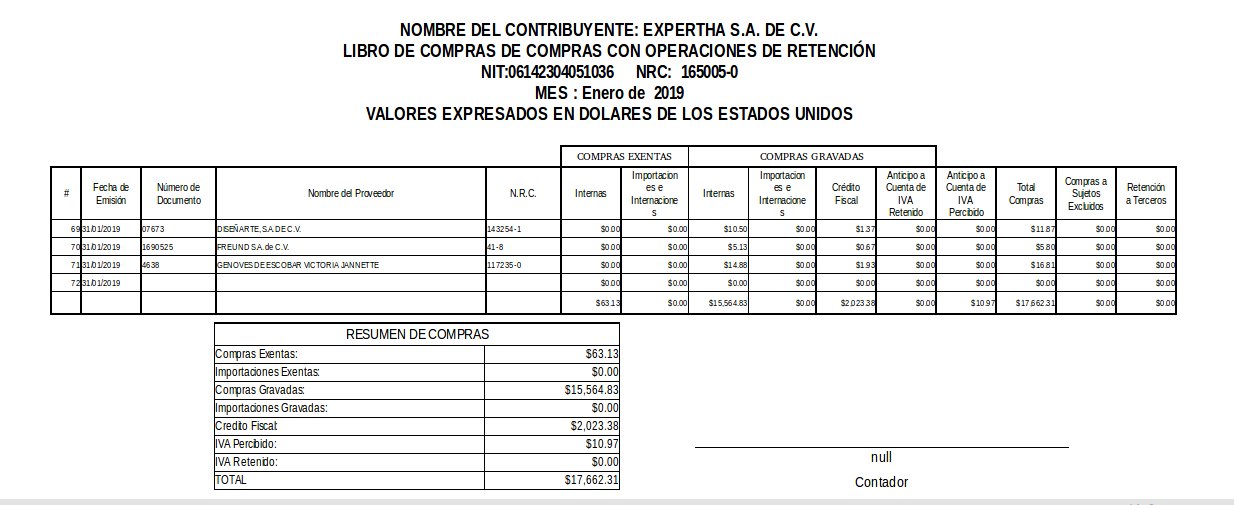
Reports.

# Tax Report 1

## Format



## SQL Query-Header

select c.name as Compania

,c.sv\_nit as NIT

,c.sv\_nrc as NRC

,sv\_revisa\_partida as Contador

from res\_company c

where c.id= $P{companyId}

## SQL Query-Detail

select \* from (

select ai.date\_invoice as fecha

,ai.reference as factura

,rp.name as proveedor

,rp.vat as NRC

,case

when rp.country\_id=211 then False

when rp.country\_id is null then False

else True end as Importacion

,/\*Calculando el gravado (todo lo que tiene un impuesto aplicado de iva)\*/

(select coalesce(sum(price\_subtotal\_signed),0.00)

from account\_invoice\_line ail

where invoice\_id=ai.id

and exists(select ailt.tax\_id

from account\_invoice\_line\_tax ailt

inner join account\_tax atx on ailt.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where ailt.invoice\_line\_id=ail.id and atg.name='iva')

) as Gravado,

/\*Calculando el excento que no tiene iva\*/

(Select coalesce(sum(price\_subtotal\_signed),0.00)

from account\_invoice\_line ail

where invoice\_id=ai.id

and not exists(select ailt.tax\_id

from account\_invoice\_line\_tax ailt

inner join account\_tax atx on ailt.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where ailt.invoice\_line\_id=ail.id and atg.name='iva')

) as Exento

,/\*Calculando el iva\*/

(Select coalesce(sum(ait.amount),0.00)

from account\_invoice\_tax ait

inner join account\_tax atx on ait.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where invoice\_id=ai.id

and atg.name='iva'

) as Iva

,/\*Calculando el retenido\*/

(Select coalesce(sum(ait.amount),0.00)

from account\_invoice\_tax ait

inner join account\_tax atx on ait.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where invoice\_id=ai.id

and atg.name='retencion'

