)

Bruno Vezoli

(201150)



Contents

[1. Abstract 2](#_Toc499845173)

[2. Introducción 2](#_Toc499845174)

[3. Diagrama 2](#_Toc499845175)

[4. Análisis de la Solución 2](#_Toc499845176)

[5. Identificación y Clasificación de Restricciones 5](#_Toc499845177)

[6. Datos de prueba Restricciones de Integridad 11](#_Toc499845178)

[7. Explicación de Procedimientos 14](#_Toc499845179)

[7.1 Procedimiento 1 14](#_Toc499845180)

[7.2 Procedimiento 2 16](#_Toc499845181)

[7.3 Procedimiento 3 16](#_Toc499845182)

[7.4 Procedimiento 4 17](#_Toc499845183)

[7.5 Procedimiento 5 19](#_Toc499845184)

[8. Datos de Prueba Procedimientos 22](#_Toc499845185)

# Abstract

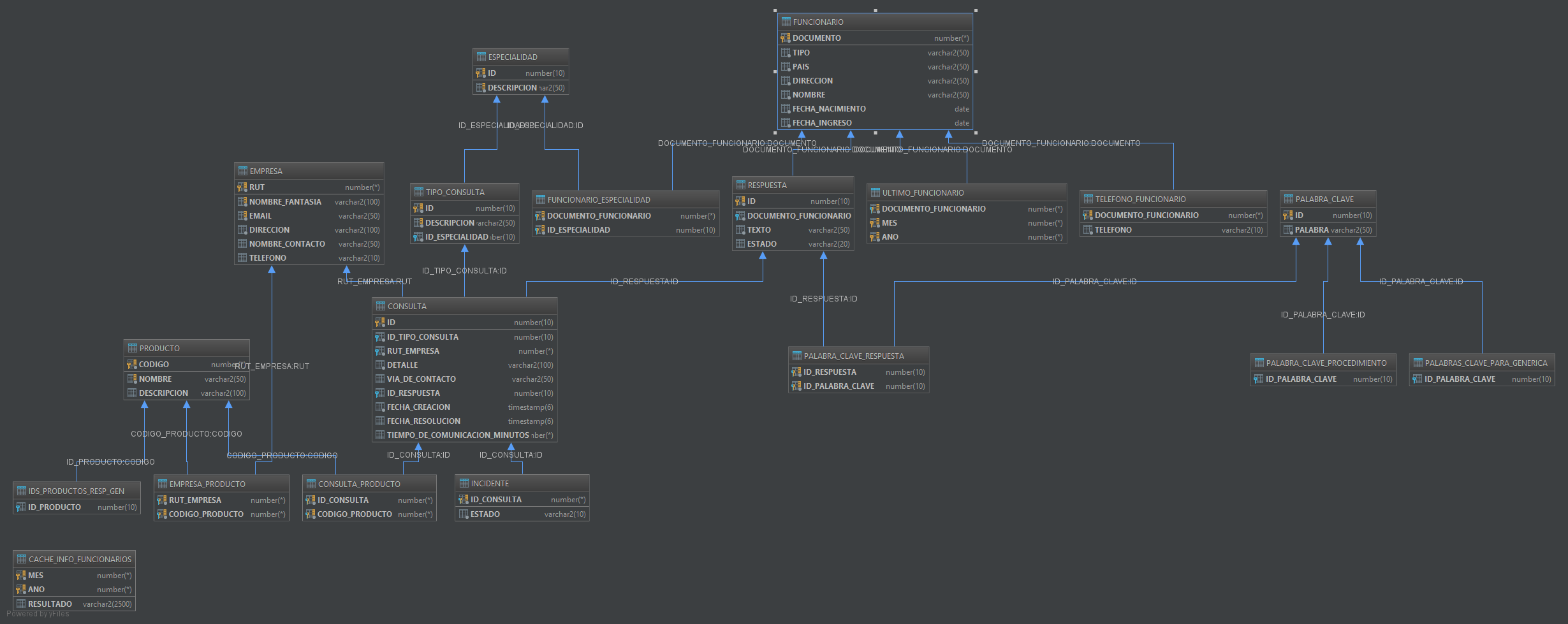
El sistema *Sistema de Gestión al Cliente* es un sistema para la automatización de consultas e incidentes orientado a grandes empresas.

Se relevaron los requerimientos en base a la información provista y se construyó una solución en base de datos del problema utilizando el motor de base de datos Oracle y el lenguaje procedural PL/SQL.

# Introducción

El presente documento contiene la solución al problema presentado incluyendo, el diagrama del Modelo Entidad Relación, el DDL completo de la solución así como el análisis del diseño efectuado, los supuestos realizados y un análisis de las restricciones de integridad detectadas. También se incluyen los casos de prueba con las salidas esperadas por pantalla.

# Diagrama



# Análisis de la Solución

El objetivo de esta sección del documento es el análisis de la solución propuesta y la justificación del diseño escogido.

***Diseño general de la solución***

Del primer párrafo del documento *“Hasta el momento cada uno de los técnicos registra en una planilla electrónica la empresa, nombre de contacto, tiempo que lleva la comunicación, el detalle de la consulta y la respuesta brindada al cliente”*  se desprende que un técnico registra en la planilla electrónica la consulta efectuada por el cliente por lo cual se crea una entidad **CONSULTA** y se incluye en la misma los atributos **detalle** y **tiempo\_de\_comunicacion\_minutos** así como la **via\_de\_contacto** (Teléfonica, por correo electrónico o personal) que es mencionada en la misma sección. También se agregan los atributos **fecha\_creacion** con la fecha de ingreso de la consulta al sistema y **fecha\_resolucion** para cuando la consulta es respondida.

Para las consultas que sean necesarias estudiarlas en profundidad se crea la tabla **INCIDENTE**, con **FOREIGN\_KEY** **id\_consulta** que referencia al id de la consulta, y el estado del mismo, con valores *Pendiente* o *Resuelto*.

Una consulta siempre tiene un tipo de consulta. Asumimos que existirán varias consultas para un mismo tipo de consulta, por lo que creamos una entidad **TIPO\_CONSULTA** con el atributo **descripción** y agregamos una **FOREIGN\_KEY id\_tipo\_consulta** que hace referencia al id del tipo de consulta.

Del párrafo *“Los clientes registrados son empresas, las cuales se identifican por el RUT o nombre fantasía. Además se cuenta con sus datos de contacto que pueden ser por ejemplo dirección, mail y/o número de teléfono.”* se crea una entidad **Empresa** con los atributos **rut**, **nombre\_fantasia**, **email**, **dirección**, **nombre\_contacto** y **teléfono.** Así mismo como una **Consulta** es realizada por un cliente (en este caso los clientes son empresas) se crea una **FOREIGN\_KEY rut\_empresa** en la entidad **Consulta** que referencia al **rut** de la **Empresa**, su clave primaria.

*“La empresa cuenta con varios productos los cuales son identificados por un código, tienen un nombre único y una descripción.”* de esta información se extrae la entidad **Producto** con los atributos **código**, **nombre**, y **descripción.** Suponemos a los productos como algo genérico, es decir, el Producto A puede pertenecer a varias empresas. Un producto no es exclusivo de una sola empresa. Así mismo, por letra, una empresa tiene varios productos por lo que la relación entre **Producto** y **Empresa** es de **N a N**, por lo que creamos la tabla auxiliar **Empresa\_Producto** con **FOREIGN\_KEYS rut\_empresa** y **código\_producto**, que referencian a las claves primarias de empresa y producto respectivamente**.**

De igual forma, los clientes realizan consultas sobre los productos. Por lo que una consulta puede tener varios productos y un producto pertenecer a varias consultas. Para ello creamos la tabla auxiliar **Consulta\_Producto**, que representa la relación de **N a N**, con las **FOREIGN\_KEYS id\_consulta y código\_producto**, que referencian a las claves primarias de consulta y producto respectivamente.

Del párrafo *“De los funcionarios se conoce su nombre, dirección y algún teléfono, pero se desea contar con más información por lo que se requiere además registrar país, tipo y número de documento, nombre completo, fecha de nacimiento, fecha de ingreso a la empresa y sus especialidades. Las especialidades definidas actualmente son: técnica, de investigación, funcional. No se descarta que más adelante puedan surgir otras”* se extrae la entidad **Funcionario** con los atributos **documento**, **tipo**, **país**, **dirección**, **nombre**, **fecha\_nacimiento** y **fecha\_ingreso**.

Así mismo se especifica que un funcionario puede tener varias especialidades y que ellas actualmente son *técnica, de investigación y funcional* pero podrían ser más. Para ello, en lugar de utilizar un atributo *especialidad* en esta tabla, creamos la entidad **Especialidad** con atributo **descripcion** y la tabla auxiliar **Funcionario\_Especialidad**  para representar que varios funcionarios pueden compartir especialidad y que un funcionario puede tener más de una. Los atributos **Documento\_Funcionario** e **Id\_Especialidad** son **FOREIGN\_KEYS** que referencian a las claves de las tablas **Funcionario** y **Especialidad** respectivamente.

Dado que el funcionario tiene “algún” teléfono, pudiendo ser estos uno o más, se crea la tabla **TELEFONO\_FUNCIONARIO** con el atributo **teléfono** y **FOREIGN\_KEY** **documento\_funcionario** que referencia a la clave primaria de **Funcionario**.

Cada consulta ingresada deberá, eventualmente, tener una respuesta. Por lo que se crea la entidad **Respuesta** con atributos **documento\_funcionario**, **texto** y **estado.** Siendo **documento\_funcionario** una **FOREIGN\_KEY** que referencia a la entidad **Funcionario** y que en este caso representa al funcionario que ingresó la respuesta en el sistema. El atributo **texto** representa la respuesta ofrecida al cliente. Por último, la tabla **Respuesta** almacena tanto respuestas *simples* como respuestas *generalizables*. La diferencia entre ambas será que las respuestas *simples* no tienen **estado**, es decir, este atributo tendrá valor **NULL**, dado que las respuestas simples no tienen que ser auditadas por otro funcionario. En el caso de las respuestas *generalizables* este atributo tendrá los valores *Pendiente de revisión, Revisada Ok o Rechazada,* según como haya sido la auditoría sobre esta respuesta.

Se especifica en la letra que los tipos de consulta están relacionados con las especialidades de los funcionarios, de forma de que ciertos tipos de consulta sean más eficientemente atendidos por funcionarios especializados. Para ello, en la tabla **TipoConsulta**, se agrega la **FOREIGN\_KEY id\_especialidad** que referencia a la clave de la tabla **Especialidad**. Por lo tanto, un tipo de consulta puede estar asignado únicamente a una especialidad y una especialidad puede atender varios tipos de consulta.

Una respuesta *generalizable* debe estar asociada a una especialidad particular. Dicho de otra forma, no puede utilizarse la misma respuesta *generalizable* en consultas con tipos de consulta asignados a especialidades diferentes. Esta restricción se controla mediante un *trigger* que será detallado en la siguiente sección.

El sistema permite definir palabras clave, para ello se crea la entidad **PalabraClave** con el atributo **palabra** que almacena la palabra clave deseada.

