1. La empresa busca conocer su situación en cuanto a ventas. Para esto, hace un análisis de mercado y desea conocer alguna forma de obtener mayores ganancias. Se desea conocer la información del importe vendido y la cantidad vendida del producto más y menos vendido, por tipo de producto.

De esta forma saber cual es el producto que deja mayores ganancias y cual es el que deja menor ganancia, teniendo como objetivo focalizarse en el producto que deja mayores ganancias y centrarse en la producción de este.

Además, ver cual es el producto que deja menor ganancias y analizar la situación y ver alguna forma de incrementar su rentabilidad o en su defecto retirarlo del mercado.

[Tipo,Producto,Recaudado,Cantidad\_Vendida]

SELECT

ll.\*

FROM

(SELECT

tp.tipo,

prod.producto,

SUM(fdv.cantidad \* fdv.precio\_unitario) Recaudado,

SUM(fdv.cantidad) Cantidad\_Vendido

FROM

Tipo\_Producto tp

INNER JOIN Producto prod ON prod.id\_tipo\_producto = tp.id\_tipo\_producto

INNER JOIN Factura\_Detalle\_Venta fdv ON fdv.id\_producto = prod.id\_producto

INNER JOIN Factura\_Cabecera\_Venta fcv ON fcv.id\_factura\_venta = fdv.id\_factura\_venta

WHERE

fcv.anulada = 0

GROUP BY prod.id\_producto) ll

INNER JOIN

(SELECT

Tipo, MAX(Recaudado) max, MIN(Recaudado) min

FROM

(SELECT

tp.tipo,

prod.producto,

SUM(fdv.cantidad \* fdv.precio\_unitario) Recaudado,

SUM(fdv.cantidad) Cantidad\_Vendido

FROM

Tipo\_Producto tp

INNER JOIN Producto prod ON prod.id\_tipo\_producto = tp.id\_tipo\_producto

INNER JOIN Factura\_Detalle\_Venta fdv ON fdv.id\_producto = prod.id\_producto

INNER JOIN Factura\_Cabecera\_Venta fcv ON fcv.id\_factura\_venta = fdv.id\_factura\_venta

WHERE

fcv.anulada = 0

GROUP BY prod.id\_producto) listProd

GROUP BY Tipo) maxMinTipo ON maxMinTipo.tipo = ll.tipo

AND (ll.Recaudado = maxMinTipo.max

OR ll.Recaudado = maxMinTipo.min);

Por otra parte también ofrecemos otra versión en la cual se muestre la cantidad vendida y el importe por tipo de producto y además muestre cual es el producto con mayor ganancias vendido y cual es el de menos ganancias. De una forma tal que no se tenga que ni comparar los valores de ventas, sino simplemente ver en qué columna se encuentra, por otra parte, brindar información general de los tipos de productos y su situación en ventas.

[Tipo, CantidadVendida, Importe, ProductoMasVendido, ProductoMenosVendido]

SELECT

tp.tipo as Tipo,

sum(fdv.cantidad) as CantidadVendida,

sum(fdv.cantidad\*fdv.precio\_unitario) as Importe,

pm.producto as ProductoMasVendido,

pl.producto as ProductoMenosVendido

FROM

Factura\_Detalle\_Venta fdv

INNER JOIN

Producto p ON fdv.id\_producto=p.id\_producto

INNER JOIN

Tipo\_Producto tp ON p.id\_tipo\_producto=tp.id\_tipo\_producto

INNER JOIN

Factura\_Cabecera\_Venta fcv ON fdv.id\_factura\_venta=fcv.id\_factura\_venta

LEFT JOIN

(SELECT

p.producto as Producto,

tp.id\_tipo\_producto as idTipo,

sum(fdv.cantidad\*fdv.precio\_unitario) as Importe

FROM

Factura\_Detalle\_Venta fdv

INNER JOIN

Factura\_Cabecera\_Venta fcv ON fdv.id\_factura\_venta=fcv.id\_factura\_venta

INNER JOIN

Producto p ON fdv.id\_producto=p.id\_producto

INNER JOIN

Tipo\_Producto tp ON p.id\_tipo\_producto = tp.id\_tipo\_producto

WHERE

fcv.anulada=0

GROUP BY p.producto

ORDER BY Importe desc

LIMIT 1) as pm ON tp.id\_tipo\_producto=pm.idTipo

LEFT JOIN

(SELECT

p.producto as Producto,

tp.id\_tipo\_producto as idTipo,

sum(fdv.cantidad\*fdv.precio\_unitario) as Importe

FROM

Factura\_Detalle\_Venta fdv

INNER JOIN

Factura\_Cabecera\_Venta fcv ON fdv.id\_factura\_venta=fcv.id\_factura\_venta

INNER JOIN

Producto p ON fdv.id\_producto=p.id\_producto

INNER JOIN

Tipo\_Producto tp ON p.id\_tipo\_producto = tp.id\_tipo\_producto

WHERE

fcv.anulada=0

GROUP BY p.producto

ORDER BY Importe asc

LIMIT 1) as pl ON tp.id\_tipo\_producto=pl.idTipo

GROUP BY tp.tipo

ORDER BY tp.tipo;

1. El contador de la empresa Alberto Hernandez desea corroborar los importes de deber y haber de la empresa para el arqueo de caja. Para esto desea ver el total vendido y el total comprado por fecha. Ordenado por fecha.

