1

Q1. b) RA

i)
$$ppe\_or\_testing := \sigma_{tagname='PPE' \vee tagname='Testing'}(ProductTag)$$
$$names := \pi_{sname}(ppe\_or\_testing \bowtie Catalog \bowtie Suppliers)$$

ii)
$$ppe\_filter := \sigma_{tagname='PPE'}(ProductTag) \bowtie \sigma_{cost<10 \wedge cost>420}(Catalog)$$
$$sids := \pi_{sid}(ppe\_filter)$$

iii)
$$to\_remove := \sigma_{tagname='PPE'}(ProductTag) \bowtie \sigma_{10<=cost<=1337}(Catalog)$$
$$sids := \pi_{sid}(Catalog) - \pi_{sid}(to\_remove)$$

iv)
$$cleaning := \rho_{cleaning}(\pi_{pid}(\sigma_{tagname='Cleaning'}(ProductTag)))$$
$$suppliers := \sigma_{sid}(Catalog \div cleaning)$$

v)
$$cat1 := \rho_{cat1}(Catalog)$$
$$cat2 := \rho_{cat2}(Catalog)$$
$$pairs := \sigma_{cat1.sid<>cat2.sid}(cat1 \times cat2)$$
$$filter := \sigma_{cat1.cost>=cat2.cost \times 1.2}(pairs)$$
$$format := \rho_{format(sid1,sid2)}(\pi_{cat1.sid,cat2.sid}(filter))$$

vi)
$$cat1 := \rho_{cat1}(Catalog)$$
$$cat2 := \rho_{cat2}(Catalog)$$
$$unique := \sigma_{cat1.sid<>cat2.sid \wedge cat1.pid=cat2.pid}(cat1 \times cat2)$$
$$format := \rho_{format(pid)}(\sigma_{cat1.pid}(unique))$$

vii)
$$suptech1 := \rho_{suptech1(sid,pid,cost)}(\pi_{Catalog.sid,Catalog.pid,Catalog.cost}((\sigma_{tagname='SuperTech'}(ProductTag) \bowtie Catalog \bowtie \sigma_{scountry='USA'}(Suppliers)))$$
$$suptech2 := \rho_{suptech2}(suptech2)$$
$$pairs := suptech1 \bowtie_{subtech1.cost<subtech2.cost} suptech2$$
$$less\_than\_something := \rho_{less\_than\_something(pid)}(\pi_{subtech1.pid}(pairs))$$
$$most\_expensive := \pi_{pid}(Products) - less\_than\_something$$
$$answer := \pi_{sid}(most\_expensive \bowtie Catalog)$$

viii)

$suptech1 := \rho_{suptech1(sid,pid,cost)}(\pi_{Catalog.sid,Catalog.pid,Catalog.cost}((\sigma_{tagname='SuperTech'}(ProductTag) \bowtie Catalog \bowtie \sigma_{scountry='USA'}(Suppliers)))$

$suptech2 := \rho_{suptech2}(suptech2)$

$pairs := suptech1 \bowtie_{subtech1.cost<subtech2.cost} suptech2$

$less\_than\_something := \rho_{less\_than\_something(pid)}(\pi_{subtech1.pid}(pairs))$

$most\_expensive := \pi_{pid}(Products) - less\_than\_something$

$most\_expensive\_items := \rho_{most\_expensive\_items(sid,pid,cost)}(\pi_{subtech1.sid,subtech1.pid,subtech1.cost}(subtech1$
$\bowtie_{suptech1.pid=most\_expensive.pid} (most\_expensive)))$

$rem\_most\_exp1 := subtech1 - most\_expensive\_items$

- Most expensive item removed, find the most expensive item in tables without the previous most expensive item, so find second most expensive item

$rem\_most\_exp2 := \rho_{rem\_most\_exp2}(rem\_most\_exp1)$

$rem\_pairs := rem\_most\_exp1 \bowtie_{rem\_most\_exp1.cost<rem\_most\_exp2.cost} rem\_most\_exp2$

$rem\_less\_than := \rho_{rem\_less\_than(pid}(\pi_{rem\_most\_exp1.pid}(rem\_pairs))$

$second := \pi_{sid}(Products) - most\_expensive - rem\_less\_than$

$answer := \pi_{sid}(second \bowtie Catalog)$

ix)

$products := \rho_{products(pid,sid,cost)}(\pi_{Product.pid,Suppliers.sid}(Product \bowtie Catalog))$

$unique := \pi_{pid}(R1)$

$all := products \div unique$

$filter := \sigma_{cost<69}(all)$

$answer := \pi_{sid}(filter)$

x)