Por último, a las respuestas se le pueden asignar palabras clave. Consideramos que una respuesta puede tener múltiples palabras clave y que una palabra clave puede ser incluída en múltiples respuestas. Dicho de otra forma la relación es de **N a N.** Para ello se crea la tabla **PALABRA\_CLAVE\_RESPUESTA** con **FOREIGN\_KEYS id\_respuesta** e **id\_palabra\_clave** que referencian a las tablas **Respuesta** y **Palabra\_Clave** respectivamente.

***Supuestos efectuados***

* Se asume que a la hora de efectuar consultas la persona que se contacta en representación del cliente es siempre la misma, en definitiva, un único *nombre\_contacto* por empresa.
* Se asume que los productos son genéricos dentro del sistema y, por lo tanto, pueden relacionarse con varias empresas.

# Identificación y Clasificación de Restricciones

**TABLA CONSULTA**

|  |  |  |
| --- | --- | --- |
| RESTRICCIÓN | TIPO DE RESTRICCIÓN | IMPLEMENTACIÓN |
| ID NOT NULL PRIMARY KEY | **Entidad** | **Estructural** |
| ID\_TIPO\_CONSULTA NOT NULL | **Dominio** | **Estructural** |
| ID\_TIPO\_CONSULTA CONSTRAINT CONST\_TO\_TIPO\_CONS\_FK REFERENCES TIPO\_CONSULTA | **Referencial** | **Estructural** |
| RUT\_EMPRESA NOT NULL | **Dominio** | **Estructural** |
| RUT\_EMPRESA CONSTRAINT CONSULTA\_TO\_EMPRESA\_FK REFERENCES EMPRESA | **Referencial** | **Estructural** |
| DETALLE NOT NULL | **Dominio** | **Estructural** |
| VIA\_DE\_CONTACTO CONSTRAINT CONSULTA\_CONTACTO CHECK (VIA\_DE\_CONTACTO\_IN) | **Dominio** | **Estructural** |
| ID\_RESPUESTA CONSTRAINT CONSULTA\_TO\_RESP\_FK REFERENCES RESPUESTA | **Referencial** | **Estructural** |
| FECHA\_CREACION NOT NULL | **Dominio** | **Estructural** |
| Consulta\_id | **Semántica** | **No Estructural** |
| FECHA\_CREACION\_CONSULTA | **Semántica** | **No Estructural** |
| FECHA\_RESPUESTA\_CONSULTA | **Semántica** | **No Estructural** |
| RESPUESTA\_INCIDENTE | **Semántica** | **No Estructural** |
| RESPUESTA\_ESPECIALIDAD\_UNICA | **Semántica** | **No Estructural** |

**Comentarios**

* El *trigger* **FECHA\_CREACION\_CONSULTA** se ejecuta al momento de ingresar una nueva consulta y, setea la fecha de creación de la consulta como la fecha actual del sistema.
* El *trigger* **FECHA\_RESPUESTA\_CONSULTA** se ejecuta al momento de actualizar una consulta, y setea la fecha de resolución de una consulta como la fecha actual del sistema siempre que a la misma se le esté asignando una respuesta.
* El *trigger* **RESPUESTA\_INCIDENTE** se ejecuta al momento de actualizar una consulta y, si la misma tiene respuesta, se marca la incidencia relacionada como resuelta.
* El *trigger* **RESPUESTA\_ESPECIALIDAD\_UNICA** se ejecuta al momento de crear o actualizar una consulta y valida que una respuesta *genérica* solamente pueda estar relacionada con una *especialidad.* Por lo que si se ingresa o actualiza una consulta relacionada y se le asigna respuesta *genérica* cuyo tipo de consulta tiene una especialidad distinta a la original se producirá un error.

**TABLA CONSULTA\_PRODUCTO**

|  |  |  |
| --- | --- | --- |
| RESTRICCIÓN | TIPO DE RESTRICCIÓN | IMPLEMENTACIÓN |
| ID\_CONSULTA NOT NULL | **Dominio** | **Estructural** |
| CODIGO\_PRODUCTO NOT NULL | **Dominio** | **Estructural** |
| ID\_CONSULTA CONSTRAINT CONS\_PROD\_TO\_CONS\_FK REFERENCES CONSULTA | **Referencial** | **Estructural** |
| CODIGO\_PRODUCTO CONSTRAINT CONS\_PROD\_TO\_PROD\_FK REFERENCES PRODUCTO | **Referencial** | **Estructural** |
| CONSTRAINT CONSULTA\_PRODUCTO\_PK PRIMARY\_KEY (ID\_CONSULTA, CODIGO\_PRODUCTO) | **Entidad** | **Estructural** |

**TABLA EMPRESA**

|  |  |  |
| --- | --- | --- |
| RESTRICCIÓN | TIPO DE RESTRICCIÓN | IMPLEMENTACIÓN |
| RUT NOT NULL PRIMARY KEY | **Entidad** | **Estructural** |
| NOMBRE\_FANTASIA NOT NULL UNIQUE | **Entidad** | **Estructural** |
| EMAIL NOT NULL UNIQUE | **Entidad** | **Estructural** |
| DIRECCION NOT NULL | **Dominio** | **Estructural** |

**TABLA EMPRESA\_PRODUCTO**

|  |  |  |
| --- | --- | --- |
| RESTRICCIÓN | TIPO DE RESTRICCIÓN | IMPLEMENTACIÓN |
| RUT\_EMPRESA NOT NULL | **Dominio** | **Estructural** |
| CODIGO\_PRODUCTO NOT NULL | **Dominio** | **Estructural** |
| RUT\_EMPRESA CONSTRAINT EMP\_PROD\_TO\_EMPRESA\_FK REFERENCES EMPRESA | **Referencial** | **Estructural** |
| CODIGO\_PRODUCTO CONSTRAINT EMP\_PROD\_TO\_PRODUCTO\_FK REFERENCES PRODUCTO | **Referencial** | **Estructural** |
| CONSTRAINT EMPRESA\_PRODUCTO\_PK PRIMARY\_KEY (RUT\_EMPRESA, CODIGO\_PRODUCTO) | **Entidad** | **Estructural** |

**TABLA ESPECIALIDAD**

|  |  |  |
| --- | --- | --- |
| RESTRICCIÓN | TIPO DE RESTRICCIÓN | IMPLEMENTACIÓN |
| ID NOT NULL PRIMARY KEY | **Entidad** | **Estructural** |
| DESCRIPCION NOT NULL UNIQUE | **Entidad** | **Estructural** |
| CONSTRAINT ESPECIALIDAD\_DESCRIPCION CHECK (DESCRIPCION IN ()) | **Dominio** | **Estructural** |
| Especialidad\_id | **Semántica** | **No Estructural** |

**TABLA FUNCIONARIO**

|  |  |  |
| --- | --- | --- |
| RESTRICCIÓN | TIPO DE RESTRICCIÓN | IMPLEMENTACIÓN |
| DOCUMENTO NOT NULL PRIMARY KEY | **Entidad** | **Estructural** |
| TIPO NOT NULL | **Dominio** | **Estructural** |
| PAIS NOT NULL | **Dominio** | **Estructural** |
| DIRECCION NOT NULL | **Dominio** | **Estructural** |
| NOMBRE NOT NULL | **Dominio** | **Estructural** |
| FECHA\_NACIMIENTO NOT NULL | **Dominio** | **Estructural** |
| FECHA\_INGRESO NOT NULL | **Dominio** | **Estructural** |

**TABLA FUNCIONARIO\_ESPECIALIDAD**

|  |  |  |
| --- | --- | --- |
| RESTRICCIÓN | TIPO DE RESTRICCIÓN | IMPLEMENTACIÓN |
| DOCUMENTO\_FUNCIONARIO NOT NULL | **Dominio** | **Estructural** |
| ID\_ESPECIALIDAD NOT NULL | **Dominio** | **Estructural** |
| DOCUMENTO\_FUNCIONARIO CONSTRAINT FUNC\_ESP\_TO\_FUNCIONARIO\_FK REFERENCES FUNCIONARIO | **Referencial** | **Estructural** |
| ID\_ESPECIALIDAD CONSTRAINT FUNC\_ESP\_TO\_ESP\_FK REFERENCES ESPECIALIDAD | **Referencial** | **Estructural** |
| CONSTRAINT FUNCIONARIO\_ESPECIALIDAD\_PK PRIMARY\_KEY (DOCUMENTO\_FUNCIONARIO, ID\_ESPECIALIDAD) | **Entidad** | **Estructural** |

**TABLA INCIDENTE**

|  |  |  |
| --- | --- | --- |
| RESTRICCIÓN | TIPO DE RESTRICCIÓN | IMPLEMENTACIÓN |
| ID\_CONSULTA NOT NULL PRIMARY KEY | **Entidad** | **Estructural** |
| ID\_CONSULTA CONSTRAINT INCIDENTE\_TO\_CONS\_FK REFERENCES CONSULTA | **Referencial** | **Estructural** |
| ESTADO CONSTRAINT INCIDENTE\_ESTADO CHECK (ESTADO IN ()) | **Dominio** | **Estructural** |
| INCIDENTE\_YA\_RESUELTO | **Semántica** | **No Estructural** |
| INCIDENTE\_NO\_LO\_ES | **Semántica** | **No Estructural** |

**Comentarios**

* El *trigger* **INCIDENTE\_YA\_RESUELTO** se ejecuta al momento de crear una incidencia y valida que no pueda crearse una incidencia para una consulta que ya ha sido resuelta.
* El *trigger* **INCIDENTE\_NO\_LO\_ES** se ejecuta al momento de crear una incidencia y valida que no pueda crearse una incidencia para una consulta con una antigüedad menor a 24hs.