) as Retenido

,/\*Calculando el percibido\*/

(Select coalesce(sum(ait.amount),0.00)

from account\_invoice\_tax ait

inner join account\_tax atx on ait.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where invoice\_id=ai.id

and atg.name='percepcion'

) as Percibido

,/\*Calculando el excluido\*/

(Select coalesce(sum(ait.amount),0.00)

from account\_invoice\_tax ait

inner join account\_tax atx on ait.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where invoice\_id=ai.id

and atg.name='excluido'

) as Excluido

,/\*Calculando el retencion a terceros\*/

(Select coalesce(sum(ait.amount),0.00)

from account\_invoice\_tax ait

inner join account\_tax atx on ait.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where invoice\_id=ai.id

and atg.name='retencion3'

) as retencion3

from account\_invoice ai

inner join res\_partner rp on ai.partner\_id=rp.id

where ai.company\_id= $P{companyId}

and date\_part('year',COALESCE(ai.sv\_fecha\_tax,ai.date\_invoice))= $P{year}

and date\_part('month',COALESCE(ai.sv\_fecha\_tax,ai.date\_invoice))= $P{month}

and ai.type='in\_invoice'

and ((ai.sv\_no\_tax is null ) or (ai.sv\_no\_tax=false))

and ai.state in ('open','paid')

and ((ai.sv\_importacion\_number is null) or (trim(ai.sv\_importacion\_number)=''))

union all

select min(SI.fecha) as fecha

,SI.sv\_importacion\_number as factura

,'DGT' as proveedor

,'DGT' as NRC

,True as Importacion

,sum (SI.Gravado) as Gravado

,sum (SI.Exento) as Exento

,sum (SI.Iva) as Iva

,sum (SI.Retenido) as Retenido

,sum (SI.Percibido) as Percibido

,sum (SI.Excluido) as Excluido

,sum (SI.retencion3) as retencion3

from (

select ai.date\_invoice as fecha

,ai.reference as factura

,ai.sv\_importacion\_number

,rp.name as proveedor

,rp.vat as NRC

,case

when rp.country\_id=211 then False

when rp.country\_id is null then False

else True end as Importacion

,/\*Calculando el gravado (todo lo que tiene un impuesto aplicado de iva)\*/

(select coalesce(sum(price\_subtotal\_signed),0.00)

from account\_invoice\_line ail

where invoice\_id=ai.id

and exists(select ailt.tax\_id

from account\_invoice\_line\_tax ailt

inner join account\_tax atx on ailt.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where ailt.invoice\_line\_id=ail.id and atg.name='iva')

) as Gravado,

/\*Calculando el excento que no tiene iva\*/

(Select coalesce(sum(price\_subtotal\_signed),0.00)

from account\_invoice\_line ail

where invoice\_id=ai.id

and not exists(select ailt.tax\_id

from account\_invoice\_line\_tax ailt

inner join account\_tax atx on ailt.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where ailt.invoice\_line\_id=ail.id and atg.name='iva')

) as Exento

,/\*Calculando el iva\*/

(Select coalesce(sum(ait.amount),0.00)

from account\_invoice\_tax ait

inner join account\_tax atx on ait.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where invoice\_id=ai.id

and atg.name='iva'

) as Iva

,/\*Calculando el retenido\*/

(Select coalesce(sum(ait.amount),0.00)

from account\_invoice\_tax ait

inner join account\_tax atx on ait.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where invoice\_id=ai.id

and atg.name='retencion'

) as Retenido

,/\*Calculando el percibido\*/

(Select coalesce(sum(ait.amount),0.00)

from account\_invoice\_tax ait

inner join account\_tax atx on ait.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where invoice\_id=ai.id

and atg.name='percepcion'