$\pi_{pid}(\sigma_{quantity=0}(Inventory))$

Q1. c) RA

i)
$Cat1 := \rho_{Cat1}(Catalog)$
$Cat2 := \rho_{Cat2}(Catalog)$

Find business relationships

$JoinedSuppliers := Subsuppliers \bowtie_{Subsuppliers.sid=Cat1.sid} (Cat1) \bowtie_{Subsuppliers.subid=Cat2.sid} (Cat2)$

Find products in both catalogs with quantity of 0

$Both := \sigma_{Cat1.pid=Cat2.pid}(JoinedSuppliers)$
$Stock := Both \bowtie_{Inventory.pid=Cat1.pid} (Inventory)$
$None := \sigma_{quantity=0}(Stock)$

Format to desired format

$Format := \rho_{Format(pid,sid1,sid2,cost1,cost2)}(\pi_{Cat1.pid,Subsuppliers.sid,Subsuppliers.subid,Cat1.cost,Cat2.cost}(None))$

ii)
$Cat1 := \rho_{Cat1}(Catalog)$
$Cat2 := \rho_{Cat2}(Catalog)$

Get products supplied by different suppliers with the same price

$Filtered := \sigma_{Cat1.sid<>Cat2.sid \wedge Cat1.pid=Cat2.pid \wedge Cat1.cost=Cat2.cost}(Cat1 \times Cat2)$

Format to desired form

$Format := \rho_{Format(pid,sid,cost)}(\pi_{Cat1.pid,Cat1.sid,Cat1.cost}(Filtered))$

iii) Create duplicates of the tags and get cross product

$Tag1 := \rho_{Tag1}(ProductTag)$
$Tag2 := \rho_{Tag2}(ProductTag)$
$Tag3 := \rho_{Tag3}(ProductTag)$
$Cross := Tag1 \times Tag2 \times Tag3$

Find products with three different tags

$SameProduct := \sigma_{Tag1.pid=Tag2.pid=Tag3.pid \wedge Tag1.tagname<>Tag2.tagname<>Tag3.tagname}(Cross)$

Find products where one tag is PPE and second tag is Super Tech

$Filter := \sigma_{Tag1.tagname='PPE' \wedge Tag2.tagname='SuperTech'}(SameProduct)$

Get the required information from the other tables and format for answer

$Products := \rho_{pid}(\pi_{Tag1.pid}(Filter))$
$All := (Products \bowtie_{Products.pid=Catalog.pid} Catalog) \bowtie_{Products.pid=Product.pid} Product$
$Format := \rho_{Format(pid,pname,cost)}(\pi_{Products.pid,pname,cost}(All))$

iv)
Create copies of subsuppliers

$Sub1 := \rho_{Sub1}(Subsuppliers)$
$Sub2 := \rho_{Sub2}(Subsuppliers)$

Find reciprocal suppliers

$Reciprocal := Sub1 \bowtie_{Sub1.sid=Sub2.subid \land Sub1.subid=Sub2.sid} Sub2$
$Needed1 := \rho_{sid,subid}(\pi_{Sub1.sid,Sub1.subid}(Reciprocal))$
$Needed2 := \rho_{Needed2}(Needed1)$

Find pairs that have at least two unique combinations, e.g. (sid: Super, subid: Tech) won't be returned by (sid: Super, subid: Tech), (sid: Super, subid: OwO) will both be returned (even if we also have (sid: OwO, subid: Super))

$Cross := \pi_{Needed1.sid,Needed1.subid}(\sigma_{Needed1.sid=Needed2.subid \land Needed1.subid=Needed2.sid}(Needed1 \times Needed2))$
$Diff := \rho_{Diff(sid,subid)}(\pi_{Needed1.sid,Needed1.subid}(\sigma_{Needed1.sid=Needed2.sid \land Needed1.subid<>Needed2.subid}(Cross)))$

To remove these duplicate tuples, we find the reverse of all the tuples and subtract them from the ones with uncommon pairs. e.g. (sid: OwO, subid: Super) -¿ (sid: Super, subid: OwO) so when we subtract it, this tuple gets removed.

$Reverse := \rho_{Reverse(sid,subid)}(\pi_{Needed1.subid,Needed1.sid}(Cross))$
$RemoveDuplicates := Diff - Reverse$

Format for answer

$Format := \rho_{Format(recipocalSid,sid,name,address)}(\pi_{Needed1.subid,Needed1.sid,sname,saddress}(RemoveDuplicates$
$\bowtie_{Needed1.sid=Suppliers.sid} Suppliers))$