**TABLA PALABRA\_CLAVE**

|  |  |  |
| --- | --- | --- |
| RESTRICCIÓN | TIPO DE RESTRICCIÓN | IMPLEMENTACIÓN |
| ID PRIMARY KEY | **Entidad** | **Estructural** |
| PALABRA NOT NULL | **Dominio** | **Estructural** |
| Palabra\_clave\_id | **Semántica** | **No Estructural** |

**TABLA PALABRA\_CLAVE\_RESPUESTA**

|  |  |  |
| --- | --- | --- |
| RESTRICCIÓN | TIPO DE RESTRICCIÓN | IMPLEMENTACIÓN |
| ID\_RESPUESTA NOT NULL | **Dominio** | **Estructural** |
| ID\_PALABRA\_CLAVE NOT NULL | **Dominio** | **Estructural** |
| ID\_RESPUESTA CONSTRAINT P\_CLAVE\_RESP\_TO\_RESP\_FK REFERENCES RESPUESTA | **Referencial** | **Estructural** |
| ID\_PALABRA\_CLAVE CONSTRAINT P\_CLAVE\_RESP\_TO\_P\_CLAVE\_FK REFERENCES PALABRA\_CLAVE | **Referencial** | **Estructural** |
| CONSTRAINT PALABRA\_CLAVE\_RESPUESTA\_PK PRIMARY\_KEY (ID\_RESPUESTA, ID\_PALABRA\_CLAVE) | **Entidad** | **Estructural** |

**TABLA PRODUCTO**

|  |  |  |
| --- | --- | --- |
| RESTRICCIÓN | TIPO DE RESTRICCIÓN | IMPLEMENTACIÓN |
| CODIGO NOT NULL PRIMARY KEY | **Entidad** | **Estructural** |
| NOMBRE NOT NULL UNIQUE | **Entidad** | **Estructural** |

**TABLA RESPUESTA**

|  |  |  |
| --- | --- | --- |
| RESTRICCIÓN | TIPO DE RESTRICCIÓN | IMPLEMENTACIÓN |
| ID NOT NULL PRIMARY KEY | **Entidad** | **Estructural** |
| DOCUMENTO\_FUNCIONARIO NOT NULL | **Dominio** | **Estructural** |
| DOCUMENTO\_FUNCIONARIO CONSTRAINT RESPUESTA\_TO\_FUNCIONARIO\_FK REFERENCES FUNCIONARIO | **Referencial** | **Estructural** |
| TEXTO NOT NULL | **Dominio** | **Estructural** |
| ESTADO CONSTRAINT RESPUESTA\_SIMPLE\_ESTADO CHECK (ESTADO IN ()) | **Dominio** | **Estructural** |
| Respuesta\_id | **Semántica** | **No Estructural** |

**TABLA TELEFONO\_FUNCIONARIO**

|  |  |  |
| --- | --- | --- |
| RESTRICCIÓN | TIPO DE RESTRICCIÓN | IMPLEMENTACIÓN |
| DOCUMENTO\_FUNCIONARIO NOT NULL PRIMARY KEY | **Entidad** | **Estructural** |
| DOCUMENTO\_FUNCIONARIO CONSTRAINT TEL\_FUNC\_TO\_FUNCIONARIO\_FK REFERENCES FUNCIONARIO | **Referencial** | **Estructural** |
| TELEFONO NOT NULL | **Dominio** | **Estructural** |

**TABLA TIPO\_CONSULTA**

|  |  |  |
| --- | --- | --- |
| RESTRICCIÓN | TIPO DE RESTRICCIÓN | IMPLEMENTACIÓN |
| ID NOT NULL PRIMARY KEY | **Entidad** | **Estructural** |
| DESCRIPCION NOT NULL UNIQUE | **Entidad** | **Estructural** |
| ID\_ESPECIALIDAD NOT NULL | **Dominio** | **Estructural** |
| ID\_ESPECIALIDAD CONSTRAINT TIPO\_CONS\_TO\_ESP\_FK REFERENCES ESPECIALIDAD | **Referencial** | **Estructural** |
| Tipo\_consulta\_id | **Semántica** | **No Estructural** |

**Comentarios generales**

* Para el manejo de ids autoincrementales en las tablas se definen *sequences* que permiten extraer valores numéricos e insertarlos, incrementalmente, en los ids correspondientes. Para ello también se utilizan *triggers,* con la cláusula *before* *insert,* en las tablas que tengan ids como clave primaria y, previo a insertar un nuevo registro, este valor se incrementa. Este es el caso de por ejemplo las tablas **Especialidad**, **Tipo\_Consulta**, **Respuesta**, **Palabra\_Clave** y **Consulta** que definen los *triggers* **especialidad\_id**, **tipo\_consulta\_id**, **respuesta\_id**, **palabra\_clave\_id** y **consulta\_id** respectivamente que incrementan el valor de las *sequences* definidas.

# Datos de prueba Restricciones de Integridad

***Trigger FECHA\_CREACION\_CONSULTA***

Dados los siguientes datos de prueba

**INSERT INTO** EMPRESA **VALUES** (**'123456'**, **'Mi Empresa SA'**, **'miempresa@gmail.com'**, **'Avda Italia 1234'**, **'Juan Perez'**, **'099334455'**);  
  
**INSERT INTO** ESPECIALIDAD (**DESCRIPCION**) **VALUES** (**'Funcional'**);  
  
**INSERT INTO** TIPO\_CONSULTA (**DESCRIPCION**, **ID\_ESPECIALIDAD**) **VALUES** (**'Repuestos'**, (**SELECT** *MAX*(**ID**) **FROM** ESPECIALIDAD));  
  
**INSERT INTO** CONSULTA (**ID\_TIPO\_CONSULTA**, **RUT\_EMPRESA**, **DETALLE**, **VIA\_DE\_CONTACTO**, **TIEMPO\_DE\_COMUNICACION\_MINUTOS**)  
**VALUES** ((**SELECT** *MAX*(**ID**) **FROM** TIPO\_CONSULTA), **'123456'**, **'Una consulta'**, **'Personal'**, 20);

El trigger debería insertarle automáticamente una fecha de creación.

Si consultamos

**select** *\** **from** CONSULTA;

Vemos que esto funciona correctamente, se le asigna una **FECHA\_CREACION**



***Trigger FECHA\_RESPUESTA\_CONSULTA***

Considerando los datos de prueba anteriores, si creamos una respuesta genérica (para lo que tenemos que crear un funcionario)

**INSERT INTO** FUNCIONARIO (**DOCUMENTO**, **TIPO**, **PAIS**, **DIRECCION**, **NOMBRE**, **FECHA\_NACIMIENTO**, **FECHA\_INGRESO**)  
**VALUES** (47338517, **'Operario'**, **'Uruguay'**, **'Cuareim 1122'**, **'Juan Rodriguez'**, **'02/09/1996'**, **'08/08/2016'**);  
  
**INSERT INTO** RESPUESTA (**DOCUMENTO\_FUNCIONARIO**, **TEXTO**, **ESTADO**) **VALUES** (47338517, **'Respuesta Generica A'**, **'Revisada Ok'**);

Y luego le asignamos la respuesta a la consulta anterior

**UPDATE** CONSULTA **SET ID\_RESPUESTA** = (**SELECT** *MAX*(**ID**) **FROM** RESPUESTA) **WHERE ID** = 1;

Si consultamos

**select** *\** **from** CONSULTA;

Vemos que a la consulta se le asigna una **FECHA\_RESOLUCION**

****

***Trigger RESPUESTA\_INCIDENTE***

Si ingresamos una nueva consulta, con fecha de creación por lo menos un día mayor a la actual, para esto insertamos la consulta y luego actualizamos su fecha de creación manualmente (dado que la fecha de creación es modificada por el trigger al momento de la inserción)

**INSERT INTO** CONSULTA (**ID\_TIPO\_CONSULTA**, **RUT\_EMPRESA**, **DETALLE**, **VIA\_DE\_CONTACTO**, **TIEMPO\_DE\_COMUNICACION\_MINUTOS**)  
**VALUES** ((**SELECT** *MAX*(**ID**) **FROM** TIPO\_CONSULTA), **'123456'**, **'Consulta 2'**, **'Personal'**, 20);

**UPDATE** CONSULTA **SET FECHA\_CREACION** = **'28/11/2017' WHERE ID** = 3

Y luego generamos una incidencia para dicha consulta

**INSERT INTO** INCIDENTE (**ID\_CONSULTA**, **ESTADO**) **VALUES** (3, **'Pendiente'**);

Si le asignamos una respuesta a dicha consulta, podremos verificar que el incidente queda en estado *resuelto.*

**UPDATE** CONSULTA **SET ID\_RESPUESTA** = (**SELECT** *MAX*(**ID**) **FROM** RESPUESTA) **WHERE ID** = 3;

**select** *\** **from** INCIDENTE;



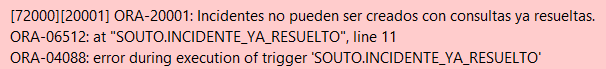
***Trigger INCIDENTE\_YA\_RESUELTO***

Si para la misma consulta anterior, ya resuelta, intentamos generar un nuevo incidente se producirá un error ya que el trigger INDICENTE\_YA\_RESUELTO no permite la creación de nuevas incidencias en una consulta ya resuelta.

Intentamos nuevamente

**INSERT INTO** INCIDENTE (**ID\_CONSULTA**, **ESTADO**) **VALUES** (3, **'Pendiente'**);

Y obtenemos el error esperado



***Trigger INCIDENTE\_NO\_LO\_ES***

Este trigger valida que no se puedan generar incidencias para consultas con antigüedad menor a 24 hs. Para ello insertamos una nueva consulta, y se le asignará fecha de creación la actual en el sistema

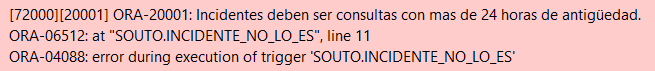
**INSERT INTO** CONSULTA (**ID\_TIPO\_CONSULTA**, **RUT\_EMPRESA**, **DETALLE**, **VIA\_DE\_CONTACTO**, **TIEMPO\_DE\_COMUNICACION\_MINUTOS**)  
**VALUES** ((**SELECT** *MAX*(**ID**) **FROM** TIPO\_CONSULTA), **'123456'**, **'Consulta 3'**, **'Personal'**, 20);



Luego generamos una incidencia

**INSERT INTO** INCIDENTE (**ID\_CONSULTA**, **ESTADO**) **VALUES** (5, **'Pendiente'**);

Y obtenemos el error



***Trigger RESPUESTA\_ESPECIALIDAD\_UNICA***

Este *trigger* valida que si se ingresa una consulta y a la misma se le asigna una respuesta *generalizable* que ya ha sido utilizada se deberá cumplir que el tipo de consulta de dicha consulta tenga asignado la misma *especialidad.* Es decir, una respuesta *generalizable* no puede responder consultas cuyo tipo de consulta tiene especialidades diferentes.

En primer lugar ingresamos los datos necesarios, la empresa, dos especialidades y tipos de consulta con distintas especialidades

**INSERT INTO** EMPRESA **VALUES** (**'123456'**, **'Mi Empresa SA'**, **'miempresa@gmail.com'**, **'Avda Italia 1234'**, **'Juan Perez'**, **'099334455'**);  
  
**INSERT INTO** ESPECIALIDAD (**DESCRIPCION**) **VALUES** (**'Funcional'**);  
**INSERT INTO** ESPECIALIDAD (**DESCRIPCION**) **VALUES** (**'Investigacion'**);

**INSERT INTO** TIPO\_CONSULTA (**DESCRIPCION**, **ID\_ESPECIALIDAD**) **VALUES** (**'Tipo F'**, (**SELECT ID FROM** ESPECIALIDAD **WHERE** ESPECIALIDAD.**DESCRIPCION** = **'Funcional'**));  
**INSERT INTO** TIPO\_CONSULTA (**DESCRIPCION**, **ID\_ESPECIALIDAD**) **VALUES** (**'Tipo I'**, (**SELECT ID FROM** ESPECIALIDAD **WHERE** ESPECIALIDAD.**DESCRIPCION** = **'Investigacion'**));

Luego ingresamos un funcionario y una respuesta *generalizable*

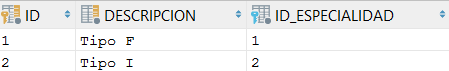
**INSERT INTO** FUNCIONARIO (**DOCUMENTO**, **TIPO**, **PAIS**, **DIRECCION**, **NOMBRE**, **FECHA\_NACIMIENTO**, **FECHA\_INGRESO**)  
**VALUES** (47338517, **'Operario'**, **'Uruguay'**, **'Cuareim 1122'**, **'Juan Rodriguez'**, **'02/09/1996'**, **'08/08/2016'**);  
  
**INSERT INTO** RESPUESTA (**DOCUMENTO\_FUNCIONARIO**, **TEXTO**, **ESTADO**) **VALUES** (47338517, **'Respuesta Generica A'**, **'Revisada Ok'**);

Hasta ese momento nadie ha utilizado la respuesta *generalizable* por lo que *cualquier* consulta puede ser respondida con esta respuesta, independientemente de la especialidad de su tipo de consulta.

