CPSC 304 Project Cover Page

Milestone #: 4

Date: August 4, 2024

Group Number: 32

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Aditya Goel	84873279	c3a6w	agoel25@student.ubc.ca
Ashmit Gupta	38002341	h6e5w	ashmitg229@gmail.com
Kaz Tahara-Edmonds	50257658	b1v9c	kazkte@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Project Description

Our project models a cloud service provider, offering customers the ability to create, update and delete projects to use the offered services. Users can utilize various services, create instances and view detailed metrics and information about the services and instances. An admin console is also available to view all the information throughout the system in one place. Additionally, users can also select filtering conditions to sort through services and find exactly what they need. Error handling and SQL injection safety has been added throughout the application.

Schema Differences

- 1. Changed CardNumber from INTEGER data type to CHAR(16) in PaymentInfo, BillingInfo and Bills because 'Integer' type in Java cannot handle such a long range of values and card numbers have to be 16 digits long.
- 2. Removed all ON UPDATE SET DEFAULT clauses because oracleDB doesn't support this operation.
- 3. Add PartnerEmail to Project Table: Fulfilled a rubric item for INSERT query by adding a foreign key (PartnerEmail) that can be set to NULL, allowing backend error handling and user notification.
- 4. Project Table: the name has been set to unique as that fit our requirements better

SQL Initialization

1. Customer(<u>Email</u>, Name, PhoneNumber, Password, Address)

EMAIL	NAME	PHONENUMBER	PASSWORD	ADDRESS
john.doe@dummy.com	John Doe	1234567890	password	9090 main st, Vancouver
james.w@dummy.com	James Welch	2345678901	security1	212 oak st, Tokyo
arjund122@dummy.com	Arjun Div	3456789012	12345pass	1009 nh 29, New Delhi
jin.park@dummy.com	Jin Park	4567890123	password	Blue springs, Seoul
jane.doe@dummy.com	Jane Doe	5678901234	num123	2100 Student Union, Vancouver

2. PartOf(Email, ProjectID)



3. Project(ProjectID, Name, Description, CreationDate, Status, PartnerEmail)

PROJECTID NAME	DESCRIPTION	CREATIOND STATUS	PARTNEREMAIL
1 Migrate 2 Analytics 3 OpsPipeLine 4 ModelTraining 5 Store 6 Optimization 7 DevOps 8 Networking	Migrating app to cloud Cloud analytics Operations Pipeline AI models on cloud GPU Cloud storage Optimizing all workflows Creating DevOps workflows Creating a cloud network	20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active	james.w@dummy.com john.doe@dummy.com john.doe@dummy.com john.doe@dummy.com john.doe@dummy.com james.w@dummy.com james.w@dummy.com

4. PaymentInfo(<u>CardNumber</u>, <u>Email</u>, CVV, <u>PostalCode</u>, ExpiryDate, IsDefault)

CARDNUMBER	EMAIL	CVV	POSTALCODE	EXPIRYDAT	ISDEF
7878932100108375 2233600070708181 2378001083456789 0101999963531010 1763867695430090	john.doe@dummy.com james.w@dummy.com arjund122@dummy.com jin.park@dummy.com john.doe@dummy.com john.doe@dummy.com	900 500 400 249 234	V6T1Z2 135-0016 110001 01014 89116 145-0654	01-DEC-24 01-DEC-25 31-DEC-24 30-JUN-26 25-MAY-25 01-JUN-29 03-MAR-27	TRUE TRUE TRUE TRUE FALSE

5. AddressInfo(<u>PostalCode</u>, City, Country)

POSTALCODE	CITY	COUNTRY
V6T1Z2 135-0016 110001 01014 89116 145-0654 1205	Vancouver Tokyo New Delhi Seoul Las Vegas Kyoto Geneva	Canada Japan India South Korea United States Japan Switzerland

