**Assignment 3b - Schema**

Query 1: SQLXML

/\* SQLXML\_query1 with xml datatype (address)

used tables: Customer, Order, Robot

\*/

SELECT xmlroot(

xmlelement(name customer\_data,

xmlagg(

xmlelement(name customer\_details, xmlattributes(cu.customer\_id as customer\_id),

xmlforest(cu.lname, cu.fname, cu.bday, cu.address),

xmlelement(name order\_data, xmlelement(name orders, xmlattributes(o.order\_id as order\_id),

xmlforest(o.item\_id, o.robot\_id))),

xmlelement(name robots, xmlattributes(r.robot\_id as robot\_id),

xmlforest(r.rname, r.rtype, r.rfunction)))))

,version 1.0, standalone yes)

FROM public."Customer" cu

FULL JOIN public."Order" o

ON cu.customer\_id=o.customer\_id

FULL JOIN public."Robot" r

ON o.robot\_id=r.robot\_id

WHERE cu.customer\_id < 31

Result 1: 19 first lines of the result as a table

Application, table, Excel

Description automatically generated

Query 2: SQLXML

/\* SQLXML\_query2 with xml datatype (orderdetail)

used tables: Order, OrderItem, Ingredient

\*/

SELECT xmlroot(

xmlelement(name order\_data\_all,

xmlagg(

xmlelement(name order\_meta, xmlattributes(o.order\_id as order\_id), xmlforest(o.order\_date, o.orderdetail),

xmlelement(name orderitems, xmlattributes(oi.item\_id as item\_id), xmlforest(oi.itname, oi.itsize, oi.itcost)),

xmlelement(name ingredients, xmlattributes(i.ingredient\_id as ingredient\_id), xmlforest(i.inname, i.inalergic, i.intype)))))

,version 1.0, standalone yes)

FROM public."Order" o

FULL JOIN public."OrderItem" oi

ON o.item\_id=oi.item\_id

FULL JOIN public."Ingredient" i

ON oi.ingredient\_id=i.ingredient\_id

WHERE o.order\_id > 100 AND o.order\_id < 131

Result 2: 19 first lines of the result as a table

Table

Description automatically generated

Query 3: SQL

/\* SQLXML\_query3 without xml datatype

used tables: OrderItem, Ingredient, Stock

\*/

SELECT xmlroot(

xmlelement(name orderitem\_burgerbun,

xmlagg(

xmlelement(name orderitem\_deepdive,

xmlelement(name orderitem\_meta, xmlattributes(oi.item\_id as item\_id), xmlforest(oi.itname, oi.itsize, oi.itcost)),

xmlelement(name ingredient\_info,

xmlelement(name ingredient, xmlattributes(i.ingredient\_id as ingredient\_id),

xmlforest(i.inname, i.inalergic, i.intype)),

xmlelement(name stock\_status, xmlelement(name stock\_info, xmlattributes(s.stock\_id as stock\_id),

xmlforest(s.scurrent, s.snew, s.smaxcapa))))))

), version 1.0, standalone yes)

FROM public."OrderItem" oi

FULL JOIN public."Ingredient" i

ON oi.ingredient\_id=i.ingredient\_id

FULL JOIN public."Stock" s

ON s.stock\_id=i.stock\_id

WHERE oi.itname ='Burgerbun'

Result 3: 19 first lines of the result as a table

Table

Description automatically generated

Query 4: SQL

/\* SQLXML\_query4 without xml datatype

used tables: Robot, Delivery, Stock

\*/

SELECT xmlroot(

xmlelement(name robot\_sandra,

xmlagg( xmlelement(name robot\_taskboard,

xmlelement(name robot\_info, xmlattributes(r.robot\_id as robot\_id),

xmlforest(r.rname, r.rtype, r.rfunction)),

xmlelement(name delivery\_info,

xmlelement(name delivery, xmlattributes(d.delivery\_id as delivery\_id),

xmlforest(d.ddate, d.vendor, d.damount)),

xmlelement(name stock\_replenishment,

xmlelement(name stock\_info, xmlattributes(s.stock\_id as stock\_id),

xmlforest(s.ingredient\_id, s.scurrent, s.snew, s.smaxcapa))))))

), version 1.0, standalone yes)

FROM public."Robot" r

FULL JOIN public."Delivery" d

ON r.delivery\_id=d.delivery\_id

FULL JOIN public."Stock" s

ON s.stock\_id=d.stock\_id

WHERE r.rname='Sandra' AND (d.vendor='Cocacola Comp.' OR d.vendor='FreshMarket')

Result 4: 19 first lines of the result as a table

Graphical user interface, table

Description automatically generated