**INSERT INTO** CONSULTA (**ID\_TIPO\_CONSULTA**, **RUT\_EMPRESA**, **DETALLE**, **VIA\_DE\_CONTACTO**, **TIEMPO\_DE\_COMUNICACION\_MINUTOS**, **ID\_RESPUESTA**)  
**VALUES** ((**SELECT ID FROM** TIPO\_CONSULTA **WHERE DESCRIPCION** = **'Tipo F'**), **'123456'**, **'Una consulta'**, **'Personal'**, 20, 1);

Ingresamos una consulta, con id de respuesta la respuesta *generalizable* anterior. Nótese que el tipo de consulta es **Tipo F** y su especialidad es **Funcional**.

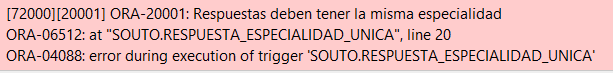




Si ahora quisiéramos ingresar una nueva consulta, con id tipo de consulta **Tipo I** y la misma respuesta *generalizable*, se debería producir un error, ya que los tipos de consulta tienen especialidades diferentes.

**INSERT INTO** CONSULTA (**ID\_TIPO\_CONSULTA**, **RUT\_EMPRESA**, **DETALLE**, **VIA\_DE\_CONTACTO**, **TIEMPO\_DE\_COMUNICACION\_MINUTOS**, **ID\_RESPUESTA**)  
**VALUES** ((**SELECT ID FROM** TIPO\_CONSULTA **WHERE DESCRIPCION** = **'Tipo I'**), **'123456'**, **'Una consulta con otro tipo'**, **'Personal'**, 20, 1);

Y en este caso obtenemos el error esperado



# Explicación de Procedimientos

## Procedimiento 1

1. Recolección de posibles respuestas genéricas: Con el objetivo de potenciar el área de atención al cliente a través de respuestas genéricas, se requiere un servicio que sirva para catalogar y almacenar respuestas que podrían llegar a generalizarse. Los tipos de consultas están relacionadas con las especialidades de los funcionarios. Por ejemplo, la especialidad “Funcional” es la más capacitada para atender las consultas de tipo “Falla de producto” o “Facturación”. El servicio será invocado desde la aplicación cuando el funcionario que responde una consulta a un determinado problema, considera que la respuesta puede llegar a generalizarse. El servicio recibe como parámetro el tipo de consulta, el detalle de la consulta, el o los productos relacionados, la respuesta ofrecida y el funcionario. La respuesta debe quedar almacenada en la base de datos en el estado “Pendiente de revisión”, para que luego pueda ser tomada por otro funcionario que la revise, la modifique si es necesario, defina sus palabras clave y la pase a un estado “Revisada OK”.

Se crea una tabla nueva para recibir productos por parámetro

**CREATE** **TABLE** IDS\_PRODUCTOS\_RESP\_GEN (

ID\_PRODUCTO **NUMBER**(10) **CONSTRAINT** ID\_PRODUCTO\_FK **REFERENCES** PRODUCTO

);

Procedimiento:

**CREATE** **OR** **REPLACE** **PROCEDURE** GenerarRespuestaGenerica(id\_tipo\_cons **IN** **NUMBER**,

detalle\_consulta **in** **VARCHAR**,

respuesta\_ofrecida **IN** **VARCHAR**,

doc\_funcionario **IN** **NUMBER**,

rut\_cliente **IN** **NUMBER**,

via\_comunicacion **in** **VARCHAR**,

tiempo\_de\_comunicacion **in** **NUMBER**)

**IS**

id\_nueva\_consulta **NUMBER**;

id\_nueva\_respuesta **NUMBER**;

**BEGIN**

-- Se1eccionamos los productos pasados por parametro

**DECLARE** **CURSOR** cursor\_productos **IS**

**SELECT** ID\_PRODUCTO

**FROM** IDS\_PRODUCTOS\_RESP\_GEN;

**BEGIN**

-- Se inserta la respuesta generalizable en estado 'Pendiente de revision'

**INSERT** **INTO** RESPUESTA (DOCUMENTO\_FUNCIONARIO, TEXTO, ESTADO)

**VALUES** (doc\_funcionario, respuesta\_ofrecida, 'Pendiente de revision')

**RETURNING** **ID** **into** id\_nueva\_respuesta;

-- Se inserta la consulta recibida asociada a la respuesta

**INSERT** **INTO** CONSULTA (ID\_TIPO\_CONSULTA, RUT\_EMPRESA, DETALLE, VIA\_DE\_CONTACTO, ID\_RESPUESTA, TIEMPO\_DE\_COMUNICACION\_MINUTOS)

**VALUES** (id\_tipo\_cons, rut\_cliente, detalle\_consulta, via\_comunicacion, id\_nueva\_respuesta, tiempo\_de\_comunicacion)

**RETURNING** **ID** **into** id\_nueva\_consulta;

-- Por cada uno de los productos recibidos por parametro se asocia a la consulta a

-- través de la tabla intermedia

**FOR** producto **IN** cursor\_productos

**LOOP**

**INSERT** **INTO** CONSULTA\_PRODUCTO (ID\_CONSULTA, CODIGO\_PRODUCTO) **VALUES** (id\_nueva\_consulta, producto.ID\_PRODUCTO);

**END LOOP**;

**COMMIT**;

**END**;

**END**;

## Procedimiento 2

2. Generación de incidencias: Se requiere un servicio que genere automáticamente las incidencias para las consultas que lleven más de 24hs sin responder. Nota: Una incidencia es una consulta que debe ser estudiada en profundidad.

--- Procedimiento 2

**CREATE** **OR** **REPLACE** **PROCEDURE** GenerarIncidentes

**IS**

**BEGIN**

-- Declaramos cursor que nos permitirá iterar sobre las consultas deseadas

**DECLARE** **CURSOR** Consultas **IS**

-- Query que nos trae todas las consultas de más de 24 horas de antigüedad que no son

-- incidentes

**SELECT** **ID**

**FROM** CONSULTA

**WHERE** ID\_RESPUESTA **IS** **NULL**

-- Control de antigüedad

**AND** **EXTRACT**(**DAY** **FROM** (sys\_extract\_utc(**systimestamp**) - FECHA\_CREACION)) >= 1

-- Control de que no sea un incidente

**AND** **ID** **NOT** **IN** (**SELECT** ID\_CONSULTA **FROM** INCIDENTE);

**BEGIN**

-- Se itera sobre el cursor

**FOR** id\_consulta\_t **IN** Consultas

**LOOP**

-- Se inserta cada consulta como incidente pendiente

**INSERT** **INTO** INCIDENTE (ID\_CONSULTA, ESTADO) **VALUES** (id\_consulta\_t.**ID**, 'Pendiente');

-- El commit esta ubicado dentro del for porque en caso de que el procedimiento se

-- detenga en el medio, por letra deberá continuar en donde se quedó, con ayuda de

-- este commit y de que la consulta que popula el cursor trae solamente las

-- consultas que no son incidentes, podemos estar seguros de que el comportamiento

-- pedido se cumplirá.

**COMMIT**;

**END LOOP**;

**END**;

**END**;

## Procedimiento 3

3. Definición de respuestas genéricas: Para facilitar el proceso de revisión de las respuestas genéricas, se requiere un servicio que actualice las respuestas pendientes de revisión. Debe recibir por parámetro el identificador de una respuesta genérica y el estado al que pasará. Si es rechazada, se pasa al estado “Rechazada”. Si es aceptada, se pasa al estado “Revisada OK”. En este último caso, además se debe pasar por parámetro las palabras clave definidas.

Creación de table de parámetros:

**CREATE** **TABLE** PALABRAS\_CLAVE\_PARA\_GENERICA (

ID\_PALABRA\_CLAVE **NUMBER**(10) **CONSTRAINT** ID\_PALABRA\_CLAVE\_FK **REFERENCES** PALABRA\_CLAVE

);

Procedimiento:

**CREATE** **OR** **REPLACE** **PROCEDURE** RevisarRespuestaGenerica(id\_respuesta\_in **IN** **NUMBER**, estado\_deseado **IN** **VARCHAR**) **IS**

palabras\_clave\_query **VARCHAR**(100);

**BEGIN**

-- Se seleccionan las palabras clave que se cargaron como parametro

**DECLARE** **CURSOR** palabraParametro **IS**

**SELECT** ID\_PALABRA\_CLAVE

**FROM** PALABRAS\_CLAVE\_PARA\_GENERICA;

**BEGIN**

-- Se le agrega el estado deseado a la respuesta deseada solamente si el estado

-- era nulo

**UPDATE** RESPUESTA

**SET** RESPUESTA.ESTADO = estado\_deseado

**WHERE** RESPUESTA.**ID** = id\_respuesta\_in **AND** RESPUESTA.ESTADO **IS** **NOT** **NULL**;

-- Se le borran las palabras claves que poseia

**DELETE** **FROM** PALABRA\_CLAVE\_RESPUESTA

**WHERE** ID\_RESPUESTA = id\_respuesta\_in;

**IF** estado\_deseado = 'Revisada Ok'

**THEN**

-- Si fue aceptada como respuesta generica se le asignan las palabras claves

-- cargadas como parametro

**FOR** palabra\_clave\_t **IN** palabraParametro

**LOOP**

**INSERT** **INTO** PALABRA\_CLAVE\_RESPUESTA (ID\_RESPUESTA, ID\_PALABRA\_CLAVE)

**VALUES** (id\_respuesta\_in, palabra\_clave\_t.ID\_PALABRA\_CLAVE);

**END LOOP**;

**END IF**;

-- Se borra la tabla de parametros

**DELETE** **FROM** PALABRAS\_CLAVE\_PARA\_GENERICA;

**COMMIT**;

**END**;

**END**;

## Procedimiento 4

Respuestas genéricas de una consulta: Se requiere un servicio que reciba un tipo de consulta, una o varias palabras clave y devuelva: el/los funcionario(s) especializados en dicho tipo de consultas y la/las respuesta(s) genérica(s) relacionada(s).  
Importante: Solo incluir las respuestas que fueron revisadas (estado “Revisada OK”).

Primero cabe aclarar que creamos una tabla nueva para recibir las palabras claves como parámetro, para permitir que se reciban todas las que el usuario quiera.