6. BillingInfo(<u>CardNumber</u>, PaymentType)

CARDNUMBER	PAYMEN
4536123465366009 7878932100108375 2233600070708181 2378001083456789 0101999963531010	Credit Credit Credit Credit
1763867695430090 0065142595430090	Debit

7. CustomerUsesService(Email, Name)

EMAIL	NAME
arjund122@dummy.com james.w@dummy.com jane.doe@dummy.com jin.park@dummy.com john.doe@dummy.com john.doe@dummy.com	GPUb authIt cSQL RapidX authIt gamma

 $8. \ \ \, \mathsf{ProjectUsesService}(\underline{\textbf{ProjectID}},\,\underline{\textbf{Name}})$

PROJECTID	NAME
1	GPUb
1	RapidX
	RapidZ
1	aSQL
	authIt
	bSQL
	beta
	cSQL
	dSQL
	eSQL
1	gamma
PROJECTID	NAME
1	gigaGPU
	mlGPU
	veryFy
1	veryFyIt
2	RapidX
	RapidX
	RapidX
	RapidX
	GPUb RapidX
	RapidZ
· ·	Napiaz
PROJECTID	NAME
6	aSOL
	authIt
6	bSQL
6	beta
	cSQL
	dSQL
	eSQL
	gamma
	gigaGPU
	mlGPU veryFy
- 0	veryry
PROJECTID	NAME
6	veryFyIt
	GPUb
	RapidX
	authIt
	dSQL
	gamma
7	
8	RapidX

9. ServiceType(<u>Type</u>, CostType)

ТҮРЕ	COSTTYPE
Serverless Authentication GPUCompute HPC Database	PerJob PerUser PerDay PerDay PerQuery

 $10. Service Details (\underline{Name}, \textbf{Type}, Description, Status, CostPerUnit)$

NAME	TYPE	DESCRIPTION	STATUS	COSTPERUNIT
gamma authīt GPUb RapidX cSQL dSQL veryFy gigaGPU mIGPU aSQL bSQL	Serverless Authentication GPUCompute HPC Database Database Authentication GPUCompute GPUCompute Database Database	Handling serverless functions Authentication with large capacity GPU-based rendering service High performance computing service Cloud SQL database service Dynamic SQL Faster authentication GPU-based gaming GPU-based ML training Agile SQL Broadcom SQL	Active	2 2 10 3 3 1 10 9 11
NAME	TYPE	DESCRIPTION	STATUS	COSTPERUNIT
eSQL RapidZ beta veryFyIt	Database HPC Serverless Authentication	Eth SQL Better High performance computing service Handling serverless functions slower Even faster authentication	Active Active Active Active	1 3 2 3

11. Storage(Name, Capacity, Latency)

NAME	CAPACITY	LATENCY
cSQL	300	7
dSQL	200	9
aSQL	100	7
bSQL	200	7
eSQL	500	7

12. Compute(Name, CPUCores, MaxMemory)

NAME	CPUCORES	MAXMEMORY
GPUb	32	128
gigaGPU	8	128
mlGPU	32	1024
RapidX	16	64
RapidZ	16	64

13. Functional (Name, Timeout, Concurrency Limit)

NAME	TIMEOUT	CONCURRENCYLIMIT
gamma	300	10
authIt	200	20
veryFy	200	40
veryFyIt	200	40
beta	200	70

14. Region(Name, Location, Capacity, Status)

NAME	LOCATION	CAPACITY	STATUS
us-east-1 us-east-2 eu-central-1 eu-west-1 sa-east-1	Virginia Oregon Frankfurt Ireland Caracas	500 300 300	Active Active Active Active Active

$15.\,Available In (\underline{\textbf{ServiceName}},\, \underline{\textbf{RegionName}})$

SERVICENAME	REGIONNAME
GPUb GPUb RapidX RapidX RapidZ aSQL authIt authIt bSQL beta cSQL	eu-central-1 us-east-1
SERVICENAME	REGIONNAME
cSQL dSQL eSQL gamma gigaGPU mlGPU veryFy veryFyIt	us-east-1 us-east-1 us-east-1 us-east-1 us-east-1 us-east-1 us-east-1 us-east-1