) as Percibido

,/\*Calculando el excluido\*/

(Select coalesce(sum(ait.amount),0.00)

from account\_invoice\_tax ait

inner join account\_tax atx on ait.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where invoice\_id=ai.id

and atg.name='excluido'

) as Excluido

,/\*Calculando el retencion a terceros\*/

(Select coalesce(sum(ait.amount),0.00)

from account\_invoice\_tax ait

inner join account\_tax atx on ait.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where invoice\_id=ai.id

and atg.name='retencion3'

) as retencion3

from account\_invoice ai

inner join res\_partner rp on ai.partner\_id=rp.id

where ai.company\_id= $P{companyId}

and date\_part('year',COALESCE(ai.sv\_fecha\_tax,ai.date\_invoice))= $P{year}

and date\_part('month',COALESCE(ai.sv\_fecha\_tax,ai.date\_invoice))= $P{month}

and ai.type='in\_invoice'

and ((ai.sv\_no\_tax is null ) or (ai.sv\_no\_tax=false))

and ai.state in ('open','paid')

and ((ai.sv\_importacion\_number is not null) or (trim(ai.sv\_importacion\_number)!=''))

) SI

group by SI.sv\_importacion\_number

union all

select aml.date as fecha

,am.ref as factura

,rp.name as proveedor

,rp.vat as NRC

,False as Importacion

,0.0 as Gravado

,0.0 as Exento

,0.0 as Iva

,0.0 as Retenido

,aml.debit as Percibido

,0.0 as Excluido

,0.0 as retencion3

from account\_move\_line aml

inner join account\_move am on aml.move\_id=am.id

inner join account\_tax at on aml.account\_id=at.account\_id

inner join account\_tax\_group atg on at.tax\_group\_id=atg.id

left join res\_partner rp on aml.partner\_id=rp.id

where atg.name='percepcion' and not exists (select id from account\_invoice ai where ai.move\_id=am.id and ai.company\_id= $P{companyId} )

and date\_part('year',am.date)= $P{year}

and date\_part('month',am.date)= $P{month}

and am.company\_id= $P{companyId}

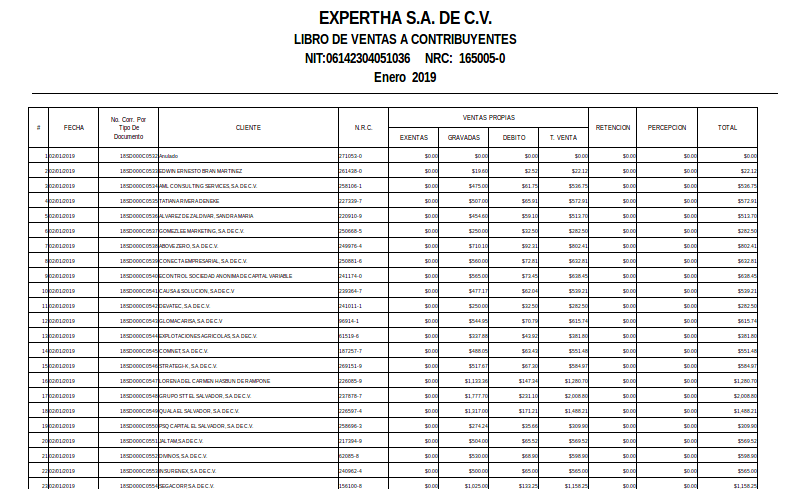
and am.state='posted'

) S

order by s.Fecha, s.Factura

# TAX Report 2

## Format



## SQL Query-Header

select c.name as Compania

,c.sv\_nit as NIT

,c.sv\_nrc as NRC

,c.sv\_revisa\_partida as Contador

from res\_company c

where c.id= $P{companyId}

## SQL Query-Detail

select \* from(

select ai.date\_invoice as fecha

,ai.reference as factura

,rp.name as cliente

,rp.vat as NRC

,ai.state as estado

,/\*Calculando el gravado (todo lo que tiene un impuesto aplicado de iva)\*/

(select coalesce(sum(price\_subtotal\_signed),0.00)

from account\_invoice\_line ail

where invoice\_id=ai.id

and exists(select ailt.tax\_id

from account\_invoice\_line\_tax ailt

inner join account\_tax atx on ailt.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where ailt.invoice\_line\_id=ail.id and atg.name='iva')