**CREATE** **TABLE** PALABRA\_CLAVE\_PROCEDIMIENTO (

ID\_PALABRA\_CLAVE **NUMBER**(10) **CONSTRAINT** PALABRA\_CLAVE\_PROCEDIMIENTO\_FK **REFERENCES** PALABRA\_CLAVE

);

Código del procedimiento:

**CREATE** **OR** **REPLACE** **PROCEDURE** RESPUESTASGENERICAS(tipo\_de\_consulta **IN** **VARCHAR2**)

**IS**

**BEGIN**

**DECLARE**

-- Declaramos cursor que nos permitira iterar sobre las respuestas genéricas

-- relacionadas con el tipo de consulta recibido

**CURSOR** RESPUESTAS **IS**

**SELECT** **R**.TEXTO

-- Selecciono todas las consultas

**FROM** CONSULTA **C**

-- Hago join con todas las respuestas de esas consultas

**JOIN** RESPUESTA **R** **ON** **C**.ID\_RESPUESTA = **R**.**ID**

-- Tambien con todos los tipos de consulta de esas consultas

**JOIN** TIPO\_CONSULTA TC **ON** **C**.ID\_TIPO\_CONSULTA = TC.**ID**

-- Luego con las especialidades de esas consultas

**JOIN** ESPECIALIDAD E **ON** TC.ID\_ESPECIALIDAD = E.**ID**

-- Joineamos con la tabla intermedia que relaciona (n-n) las respuestas con sus

-- palabras clave

**JOIN** PALABRA\_CLAVE\_RESPUESTA PCR **ON** **R**.**ID** = PCR.ID\_RESPUESTA

-- Por ultimo las palabras claves que se encontraron en la tabla intermedia

**JOIN** PALABRA\_CLAVE PC **ON** PCR.ID\_PALABRA\_CLAVE = PC.**ID**

-- Controlamos que el tipo de consulta sea el mismo que el recibido

--parametro

**WHERE** TC.DESCRIPCION = tipo\_de\_consulta

-- Controlamos que el estado de la respuesta seleccionada sea 'Revisada Ok'

**AND** **R**.ESTADO = 'Revisada Ok'

-- Controlamos que la palabra clave relacionada con la respuesta esté en la

-- tabla recibida como parametro

**AND** PC.PALABRA **IN** (**SELECT** PALABRA

**FROM** PALABRA\_CLAVE\_PROCEDIMIENTO

**JOIN** PALABRA\_CLAVE

**ON** PALABRA\_CLAVE\_PROCEDIMIENTO.ID\_PALABRA\_CLAVE = PALABRA\_CLAVE.**ID**);

-- Declaramos cursor que nos permitira iterar sobre los funcionarios

**CURSOR** FUNCIONARIOS **IS**

**SELECT** F.NOMBRE

-- Seleccionamos desde la tabla tipo de consulta

**FROM** TIPO\_CONSULTA TC

-- Basicamente pasa por los funcionarios y especialidades y nos deja con los datos

-- de cada funcionario y sus especialidades

**JOIN** ESPECIALIDAD E **ON** TC.ID\_ESPECIALIDAD = E.**ID**

**JOIN** FUNCIONARIO\_ESPECIALIDAD FE **ON** E.**ID** = FE.ID\_ESPECIALIDAD

**JOIN** FUNCIONARIO F **ON** FE.DOCUMENTO\_FUNCIONARIO = F.DOCUMENTO

-- Controlamos que el tipo de consulta de el funcionario sea el recibido

**WHERE** TC.DESCRIPCION = tipo\_de\_consulta;

**BEGIN**

-- Imprimimos el resultado del primer cursor

**DBMS\_OUTPUT**.**PUT\_LINE**('RESPUESTAS: ');

**FOR** respuesta\_cursor **IN** RESPUESTAS

**LOOP**

**DBMS\_OUTPUT**.**PUT\_LINE**(respuesta\_cursor.TEXTO);

**DBMS\_OUTPUT**.**PUT\_LINE**('----------------------------------------------');

**END LOOP**;

-- Imprimimos el resultado del segundo cursor

**DBMS\_OUTPUT**.**PUT\_LINE**('FUNCIONARIOS: ');

**FOR** funcionario\_cursor **IN** FUNCIONARIOS

**LOOP**

**DBMS\_OUTPUT**.**PUT\_LINE**(funcionario\_cursor.NOMBRE);

**DBMS\_OUTPUT**.**PUT\_LINE**('----------------------------------------------');

**END LOOP**;

-- Eliminamos los datos de la tabla para que los parámetros no queden manchados

-- para la próxima

**DELETE** **FROM** PALABRA\_CLAVE\_PROCEDIMIENTO;

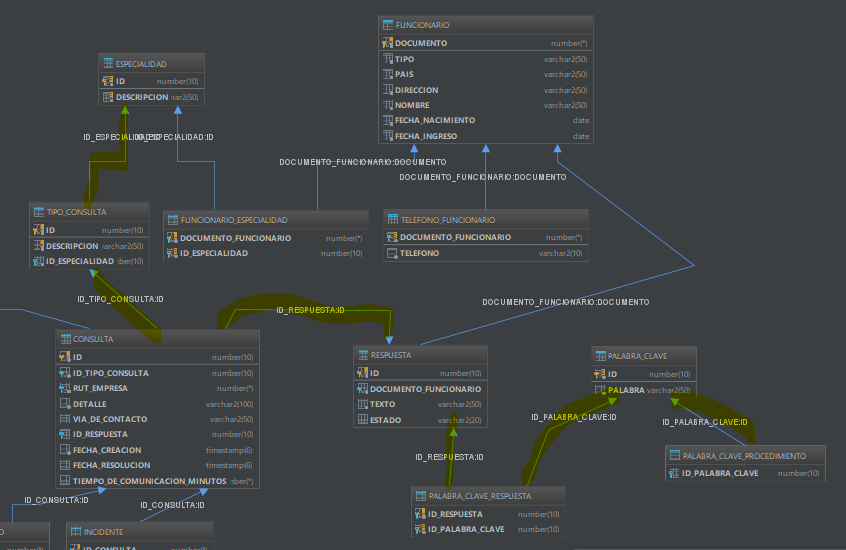
-- Se hace commit para que el delete tenga efecto

**COMMIT**;

**END**;

**END**;

Entendemos que puede ser bastante compleja la primer consulta, por lo que les dejamos un pequeño diagrama resaltando las relaciones utilizadas para llegar a la información necesaria de cada respuesta.



## Procedimiento 5

5. Actualización de las especialidades de los funcionarios: Mensualmente se actualizan las especialidades de los funcionarios. Dicha actualización se basa en la cantidad de tipos de respuestas satisfactorias de cada funcionario de la siguiente forma: Si un funcionario tiene más de 20 respuestas en el mes para un mismo tipo de consulta, se le asigna la especialidad correspondiente (si aún no la tenía). El servicio recibe por parámetro el mes y el año requerido. Si no se pasa ningún parámetro se asume el mes anterior.

**CREATE** **OR** **REPLACE** **PROCEDURE** ACTUALIZAR\_ESPECIALIDADES (mes **NUMBER** **default** **EXTRACT**(**MONTH** **FROM** (sys\_extract\_utc(**systimestamp**))), anio **number** **default** **EXTRACT**(**YEAR** **FROM** (sys\_extract\_utc(**systimestamp**))))

**IS**

count\_especialidades **NUMBER**;

**BEGIN**

-- Se declara un cursor que almacenara la cantidad de respuestas por especialidad de cada funcionario

**DECLARE** **CURSOR** RespPorEspFunc **IS**

**SELECT**

**R**.DOCUMENTO\_FUNCIONARIO,

**COUNT**(**R**.**ID**) RESPUESTAS,

E.**ID** ESPECIALIDAD

-- Desde las respuestas se llega a la especialidad

**FROM** RESPUESTA **R**

**JOIN** CONSULTA **C** **ON** **R**.**ID** = **C**.ID\_RESPUESTA

**JOIN** TIPO\_CONSULTA TC **ON** **C**.ID\_TIPO\_CONSULTA = TC.**ID**

**JOIN** ESPECIALIDAD E **ON** TC.ID\_ESPECIALIDAD = E.**ID**

-- Se controla que el mes y el año sean los que llegaron por parametro (que tienen como default los actuales)

**WHERE** **EXTRACT**(**MONTH** **FROM** **C**.FECHA\_RESOLUCION) = mes

**AND** **EXTRACT**(**YEAR** **FROM** **C**.FECHA\_RESOLUCION) = anio

-- Se controla que hayan más de 20 respuestas por especialidad

**HAVING** **COUNT**(**R**.**ID**) > 20

**GROUP** **BY** **R**.DOCUMENTO\_FUNCIONARIO, **C**.ID\_TIPO\_CONSULTA, E.**ID**;

**BEGIN**

**FOR** **row** **IN** RespPorEspFunc

-- Se itera por cada conjunto de más de 20 preguntas funcionario-especialidad

**LOOP**

-- Se chequea que el funcionario tenga esa especialidad

**SELECT** **COUNT**(ID\_ESPECIALIDAD)

**INTO** count\_especialidades

**FROM** FUNCIONARIO\_ESPECIALIDAD

**WHERE** ID\_ESPECIALIDAD = **row**.ESPECIALIDAD

**AND** DOCUMENTO\_FUNCIONARIO = **row**.DOCUMENTO\_FUNCIONARIO;

-- Si el funcionario no tiene esa especialidad se le asigna

**IF** count\_especialidades < 1

**THEN**

**INSERT** **INTO** FUNCIONARIO\_ESPECIALIDAD **VALUES** (**row**.DOCUMENTO\_FUNCIONARIO, **row**.ESPECIALIDAD);

**END IF**;

**END LOOP**;

-- Commit al final del loop para que lo haga al final y no queden datos inconsistentes.

**COMMIT**;

**END**;

**END**;

## pROCEDIMIENTO 6

Se crean las tablas

**CREATE** **TABLE** CACHE\_INFO\_FUNCIONARIOS (

MES **NUMBER** **NOT** **NULL**,

ANO **NUMBER** **NOT** **NULL**,

RESULTADO **VARCHAR**(2500),

**PRIMARY** **KEY** (MES, ANO)

);

**CREATE** **TABLE** ULTIMO\_FUNCIONARIO (

DOCUMENTO\_FUNCIONARIO **NUMBER** **NOT** **NULL** **CONSTRAINT** DOCUMENTO\_FUN\_ULTIMO\_FUN **REFERENCES** FUNCIONARIO,

MES **NUMBER** **NOT** **NULL**,

ANO **NUMBER** **NOT** **NULL**,

**PRIMARY** **KEY** (DOCUMENTO\_FUNCIONARIO, MES, ANO)

)

Procedimiento:

**CREATE** **OR** **REPLACE** **PROCEDURE** InfoFuncionarios(mes\_param **NUMBER** **DEFAULT** **NULL**, ano\_param **NUMBER** **DEFAULT** **NULL**) **IS**

mes\_comprobado **NUMBER**;

ano\_comprobado **NUMBER**;

texto\_resultado **VARCHAR**(2500);

hay\_cache **NUMBER**;

ultimo\_funcionario\_doc **NUMBER**;

**BEGIN**

**IF** mes\_param **IS** **NULL** **OR** ano\_param **IS** **NULL**

**THEN**

mes\_comprobado := **EXTRACT**(**MONTH** **FROM** **add\_months**(**sysdate**, -1));

ano\_comprobado := **EXTRACT**(**YEAR** **FROM** **add\_months**(**sysdate**, -1));

**ELSE**

mes\_comprobado := mes\_param;

ano\_comprobado := ano\_param;