[fecha, total vendido, total comprado]

SELECT

Fechas.fecha,

TRUNCATE(SUM(fdv.cantidad \* fdv.precio\_unitario),2) AS TotalVendido,

TRUNCATE(SUM(fdc.cantidad \* fdc.precio\_unitario),2) AS TotalComprado

FROM

(SELECT

fcc.fecha

FROM

Factura\_Cabecera\_Compra fcc UNION DISTINCT SELECT

fcv.fecha

FROM

Factura\_Cabecera\_Venta fcv) AS Fechas

LEFT JOIN

(SELECT \* FROM Factura\_Cabecera\_Venta WHERE anulada = 0) fcv ON Fechas.fecha = fcv.fecha

LEFT JOIN

(SELECT \* FROM Factura\_Cabecera\_Compra WHERE anulada = 0) fcc ON Fechas.fecha = fcc.fecha

LEFT JOIN

Factura\_Detalle\_Venta fdv ON fcv.id\_factura\_venta = fdv.id\_factura\_venta

LEFT JOIN

Factura\_Detalle\_Compra fdc ON fcc.id\_factura\_compra = fdc.id\_factura\_compra

WHERE

fcc.anulada = 0 OR fcv.anulada = 0

GROUP BY Fechas.fecha

ORDER BY Fechas.fecha;

1. La empresa está buscando zonas para instalar una nueva sucursal. Para esto desea conocer la información de cuánto se vendió por ciudad para conocer cuál es la que da más beneficios, ordenado por el importe vendido de mayor a menor. Para de esta manera instalar la sucursal en un lugar que asegure la máxima rentabilidad.

[país, provincia, ciudad, vendido]

SELECT

p.pais,

pr.provincia,

c.ciudad, sum(fdv.cantidad\*fdv.precio\_unitario) as Vendido

FROM

Factura\_Cabecera\_Venta fcv

INNER JOIN

Factura\_Detalle\_Venta fdv ON fcv.id\_factura\_venta=fdv.id\_factura\_venta

INNER JOIN

Cliente cl ON fcv.id\_cliente=cl.id\_cliente

INNER JOIN

Ciudad c ON cl.id\_ciudad=c.id\_ciudad

INNER JOIN

Provincia pr ON c.id\_provincia=pr.id\_provincia

INNER JOIN

Pais p ON pr.id\_pais=p.id\_pais

WHERE fcv.anulada=0

GROUP BY c.id\_ciudad

ORDER BY Vendido DESC;

1. La empresa está buscando contratar a más empleados, para esto quiere ver en donde es primordial una mayor cantidad de personal. Desea saber los datos de todos los empleados de la sucursal. Si el empleado es un investigador se desea saber el laboratorio al que está asignado, y en el caso que sea un personal de planta se desea saber el área al cual pertenece. De esta forma podrá ver qué área o que laboratorio necesita más personal.

[nombre, apellido, teléfono, DNI, laboratorio, área]

SELECT

emp.nombre,

emp.apellido,

emp.telefono,

emp.DNI,

lab.uso Laboratorio,

ar.area Area

FROM

Empleado emp

LEFT JOIN

Investigador inv ON inv.id\_empleado = emp.id\_empleado

LEFT JOIN

Personal\_Planta pp ON pp.id\_empleado = emp.id\_empleado

LEFT JOIN

Laboratorio lab ON lab.id\_laboratorio = inv.id\_laboratorio

LEFT JOIN

Area ar ON ar.id\_area = pp.id\_area

ORDER BY ar.id\_area;

1. La empresa desea tener un registro detallado de todo su personal. Se desea conocer toda la información de los empleados, id, datos, su sueldo, rol, titulo, DNI, central sindical (si es que se encuentra afiliado a alguna) y fecha de contratación.

[id, DNI, nombre, apellido, telefono, central\_sindical, rol, titulo, fecha\_contratacion, sueldo]

SELECT

e.id\_empleado as ID,

e.DNI as DNI,

e.nombre as NOMBRE,

e.apellido as APELLIDO,

e.telefono as TELEFONO,

cs.central\_sindical as CENTRAL\_SINDICAL,

ro.rol as ROL,

t.titulo as TITULO,

co.fecha as FECHA\_CONTRATACION,

ro.sueldo as SUELDO

FROM

Empleado e

LEFT JOIN

Central\_Sindical cs ON e.id\_central\_sindical=cs.id\_central\_sindical

INNER JOIN

Responsabilidad re ON e.id\_empleado=re.id\_empleado

INNER JOIN

Rol ro ON re.id\_rol=ro.id\_rol

INNER JOIN

Titulacion ti ON e.id\_empleado=ti.id\_empleado

INNER JOIN

Titulo t ON ti.id\_titulo=t.id\_titulo

INNER JOIN

Contrato co ON e.id\_empleado=co.id\_empleado

ORDER BY e.id\_empleado;

1. Recientemente se ha extraviado un artículo importante del laboratorio con id 2. Se desea ver la información de los investigadores que trabajan en dicho laboratorio y sus ingresos y egresos al laboratorio para encontrar posibles sospechosos, ya que una de las políticas de la empresa consiste en que los investigadores no pueden estar en el laboratorio fuera del horario de trabajo asignado. Gracias a un estudio confiable se sabe que la desaparición de dicho artículo se dio entre el 17 y 20 de enero del presente año. Solicita que este ordenado por fecha.