) as Gravado,

/\*Calculando el excento que no tiene iva\*/

(Select coalesce(sum(price\_subtotal\_signed),0.00)

from account\_invoice\_line ail

where invoice\_id=ai.id

and not exists(select ailt.tax\_id

from account\_invoice\_line\_tax ailt

inner join account\_tax atx on ailt.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where ailt.invoice\_line\_id=ail.id and atg.name='iva')

) as Exento

,/\*Calculando el iva\*/

(Select coalesce(sum(ait.amount),0.00)

from account\_invoice\_tax ait

inner join account\_tax atx on ait.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where invoice\_id=ai.id

and atg.name='iva'

)\*(case when ai.type='out\_refund' then -1 else 1 end) as Iva

,/\*Calculando el retenido\*/

(Select coalesce(sum(ait.amount),0.00)

from account\_invoice\_tax ait

inner join account\_tax atx on ait.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where invoice\_id=ai.id

and atg.name='retencion'

) as Retenido

,/\*Calculando el percibido\*/

(Select coalesce(sum(ait.amount),0.00)

from account\_invoice\_tax ait

inner join account\_tax atx on ait.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where invoice\_id=ai.id

and atg.name='percepcion'

) as Percibido

from account\_invoice ai

inner join res\_partner rp on ai.partner\_id=rp.id

left join account\_fiscal\_position afp on ai.fiscal\_position\_id=afp.id

where ai.company\_id= $P{companyId}

and date\_part('year',COALESCE(ai.sv\_fecha\_tax,ai.date\_invoice))= $P{year}

and date\_part('month',COALESCE(ai.sv\_fecha\_tax,ai.date\_invoice))= $P{month}

and ((ai.type='out\_invoice') or (ai.type='out\_refund'))

and ((ai.sv\_no\_tax is null ) or (ai.sv\_no\_tax=false))

and afp.sv\_contribuyente=True

and ai.state in ('open','paid')

union

select ai.date\_invoice as fecha

,ai.reference as factura

,'Anulado' as cliente

,rp.vat as NRC

,ai.state as estado

,0.0 as Gravado

,0.0 as Exento

,0.0 as Iva

,0.0 as Retenido

,0.0 as Percibido

from account\_invoice ai

inner join res\_partner rp on ai.partner\_id=rp.id

left join account\_fiscal\_position afp on ai.fiscal\_position\_id=afp.id

where ai.company\_id= $P{companyId}

and date\_part('year',COALESCE(ai.sv\_fecha\_tax,ai.date\_invoice))= $P{year}

and date\_part('month',COALESCE(ai.sv\_fecha\_tax,ai.date\_invoice))= $P{month}

and ai.type='out\_invoice'

and ((ai.sv\_no\_tax is null ) or (ai.sv\_no\_tax=false))

and afp.sv\_contribuyente=True

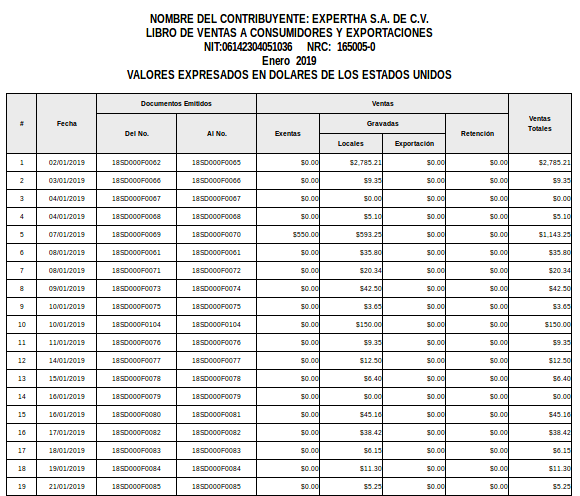
and ai.state in ('cancel')