**END IF**;

**SELECT** **COUNT**(\*)

**INTO** hay\_cache

**FROM**

CACHE\_INFO\_FUNCIONARIOS

**WHERE** CACHE\_INFO\_FUNCIONARIOS.MES = mes\_comprobado **AND** CACHE\_INFO\_FUNCIONARIOS.ANO = ano\_comprobado;

**SELECT** **COALESCE**(**MAX**(ULTIMO\_FUNCIONARIO.DOCUMENTO\_FUNCIONARIO), 0)

**INTO** ultimo\_funcionario\_doc

**FROM** ULTIMO\_FUNCIONARIO

**WHERE** ULTIMO\_FUNCIONARIO.MES = mes\_comprobado **AND** ULTIMO\_FUNCIONARIO.ANO = ano\_comprobado;

**IF** hay\_cache = 0 **OR** ultimo\_funcionario\_doc != 0

**THEN**

**IF** ultimo\_funcionario\_doc = 0

**THEN**

ActualizarCacheFuncionarios(mes\_comprobado, ano\_comprobado, mes\_comprobado || '/' || ano\_comprobado);

**END IF**;

ImprimirFuncionarios(mes\_comprobado, ano\_comprobado, ultimo\_funcionario\_doc);

**END IF**;

**SELECT** RESULTADO

**INTO** texto\_resultado

**FROM** CACHE\_INFO\_FUNCIONARIOS

**WHERE** CACHE\_INFO\_FUNCIONARIOS.MES = mes\_comprobado **AND** CACHE\_INFO\_FUNCIONARIOS.ANO = ano\_comprobado;

**DBMS\_OUTPUT**.**PUT\_LINE**(texto\_resultado);

**DELETE** **FROM** ULTIMO\_FUNCIONARIO

**WHERE** ULTIMO\_FUNCIONARIO.MES = mes\_comprobado **AND** ULTIMO\_FUNCIONARIO.ANO = ano\_comprobado;

**COMMIT**;

**END**;

**CREATE** **OR** **REPLACE** **PROCEDURE** ImprimirFuncionarios(mes\_param **NUMBER**, ano\_param **NUMBER**,

documento\_desde **NUMBER** **DEFAULT** 0) **IS**

cantidad\_respuestas **NUMBER**;

promedio\_resolucion **NUMBER**;

**BEGIN**

**DECLARE** **CURSOR** c\_funcionarios **IS** **SELECT**

FUNCIONARIO.DOCUMENTO,

FUNCIONARIO.NOMBRE,

FUNCIONARIO.FECHA\_INGRESO,

LISTAGG(ESPECIALIDAD.DESCRIPCION, ', ')

WITHIN **GROUP** (

**ORDER** **BY** ESPECIALIDAD.DESCRIPCION) "ESPECIALIDADES\_TEXTO"

**FROM** FUNCIONARIO

**JOIN** FUNCIONARIO\_ESPECIALIDAD

**ON** FUNCIONARIO.DOCUMENTO = FUNCIONARIO\_ESPECIALIDAD.DOCUMENTO\_FUNCIONARIO

**JOIN** ESPECIALIDAD **ON** FUNCIONARIO\_ESPECIALIDAD.ID\_ESPECIALIDAD = ESPECIALIDAD.**ID**

**WHERE** FUNCIONARIO.DOCUMENTO > documento\_desde

**GROUP** **BY** DOCUMENTO, FUNCIONARIO.NOMBRE, FUNCIONARIO.FECHA\_INGRESO

**ORDER** **BY** FUNCIONARIO.DOCUMENTO;

**BEGIN**

**FOR** row\_funcionario **IN** c\_funcionarios

**LOOP**

ActualizarCacheFuncionarios(mes\_param, ano\_param,

'Funcionario: ' || row\_funcionario.NOMBRE || ' Fecha de ingreso: ' ||

row\_funcionario.FECHA\_INGRESO ||

' Especialidades: ' || row\_funcionario.ESPECIALIDADES\_TEXTO);

ActualizarCacheFuncionarios(mes\_param, ano\_param,

'-----------------------------------------------------------------------');

**DECLARE** **CURSOR** c\_tipos\_de\_consulta **IS** **SELECT**

TIPO\_CONSULTA.**ID**,

TIPO\_CONSULTA.DESCRIPCION

**FROM** TIPO\_CONSULTA

**JOIN** CONSULTA **ON** TIPO\_CONSULTA.**ID** = CONSULTA.ID\_TIPO\_CONSULTA

**JOIN** RESPUESTA **ON** CONSULTA.ID\_RESPUESTA = RESPUESTA.**ID**

**WHERE** RESPUESTA.DOCUMENTO\_FUNCIONARIO = row\_funcionario.DOCUMENTO

**AND** CONSULTA.FECHA\_RESOLUCION **IS** **NOT** **NULL**

**AND** **EXTRACT**(**MONTH** **FROM** CONSULTA.FECHA\_RESOLUCION) = mes\_param **AND**

**EXTRACT**(**YEAR** **FROM** CONSULTA.FECHA\_RESOLUCION) = ano\_param

**GROUP** **BY** TIPO\_CONSULTA.**ID**, TIPO\_CONSULTA.DESCRIPCION;

**BEGIN**

**FOR** row\_tipo\_consulta **IN** c\_tipos\_de\_consulta

**LOOP**

ActualizarCacheFuncionarios(mes\_param, ano\_param, 'Tipo de Consulta: ' || row\_tipo\_consulta.DESCRIPCION);

**DECLARE** **CURSOR** c\_clientes **IS** **SELECT**

EMPRESA.NOMBRE\_FANTASIA,

**SUM**(**EXTRACT**(**MINUTE** **FROM** (CONSULTA.FECHA\_RESOLUCION -

CONSULTA.FECHA\_CREACION))) "TIEMPO\_RESOLUCION"

**FROM** EMPRESA

**JOIN** CONSULTA **ON** EMPRESA.RUT = CONSULTA.RUT\_EMPRESA

**JOIN** RESPUESTA **ON** CONSULTA.ID\_RESPUESTA = RESPUESTA.**ID**

**WHERE** CONSULTA.ID\_TIPO\_CONSULTA = row\_tipo\_consulta.**ID** **AND**

RESPUESTA.DOCUMENTO\_FUNCIONARIO = row\_funcionario.DOCUMENTO

**AND** CONSULTA.FECHA\_RESOLUCION **IS** **NOT** **NULL**

**AND** **EXTRACT**(**MONTH** **FROM** CONSULTA.FECHA\_RESOLUCION) = mes\_param **AND**

**EXTRACT**(**YEAR** **FROM** CONSULTA.FECHA\_RESOLUCION) = ano\_param

**GROUP** **BY** EMPRESA.NOMBRE\_FANTASIA;

**BEGIN**

**FOR** row\_cliente **IN** c\_clientes

**LOOP**

ActualizarCacheFuncionarios(mes\_param, ano\_param,

row\_cliente.NOMBRE\_FANTASIA || ' ' ||

row\_cliente.TIEMPO\_RESOLUCION);

**END LOOP**;

**SELECT** **COUNT**(RESPUESTA.**ID**)

**INTO** cantidad\_respuestas

**FROM** RESPUESTA

**JOIN** CONSULTA **ON** RESPUESTA.**ID** = CONSULTA.ID\_RESPUESTA

**WHERE** RESPUESTA.DOCUMENTO\_FUNCIONARIO = row\_funcionario.DOCUMENTO **AND**

CONSULTA.ID\_TIPO\_CONSULTA = row\_tipo\_consulta.**ID**

**AND** CONSULTA.FECHA\_RESOLUCION **IS** **NOT** **NULL**

**AND** **EXTRACT**(**MONTH** **FROM** CONSULTA.FECHA\_RESOLUCION) = mes\_param **AND**

**EXTRACT**(**YEAR** **FROM** CONSULTA.FECHA\_RESOLUCION) = ano\_param;

ActualizarCacheFuncionarios(mes\_param, ano\_param, 'Cantidad: ' || cantidad\_respuestas);

**SELECT** **AVG**(**EXTRACT**(**MINUTE** **FROM** (CONSULTA.FECHA\_RESOLUCION -

CONSULTA.FECHA\_CREACION)))

**INTO** promedio\_resolucion

**FROM** CONSULTA

**JOIN** RESPUESTA **ON** CONSULTA.ID\_RESPUESTA = RESPUESTA.**ID**

**WHERE** RESPUESTA.DOCUMENTO\_FUNCIONARIO = row\_funcionario.DOCUMENTO **AND**

CONSULTA.ID\_TIPO\_CONSULTA = row\_tipo\_consulta.**ID**

**AND** CONSULTA.FECHA\_RESOLUCION **IS** **NOT** **NULL**

**AND** **EXTRACT**(**MONTH** **FROM** CONSULTA.FECHA\_RESOLUCION) = mes\_param **AND**

**EXTRACT**(**YEAR** **FROM** CONSULTA.FECHA\_RESOLUCION) = ano\_param;

ActualizarCacheFuncionarios(mes\_param, ano\_param, 'Tiempo promedio: ' || promedio\_resolucion);

ActualizarCacheFuncionarios(mes\_param, ano\_param, '----------------------------');

**END**;

**END LOOP**;

**END**;

**INSERT** **INTO** ULTIMO\_FUNCIONARIO (DOCUMENTO\_FUNCIONARIO, ANO, MES)

**VALUES** (row\_funcionario.DOCUMENTO, ano, mes\_param);

**COMMIT**;

**END LOOP**;

**END**;

**END**;

**CREATE** **OR** **REPLACE** **PROCEDURE** ActualizarCacheFuncionarios(mes\_param **NUMBER**, ano\_param **NUMBER**, texto **VARCHAR**) **IS**

existe\_cache **NUMBER**;

**BEGIN**

**SELECT** **COUNT**(\*)

**INTO** existe\_cache

**FROM** CACHE\_INFO\_FUNCIONARIOS

**WHERE** CACHE\_INFO\_FUNCIONARIOS.ANO = ano\_param **AND** CACHE\_INFO\_FUNCIONARIOS.MES = mes\_param;

**IF** existe\_cache = 0

**THEN**

**INSERT** **INTO** CACHE\_INFO\_FUNCIONARIOS (MES, ANO, RESULTADO)

**VALUES** (mes\_param, ano\_param, texto || **CHR**(13) || **CHR**(10)); -- \r\n

**ELSE**

**UPDATE** CACHE\_INFO\_FUNCIONARIOS

**SET** CACHE\_INFO\_FUNCIONARIOS.RESULTADO = **CONCAT**(CACHE\_INFO\_FUNCIONARIOS.RESULTADO, texto || **CHR**(13) || **CHR**(10))

**WHERE** CACHE\_INFO\_FUNCIONARIOS.ANO = ano\_param **AND** CACHE\_INFO\_FUNCIONARIOS.MES = mes\_param;

**END IF**;

**END**;

# Datos de Prueba Procedimientos

**Procedimiento 1**

Dados los siguientes datos de prueba

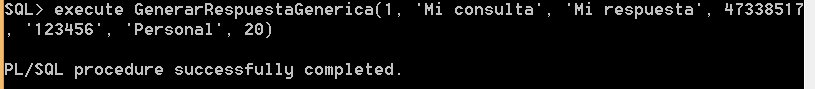
**INSERT INTO** EMPRESA **VALUES** (**'123456'**, **'Mi Empresa SA'**, **'miempresa@gmail.com'**, **'Avda Italia 1234'**, **'Juan Perez'**, **'099334455'**);  
  