[fecha\_hora, id empleado, DNI, estado (puede ser ingresa si vale 1 o egresa si vale 0)]

Select

ie.fecha\_hora,

e.id\_empleado as ID,

e.DNI, ie.ingreso

FROM

Ingresa\_Egresa ie

INNER JOIN

Empleado e ON ie.id\_empleado=e.id\_empleado

WHERE

(ie.id\_laboratorio=2) AND (DAY(ie.fecha\_hora) BETWEEN 17 AND 20) AND (MONTH(ie.fecha\_hora)=1)

ORDER BY ie.fecha\_hora;

1. La empresa desea analizar la información de aquellas facturas de venta que fueron anuladas, para de esta forma evaluar el motivo y encontrar alguna forma de que se disminuya esta posibilidad. Se desea conocer la información de aquellas facturas de venta anuladas, la fecha, el id de factura y el cliente. Se lo solicita ordenado por fecha.

[fecha, id factura, cliente]

SELECT

fcv.fecha,

fcv.id\_factura\_venta,

cliente,

SUM(cantidad\*precio\_unitario) as Monto

FROM

Factura\_Cabecera\_Venta fcv

INNER JOIN

Cliente cl ON cl.id\_cliente = fcv.id\_cliente

INNER JOIN

Factura\_Detalle\_Venta fdv ON fcv.id\_factura\_venta=fdv.id\_factura\_venta

WHERE

anulada = 1

GROUP BY fcv.id\_factura\_venta

ORDER BY fcv.fecha;

1. La empresa quiere ver su stock de productos y materiales en su sucursal. Desea conocer la cantidad de materiales disponibles en el almacén por tipo de material, al igual que los productos por tipo de producto.

[tipo, cantidad, esMaterial(el cual tendrá valor 0 si se trata de un producto y 1 si es material)]

(SELECT

tp.tipo,

SUM(prod.cantidad) cantidad,

0 esMaterial

FROM

Tipo\_Producto tp

INNER JOIN

Producto prod ON prod.id\_tipo\_producto = tp.id\_tipo\_producto

GROUP BY tp.id\_tipo\_producto)

UNION

(SELECT

tm.tipo,

SUM(mat.cantidad) cantidad,

1 esMaterial

FROM

Tipo\_Material tm

INNER JOIN

Material mat ON mat.id\_tipo\_material = tm.id\_tipo\_material

GROUP BY tm.id\_tipo\_material);

1. La empresa está buscando renovar su stock, para esto, desea conocer información de los tres productos y materiales que menos cantidad se disponga. De esta manera fabricar urgentemente aquellos productos y comprar dichos materiales.

[tipo, objeto, cantidad, esProducto(el cual tendrá valor 1 si es un producto y 0 si es un material)]

(SELECT

tipo,

producto objeto,

cantidad,

1 esProducto

FROM

Producto pr

INNER JOIN

Tipo\_Producto tp ON pr.id\_tipo\_producto = tp.id\_tipo\_producto

ORDER BY cantidad

LIMIT 3)

UNION

(SELECT

tipo,

material objeto,

cantidad,

0 esProducto

FROM

Material mat

INNER JOIN

Tipo\_Material tm ON mat.id\_tipo\_material = tm.id\_tipo\_material

ORDER BY cantidad

LIMIT 3);

1. La empresa necesita conocer el rendimiento de las ventas, para esto quiere comparar las ventas obtenidas con el promedio de todas sus ventas y ver si vendió por debajo o por encima de este. Para esto desea conocer, ordenados por fechas, los importes obtenidos esa fecha, el importe promedio y la diferencia entre el importe de la fecha correspondiente y el importe promedio.

[fecha, importe, promedio, diferencia]

SELECT

fcv.fecha AS Fecha,

SUM(fdv.cantidad \* fdv.precio\_unitario) AS Importe,

@prom:=(SELECT

TRUNCATE(AVG(fdv.cantidad \* fdv.precio\_unitario),3) AS PromedioVentas

FROM

Factura\_Detalle\_Venta fdv

INNER JOIN

Factura\_Cabecera\_Venta fcv ON fdv.id\_factura\_venta = fcv.id\_factura\_venta

WHERE

fcv.anulada = 0) AS Promedio,

TRUNCATE((SUM(fdv.cantidad \* fdv.precio\_unitario) - @prom),3) Diferencia

FROM

Factura\_Detalle\_Venta fdv

INNER JOIN

Factura\_Cabecera\_Venta fcv ON fdv.id\_factura\_venta = fcv.id\_factura\_venta

WHERE

fcv.anulada = 0

GROUP BY Fecha

ORDER BY fcv.fecha;