)S

order by s.fecha, s.factura

# TAX Report 3

Format



## SQL Query-Header

select c.name as Compania

,c.sv\_nit as NIT

,c.sv\_nrc as NRC

,c.sv\_invoice\_serie\_size as Serie

,sv\_revisa\_partida as Contador

from res\_company c

where c.id= $P{companyId}

## SQL Query-Detail

--select \* from res\_company

Select

SS.Fecha

,SS.grupo

,min(SS.Factura) as DELNum

,max(SS.Factura) as ALNum

,sum(SS.exento) as Exento

,sum(SS.GravadoLocal) as GravadoLocal

,sum(SS.GravadoExportacion) as GravadoExportacion

,Sum(SS.ivaLocal) as IvaLocal

,Sum(SS.ivaexportacion) as IvaExportacion

,Sum(SS.retenido) as Retenido

,estado

FROM (

select S.fecha

,S.factura

,S.estado

,S.grupo

,S.exento

,case

when S.sv\_region='Local' then S.Gravado

else 0.00 end as GravadoLocal

,case

when S.sv\_region!='Local' then S.Gravado

else 0.00 end as GravadoExportacion

,case

when S.sv\_region='Local' then S.Iva

else 0.00 end as IvaLocal

,case

when S.sv\_region!='Local' then S.Iva

else 0.00 end as IvaExportacion

,S.Retenido

from(

select ai.date\_invoice as fecha

,coalesce(ai.reference,ai.number) as factura

,'valid' as estado

,FG.grupo

,afp.sv\_region

,/\*Calculando el gravado (todo lo que tiene un impuesto aplicado de iva)\*/

(select coalesce(sum(price\_subtotal\_signed),0.00)

from account\_invoice\_line ail

where invoice\_id=ai.id

and exists(select ailt.tax\_id

from account\_invoice\_line\_tax ailt

inner join account\_tax atx on ailt.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where ailt.invoice\_line\_id=ail.id and atg.name='iva')

) as Gravado,

/\*Calculando el excento que no tiene iva\*/

(Select coalesce(sum(price\_subtotal\_signed),0.00)

from account\_invoice\_line ail

where invoice\_id=ai.id

and not exists(select ailt.tax\_id

from account\_invoice\_line\_tax ailt

inner join account\_tax atx on ailt.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where ailt.invoice\_line\_id=ail.id and atg.name='iva')

) as Exento

,/\*Calculando el iva\*/

(Select coalesce(sum(ait.amount),0.00)

from account\_invoice\_tax ait

inner join account\_tax atx on ait.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where invoice\_id=ai.id

and atg.name='iva'

) as Iva

,/\*Calculando el retenido\*/

(Select coalesce(sum(ait.amount),0.00)

from account\_invoice\_tax ait

inner join account\_tax atx on ait.tax\_id=atx.id

inner join account\_tax\_group atg on atx.tax\_group\_id=atg.id

where invoice\_id=ai.id

and atg.name='retencion'