**INSERT INTO** ESPECIALIDAD (**DESCRIPCION**) **VALUES** (**'Funcional'**);  
  
**INSERT INTO** TIPO\_CONSULTA (**DESCRIPCION**, **ID\_ESPECIALIDAD**) **VALUES** (**'Tipo F'**, (**SELECT ID FROM** ESPECIALIDAD **WHERE** ESPECIALIDAD.**DESCRIPCION** = **'Funcional'**));  
  
**INSERT INTO** FUNCIONARIO (**DOCUMENTO**, **TIPO**, **PAIS**, **DIRECCION**, **NOMBRE**, **FECHA\_NACIMIENTO**, **FECHA\_INGRESO**)  
**VALUES** (47338517, **'Operario'**, **'Uruguay'**, **'Cuareim 1122'**, **'Juan Rodriguez'**, **'02/09/1996'**, **'08/08/2016'**);  
  
**INSERT INTO** PRODUCTO (**CODIGO**, **NOMBRE**, **DESCRIPCION**) **VALUES** (111, **'A'**, **'Producto A'**);  
**INSERT INTO** PRODUCTO (**CODIGO**, **NOMBRE**, **DESCRIPCION**) **VALUES** (222, **'B'**, **'Producto B'**);  
  
**INSERT INTO** IDS\_PRODUCTOS\_RESP\_GEN (**ID\_PRODUCTO**) **VALUES** (111);  
**INSERT INTO** IDS\_PRODUCTOS\_RESP\_GEN (**ID\_PRODUCTO**) **VALUES** (222);

La ejecución del procedimiento *GenerarRespuestaGenerica* debería resultar en la creación de una consulta y una respuesta *generalizable* en donde ambos registros tienen que estar relacionados.

Así mismo, al existir registros en la tabla auxiliar ***Ids\_Productos\_Resp\_Gen*** la consulta tiene que estar relacionada con los *ids* de los productos insertados en dicha tabla.

Al estar la tabla inicialmente vacía el valor de los ids de los registros insertados, como el tipo de consulta y la especialidad será **1**.

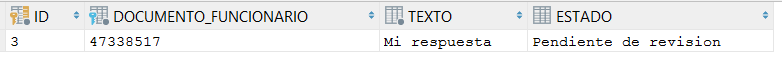
Ejecutamos el procedimiento



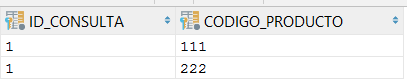
Luego ejecutamos

**select** *\** **from** RESPUESTA;  
**select** *\** **from** CONSULTA;  
**select** *\** **from** CONSULTA\_PRODUCTO;

Obtenemos







Verificamos que

1. Se crea una consulta con los datos pertinentes
2. Se crea una respuesta *generalizable* y la consulta anterior está asociada a la misma
3. La consulta creada está asociada a los productos

**Procedimiento 2**

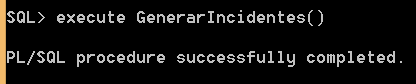
Dados los siguientes datos de prueba

**INSERT INTO** EMPRESA **VALUES** (**'123456'**, **'Mi Empresa SA'**, **'miempresa@gmail.com'**, **'Avda Italia 1234'**, **'Juan Perez'**, **'099334455'**);  
  
**INSERT INTO** ESPECIALIDAD (**DESCRIPCION**) **VALUES** (**'Funcional'**);  
  
**INSERT INTO** TIPO\_CONSULTA (**DESCRIPCION**, **ID\_ESPECIALIDAD**) **VALUES** (**'Tipo F'**, (**SELECT ID FROM** ESPECIALIDAD **WHERE** ESPECIALIDAD.**DESCRIPCION** = **'Funcional'**));  
  
**INSERT INTO** FUNCIONARIO (**DOCUMENTO**, **TIPO**, **PAIS**, **DIRECCION**, **NOMBRE**, **FECHA\_NACIMIENTO**, **FECHA\_INGRESO**)  
**VALUES** (47338517, **'Operario'**, **'Uruguay'**, **'Cuareim 1122'**, **'Juan Rodriguez'**, **'02/09/1996'**, **'08/08/2016'**);  
  
**INSERT INTO** CONSULTA (**ID\_TIPO\_CONSULTA**, **RUT\_EMPRESA**, **DETALLE**) **VALUES** ((**SELECT ID FROM** TIPO\_CONSULTA **WHERE DESCRIPCION** = **'Tipo F'**), **'123456'**, **'Mi consulta'**);  
  
**UPDATE** CONSULTA **SET FECHA\_CREACION** = **'28/11/2017' WHERE ID** = (**SELECT** *MAX*(**ID**) **FROM** CONSULTA);

La ejecución del procedimiento *GenerarIncidentes* debería resultar en la creación de un incidente nuevo en estado **Pendiente** si el procedimiento encuentra consultas ingresadas hace más de 24hs sin ser respondidas.

En este caso ingresamos una consulta, para simular que fue ingresada hace más de 24hs actualizamos el atributo **FECHA\_CREACION** dado que al ingresar un registro el trigger fecha\_creacion automáticamente le asigna la fecha actual del sistema.

Ejecutamos el procedimiento



Y consultamos los resultados

**select** *\** **from** CONSULTA;  
**select** *\** **from** INCIDENTE;

Obteniendo





Verificamos que se crea un Incidente con estado *Pendiente* para la consulta dada.

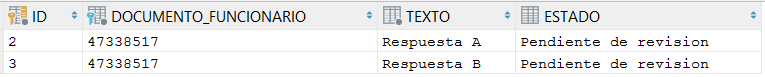
**Procedimiento 3**

Dados los siguientes datos de prueba

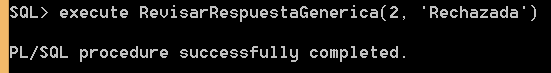
**INSERT INTO** FUNCIONARIO (**DOCUMENTO**, **TIPO**, **PAIS**, **DIRECCION**, **NOMBRE**, **FECHA\_NACIMIENTO**, **FECHA\_INGRESO**)  
**VALUES** (47338517, **'Operario'**, **'Uruguay'**, **'Cuareim 1122'**, **'Juan Rodriguez'**, **'02/09/1996'**, **'08/08/2016'**);  
  
**INSERT INTO** RESPUESTA (**DOCUMENTO\_FUNCIONARIO**, **TEXTO**, **ESTADO**) **VALUES** (47338517, **'Respuesta A'**, **'Pendiente de revision'**);  
**INSERT INTO** RESPUESTA (**DOCUMENTO\_FUNCIONARIO**, **TEXTO**, **ESTADO**) **VALUES** (47338517, **'Respuesta B'**, **'Pendiente de revision'**);  
  
**INSERT INTO** PALABRA\_CLAVE (**PALABRA**) **VALUES** (**'Palabra 1'**);  
**INSERT INTO** PALABRA\_CLAVE (**PALABRA**) **VALUES** (**'Palabra 2'**);  
  
**INSERT INTO** PALABRAS\_CLAVE\_PARA\_GENERICA (**ID\_PALABRA\_CLAVE**) **VALUES** ((**SELECT ID FROM** PALABRA\_CLAVE **WHERE PALABRA** = **'Palabra 1'**));  
**INSERT INTO** PALABRAS\_CLAVE\_PARA\_GENERICA (**ID\_PALABRA\_CLAVE**) **VALUES** ((**SELECT ID FROM** PALABRA\_CLAVE **WHERE PALABRA** = **'Palabra 2'**));

La ejecución del procedimiento *RevisarRespuestaGenerica* debería cambiar el estado de la respuesta *genérica* dada por el estado pasado por parámetro. Si el estado pasado es **Revisada Ok** además debería asignarle las *palabras clave* encontradas en la tabla **PALABRAS\_CLAVE\_PARA\_GENERICA**.

Dadas las respuestas insertadas

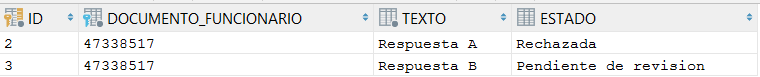


Ejecutamos



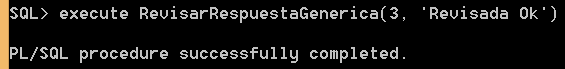
Seleccionamos las respuestas

**select** *\** **from** RESPUESTA;



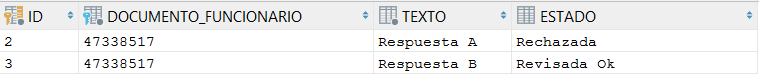
Y verificamos que la respuesta con id **2** cambió su estado a *Rechazada*.

Por último, ejecutamos



Seleccionamos las respuestas

**select** *\** **from** RESPUESTA;

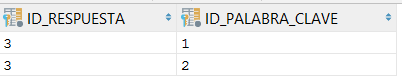


Y verificamos que la respuesta con id **3** cambió su estado a *Revisada Ok*.

Por último ejecutamos

**select** *\** **from** PALABRA\_CLAVE\_RESPUESTA;

obteniendo



Y verificamos que las palabras clave fueron asignadas a la respuesta.

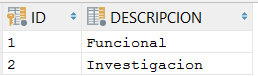
**Procedimiento 4**

Dados los siguientes datos de prueba

**INSERT INTO** EMPRESA **VALUES** (**'123456'**, **'Mi Empresa SA'**, **'miempresa@gmail.com'**, **'Avda Italia 1234'**, **'Juan Perez'**, **'099334455'**);  
  
**INSERT INTO** ESPECIALIDAD (**DESCRIPCION**) **VALUES** (**'Funcional'**);  
**INSERT INTO** ESPECIALIDAD (**DESCRIPCION**) **VALUES** (**'Investigacion'**);  
  
**INSERT INTO** FUNCIONARIO (**DOCUMENTO**, **TIPO**, **PAIS**, **DIRECCION**, **NOMBRE**, **FECHA\_NACIMIENTO**, **FECHA\_INGRESO**)  
**VALUES** (47338517, **'Operario'**, **'Uruguay'**, **'Cuareim 1122'**, **'Funcionario 1'**, **'02/09/1996'**, **'08/08/2016'**);  
  
**INSERT INTO** FUNCIONARIO (**DOCUMENTO**, **TIPO**, **PAIS**, **DIRECCION**, **NOMBRE**, **FECHA\_NACIMIENTO**, **FECHA\_INGRESO**)  
**VALUES** (47338516, **'Operario'**, **'Uruguay'**, **'Cuareim 1122'**, **'Funcionario 2'**, **'02/09/1996'**, **'08/08/2016'**);  
  
**INSERT INTO** FUNCIONARIO (**DOCUMENTO**, **TIPO**, **PAIS**, **DIRECCION**, **NOMBRE**, **FECHA\_NACIMIENTO**, **FECHA\_INGRESO**)  
**VALUES** (47338515, **'Operario'**, **'Uruguay'**, **'Cuareim 1122'**, **'Funcionario 3'**, **'02/09/1996'**, **'08/08/2016'**);  
  
**INSERT INTO** FUNCIONARIO\_ESPECIALIDAD (**DOCUMENTO\_FUNCIONARIO**, **ID\_ESPECIALIDAD**) **VALUES** (47338517, (**SELECT ID FROM** ESPECIALIDAD **WHERE DESCRIPCION** = **'Funcional'**));  
**INSERT INTO** FUNCIONARIO\_ESPECIALIDAD (**DOCUMENTO\_FUNCIONARIO**, **ID\_ESPECIALIDAD**) **VALUES** (47338516, (**SELECT ID FROM** ESPECIALIDAD **WHERE DESCRIPCION** = **'Funcional'**));  
**INSERT INTO** FUNCIONARIO\_ESPECIALIDAD (**DOCUMENTO\_FUNCIONARIO**, **ID\_ESPECIALIDAD**) **VALUES** (47338515, (**SELECT ID FROM** ESPECIALIDAD **WHERE DESCRIPCION** = **'Investigacion'**));  
  
**INSERT INTO** TIPO\_CONSULTA (**DESCRIPCION**, **ID\_ESPECIALIDAD**) **VALUES** (**'Tipo A'**, (**SELECT ID FROM** ESPECIALIDAD **WHERE DESCRIPCION** = **'Funcional'**));  
  
**INSERT INTO** RESPUESTA (**DOCUMENTO\_FUNCIONARIO**, **TEXTO**, **ESTADO**) **VALUES** (47338517, **'Respuesta A'**, **'Revisada Ok'**);  
**INSERT INTO** RESPUESTA (**DOCUMENTO\_FUNCIONARIO**, **TEXTO**, **ESTADO**) **VALUES** (47338517, **'Respuesta B'**, **'Revisada Ok'**);  
**INSERT INTO** RESPUESTA (**DOCUMENTO\_FUNCIONARIO**, **TEXTO**, **ESTADO**) **VALUES** (47338517, **'Respuesta C'**, **'Revisada Ok'**);  
  
**INSERT INTO** CONSULTA (**ID\_TIPO\_CONSULTA**, **RUT\_EMPRESA**, **DETALLE**, **ID\_RESPUESTA**)  
**VALUES** ((**SELECT** *MAX*(**ID**) **FROM** TIPO\_CONSULTA), **'123456'**, **'Consulta 1'**, (**SELECT ID FROM** RESPUESTA **WHERE TEXTO** = **'Respuesta A'**));  
  
**INSERT INTO** CONSULTA (**ID\_TIPO\_CONSULTA**, **RUT\_EMPRESA**, **DETALLE**, **ID\_RESPUESTA**)  
**VALUES** ((**SELECT** *MAX*(**ID**) **FROM** TIPO\_CONSULTA), **'123456'**, **'Consulta 2'**, (**SELECT ID FROM** RESPUESTA **WHERE TEXTO** = **'Respuesta B'**));  
  
**INSERT INTO** CONSULTA (**ID\_TIPO\_CONSULTA**, **RUT\_EMPRESA**, **DETALLE**, **ID\_RESPUESTA**)  
**VALUES** ((**SELECT** *MAX*(**ID**) **FROM** TIPO\_CONSULTA), **'123456'**, **'Consulta 3'**, (**SELECT ID FROM** RESPUESTA **WHERE TEXTO** = **'Respuesta C'**));  
  
**INSERT INTO** PALABRA\_CLAVE (**PALABRA**) **VALUES** (**'Palabra 1'**);  
**INSERT INTO** PALABRA\_CLAVE (**PALABRA**) **VALUES** (**'Palabra 2'**);  
  
**INSERT INTO** PALABRA\_CLAVE\_RESPUESTA (**ID\_RESPUESTA**, **ID\_PALABRA\_CLAVE**) **VALUES** ((**SELECT ID FROM** RESPUESTA **WHERE TEXTO** = **'Respuesta A'**), (**SELECT ID FROM** PALABRA\_CLAVE **WHERE PALABRA** = **'Palabra 1'**));  
**INSERT INTO** PALABRA\_CLAVE\_RESPUESTA (**ID\_RESPUESTA**, **ID\_PALABRA\_CLAVE**) **VALUES** ((**SELECT ID FROM** RESPUESTA **WHERE TEXTO** = **'Respuesta A'**), (**SELECT ID FROM** PALABRA\_CLAVE **WHERE PALABRA** = **'Palabra 2'**));  
  
**INSERT INTO** PALABRA\_CLAVE\_RESPUESTA (**ID\_RESPUESTA**, **ID\_PALABRA\_CLAVE**) **VALUES** ((**SELECT ID FROM** RESPUESTA **WHERE TEXTO** = **'Respuesta B'**), (**SELECT ID FROM** PALABRA\_CLAVE **WHERE PALABRA** = **'Palabra 1'**));  
**INSERT INTO** PALABRA\_CLAVE\_RESPUESTA (**ID\_RESPUESTA**, **ID\_PALABRA\_CLAVE**) **VALUES** ((**SELECT ID FROM** RESPUESTA **WHERE TEXTO** = **'Respuesta B'**), (**SELECT ID FROM** PALABRA\_CLAVE **WHERE PALABRA** = **'Palabra 2'**));  
  
**INSERT INTO** PALABRA\_CLAVE\_RESPUESTA (**ID\_RESPUESTA**, **ID\_PALABRA\_CLAVE**) **VALUES** ((**SELECT ID FROM** RESPUESTA **WHERE TEXTO** = **'Respuesta C'**), (**SELECT ID FROM** PALABRA\_CLAVE **WHERE PALABRA** = **'Palabra 1'**));  
  
**INSERT INTO** PALABRA\_CLAVE\_PROCEDIMIENTO **VALUES**((**SELECT ID FROM** PALABRA\_CLAVE **WHERE PALABRA** = **'Palabra 1'**));  
**INSERT INTO** PALABRA\_CLAVE\_PROCEDIMIENTO **VALUES**((**SELECT ID FROM** PALABRA\_CLAVE **WHERE PALABRA** = **'Palabra 2'**));

La ejecución del procedimiento *RespuestasGenericas* deberá devolver los funcionarios especializados en el tipo de consulta dado y las respuestas con las palabras clave insertadas en la tabla **PALABRA\_CLAVE\_PROCEDIMIENTO**.

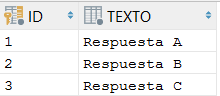
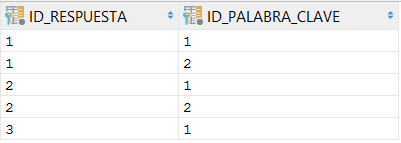
Siendo las especialidades y los tipos de consulta



Si ejecutamos el procedimiento con el id **1** deberá devolver los funcionarios **Funcionario 1** y **Funcionario 2** que son especialistas en **Funcional**, que está asociado al **Tipo A** de consulta.

Por otra parte se deberán devolver las respuestas con las siguientes palabras clave

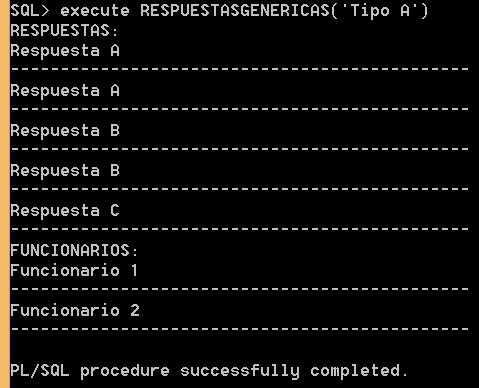




En este caso tanto **A**, **B** y **C** tienen asignada al menos una de las palabras clave.

**Importante:** ejecutar **SET SERVEROUTPUT ON;** para visualizar la salida de este procedimiento.

Si ejecutamos



Tenemos el resultado esperado.

**Procedimiento 5**

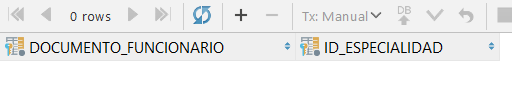
**Importante:** Tomamos la cantidad de respuestas exitosas como mayor a 2 en lugar de mayor a 20 para ejecutar los datos de prueba, de forma de que sea más entendible el script de datos.

Dados los siguientes datos de prueba

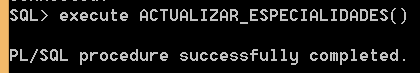
**INSERT INTO** EMPRESA **VALUES** (**'123456'**, **'Mi Empresa SA'**, **'miempresa@gmail.com'**, **'Avda Italia 1234'**, **'Juan Perez'**, **'099334455'**);  
  
**INSERT INTO** FUNCIONARIO (**DOCUMENTO**, **TIPO**, **PAIS**, **DIRECCION**, **NOMBRE**, **FECHA\_NACIMIENTO**, **FECHA\_INGRESO**)  
**VALUES** (47338517, **'Operario'**, **'Uruguay'**, **'Cuareim 1122'**, **'Funcionario 1'**, **'02/09/1996'**, **'08/08/2016'**);  
  
**INSERT INTO** ESPECIALIDAD (**DESCRIPCION**) **VALUES** (**'Funcional'**);  
  
**INSERT INTO** TIPO\_CONSULTA (**DESCRIPCION**, **ID\_ESPECIALIDAD**) **VALUES** (**'Tipo A'**, (**SELECT ID FROM** ESPECIALIDAD **WHERE DESCRIPCION** = **'Funcional'**));  
  
**INSERT INTO** RESPUESTA (**DOCUMENTO\_FUNCIONARIO**, **TEXTO**, **ESTADO**) **VALUES** (47338517, **'Respuesta A'**, **'Revisada Ok'**);  
  
**INSERT INTO** CONSULTA (**ID\_TIPO\_CONSULTA**, **RUT\_EMPRESA**, **DETALLE**, **ID\_RESPUESTA**)  
**VALUES** ((**SELECT** *MAX*(**ID**) **FROM** TIPO\_CONSULTA), **'123456'**, **'Consulta 1'**, (**SELECT ID FROM** RESPUESTA **WHERE TEXTO** = **'Respuesta A'**));  
  
**INSERT INTO** CONSULTA (**ID\_TIPO\_CONSULTA**, **RUT\_EMPRESA**, **DETALLE**, **ID\_RESPUESTA**)  
**VALUES** ((**SELECT** *MAX*(**ID**) **FROM** TIPO\_CONSULTA), **'123456'**, **'Consulta 2'**, (**SELECT ID FROM** RESPUESTA **WHERE TEXTO** = **'Respuesta A'**));  
  
**INSERT INTO** CONSULTA (**ID\_TIPO\_CONSULTA**, **RUT\_EMPRESA**, **DETALLE**, **ID\_RESPUESTA**)  
**VALUES** ((**SELECT** *MAX*(**ID**) **FROM** TIPO\_CONSULTA), **'123456'**, **'Consulta 3'**, (**SELECT ID FROM** RESPUESTA **WHERE TEXTO** = **'Respuesta A'**));

La ejecución del procedimiento *ACTUALIZAR\_ESPECIALIDADES* deberá asignarle al **Funcionario 1** la especialidad **Tipo A** si la cantidad de consultas exitosas del mes corriente es mayor a 2.

Inicialmente el funcionario no tiene especialidades



Si ejecutamos el procedimiento



Y luego consultamos las especialidades del funcionario

**select** *\** **from** FUNCIONARIO\_ESPECIALIDAD



Verificamos que se le asignó la especialidad **1**



Que es la asignada al **Tipo A**