) as Retenido

from account\_invoice ai

inner join res\_partner rp on ai.partner\_id=rp.id

inner join (select \* from FacturasAgrupadas( $P{companyId}, $P{month} , $P{year} , $P{serie} )) FG on ai.id=FG.invoice\_id

left join account\_fiscal\_position afp on ai.fiscal\_position\_id=afp.id

where ai.company\_id= $P{companyId}

and date\_part('year',COALESCE(ai.sv\_fecha\_tax,ai.date\_invoice))= $P{year}

and date\_part('month',COALESCE(ai.sv\_fecha\_tax,ai.date\_invoice))= $P{month}

and ai.type='out\_invoice'

and ((ai.sv\_no\_tax is null ) or (ai.sv\_no\_tax=false))

and afp.sv\_contribuyente=False

and ai.state in ('open','paid')

union

select ai.date\_invoice as fecha

,coalesce(ai.reference,ai.number) as factura

,ai.state as estado

,FG.grupo

,afp.sv\_region

,0.0 as Gravado

,0.0 as Exento

,0.0 as Iva

,0.0 as Retenido

from account\_invoice ai

inner join res\_partner rp on ai.partner\_id=rp.id

inner join (select \* from FacturasAgrupadas( $P{companyId} , $P{month}, $P{year} , $P{serie})) FG on ai.id=FG.invoice\_id

left join account\_fiscal\_position afp on ai.fiscal\_position\_id=afp.id

where ai.company\_id= $P{companyId}

and date\_part('year',COALESCE(ai.sv\_fecha\_tax,ai.date\_invoice))= $P{year}

and date\_part('month',COALESCE(ai.sv\_fecha\_tax,ai.date\_invoice))= $P{month}

and ai.type='out\_invoice'

and ((ai.sv\_no\_tax is null ) or (ai.sv\_no\_tax=false))

and afp.sv\_contribuyente=False

and ai.state in ('cancel')

)S

)SS

group by SS.fecha, SS.Grupo,SS.estado

order by SS.fecha, SS.Grupo

## FUNCTION **facturasagrupadas**. Required by SQL QUERY-Detail

CREATE OR REPLACE FUNCTION public.facturasagrupadas(p\_company\_id integer, month\_number integer, year\_number integer, p\_series\_lenght integer)

RETURNS TABLE(invoice\_id integer, factura\_number character varying, factura\_status character varying, grupo integer)

LANGUAGE plpgsql

AS $function$

DECLARE

var\_r record;

var\_serie varchar;

var\_fecha date;

var\_correlativo int;

var\_estado varchar;

var\_grupo int;

BEGIN

var\_grupo :=0;

FOR var\_r IN (select ROW\_NUMBER () OVER (ORDER BY f.date\_invoice,coalesce(F.reference,F.number)) as Registro

,left(coalesce(F.reference,F.number),p\_series\_lenght) as Serie

,cast(right(coalesce(F.reference,F.number),(length(coalesce(F.reference,F.number))-p\_series\_lenght)) as Integer) as correlativo

,F.date\_invoice as fecha

,case

when F.state='cancel' then 'ANULADA'

else 'Valida' end as estado

,coalesce(F.reference,F.number) as factura,F.id

from Account\_invoice F

inner join account\_fiscal\_position afp on F.fiscal\_position\_id=afp.id

where date\_part('year',COALESCE(F.sv\_fecha\_tax,F.date\_invoice))= year\_number

and date\_part('month',COALESCE(F.sv\_fecha\_tax,F.date\_invoice))= month\_number

and F.state<>'draft' and F.company\_id=p\_company\_id

and F.type in ('out\_invoice')

and ((F.sv\_no\_tax is null ) or (F.sv\_no\_tax=false))

and afp.sv\_contribuyente=False

order by fecha,factura )

LOOP

invoice\_id := var\_r.id;

factura\_number := var\_r.Factura;

factura\_status := var\_r.estado;

if ((var\_r.Serie=var\_serie) and (var\_r.fecha=var\_Fecha) and (var\_r.estado=var\_estado) and (var\_r.correlativo=(var\_correlativo+1))) then

grupo := var\_grupo;

else

var\_grupo := var\_grupo+1;

grupo := var\_grupo;

end if;

var\_serie := var\_r.Serie;

var\_fecha := var\_r.fecha;

var\_estado := var\_r.estado;

var\_correlativo := var\_r.correlativo;

RETURN NEXT;

END LOOP;

END;

$function$

;