16. ProjectSecurity(ProjectID, SecurityGroupID)

PROJECTID	SECURITYGROUPID
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8

17. SecurityProtocol(InboundProtocol, OutboundProtocol)

INBOUNDPROTOCOL	OUTBOUNDPROTOCOL
HTTP	TCP
HTTPS	UDP
FTP	SNMP
SMTP	DNS
IMAP	TLS
SSH	SSL
SFTP	FTP
ICMP	TCP

18. SecurityInfo(ProjectID, Name, InboundProtocol, CreationDate)

PROJECTID	NAME	INBOUNDPROTOCOL	CREATIOND
2 3 4 5 6 7	HyperText Transfer Protocol HyperText Transfer Protocol Secure File Transfer Protocol Simple Mail Transfer Protocol Internet Message Access Protocol Secure Shell Secure File Transfer Protocol	HTTP HTTPS FTP SMTP IMAP SSH SFTP	21–JUL–24 21–JUL–24 21–JUL–24 21–JUL–24 21–JUL–24 21–JUL–24 21–JUL–24
	Secure File Transfer Protocol Internet Control Message Protocol	SFTP ICMP	21-JUL-24 21-JUL-24

19. ServerTypeInfo(<u>ServerType</u>, Memory, Storage, OS, CPUCores)

SERVERTYPE	MEMORY	ST0RAGE	0S	CPUCORES
functional.micro functional.large compute.large compute.xlarge storage.large	1 8 8 16 32	32 64 128	Linux Linux Linux Linux Linux Windows	1 4 16 32 8

20. ServerInfo(ServerID, Name, ServerType, Status, Uptime, CreatedAt, UpdatedAt)

SERVERID N	NAME	SERVERTYPE	STATUS	UPTIME	CREATEDAT	UPDATEDAT
102 t 103 e 104 e	us-east-1 us-east-2 eu-central-1 eu-west-1 sa-east-1	functional.micro functional.large compute.large compute.xlarge storage.large	Active Active Active Active Inactive	800 600 1200	04-JAN-24 05-JAN-24 01-FEB-24 11-JUL-24 16-JUL-24	16-JUL-24 16-JUL-24 16-JUL-24

21. Instance(<u>InstanceID</u>, **Name**, **ServerID**, **ProjectID**, Type, TotalCost, Status, LaunchDate, StopDate)

INSTANCEID	NAME	SERVERID	PROJECTID	TYPE	T0TALC0ST	STATUS	LAUNCHDAT STOPDATE
2 3 4 5 15 16 6 7 8	gamma authIt GPUb RapidX cSQL cSQL cSQL cSQL gamma gamma gamma gamma	101 102 103 104 105 105 101 101	2 3 4 2 2 2 1 1 1	micro large large klarge klarge klarge klarge micro large large	140 140 200 130 170 100 90	Running	20-JUL-24 20-JUL-24 20-JUL-24 20-JUL-24 16-JUL-24 16-JUL-24 20-JUL-24 20-JUL-24 20-JUL-24 20-JUL-24 20-JUL-24
INSTANCEID 		SERVERID 101 101 101 101 101	PROJECTID 1 1 1 1		TOTALCOST 		LAUNCHDAT STOPDATE

22. Bills(InstanceID, Email, CardNumber, Cost)

INSTANCEID	EMAIL	CARDNUMBER	COST
2 3 4 5	john.doe@dummy.com james.w@dummy.com arjund122@dummy.com jin.park@dummy.com jane.doe@dummy.com john.doe@dummy.com	4536123465366009 7878932100108375 2233600070708181 2378001083456789 0101999963531010 4536123465366009	500 244 306 123 890 400

SQL Queries

Important Notes:

- 1. All gueries required for the change have been listed in the list
- 2. Question marks in the queries indicate user input
- 3. All queries are located in file DatabaseConnectionHandler.java

INSERT:

INSERT INTO project VALUES (?,?,?,?,?) [Line: 152]

• INSERT INTO PartOf VALUES (?, ?) [Line: 169]

• INSERT INTO ProjectSecurity VALUES (?, ?) [Line: 182]

• INSERT INTO PartOf VALUES (?, ?) [Line: 196]

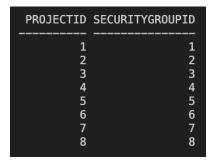
Before:

Project

PROJECTI	NAME	DESCRIPTION	CREATIOND	STATUS	PARTNEREMAIL
	? Analytics 3 OpsPipeline 4 ModelTraining 5 Store	Migrating app to cloud Cloud analytics Operations Pipeline AI models on cloud GPU Cloud storage Optimizing all workflows Creating DevOps workflows Creating a cloud network	20-JUL-24 20-JUL-24 20-JUL-24 20-JUL-24 20-JUL-24 20-JUL-24 20-JUL-24 20-JUL-24	Active Active Active Active Active Active	james.w@dummy.com john.doe@dummy.com john.doe@dummy.com john.doe@dummy.com john.doe@dummy.com james.w@dummy.com james.w@dummy.com james.w@dumy.com

PartOf

ProjectSecurity



During:



After:

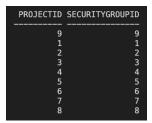
Project

PROJECTID NAME	DESCRIPTION	CREATIOND STATUS	PARTNEREMAIL
9 Cloud Operations 1 Migrate 2 Analytics 3 OpsPipetine 4 Model Training 5 Stormarianin 6 Optimization 7 DevOps 8 Networking	Creating a new Cloud Operation Migrating app to cloud Coud enalytic Coud enalytic Coud enalytic Coud straigle Ale model of the Coud of the	95-MUS-24 active 29-JUL-24 Active 29-JUL-24 Active 29-JUL-24 Active 29-JUL-24 Active 29-JUL-24 Active 29-JUL-24 Active 29-JUL-24 Active 29-JUL-24 Active	james.w@dummy.com james.w@dummy.com john.doe@dummy.com john.doe@dummy.com john.doe@dummy.com james.w@dummy.com james.w@dummy.com james.w@dummy.com james.w@dummy.com

PartOf



ProjectSecurity



DELETE [Line: 251]

• DELETE FROM project WHERE ProjectID = ?

Before:

Project

PROJECTID NAME	DESCRIPTION	CREATIOND STATUS	PARTNEREMAIL
1 Migrate 2 Analytics 3 OpsPipeLine 4 ModelTraining 5 Store 6 Optimization 7 DevOps 8 Networking	Migrating app to cloud Cloud analytics Operations Pipeline AI models on cloud GPU Cloud storage Optimizing all workflows Creating Devolps workflows Creating a cloud network	20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active	james.w@dummy.com john.doe@dummy.com john.doe@dummy.com john.doe@dummy.com john.doe@dummy.com james.w@dummy.com james.w@dummy.com james.w@dummy.com

PartOf

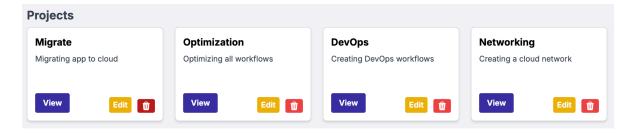
ProjectSecurity

PROJECTID	SECURITYGROUPID
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8

ProjectUsesService

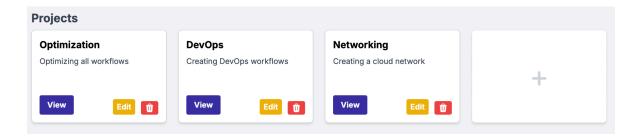


During (Deleting projectID = 1):



After:

GUI:



Project:

PROJECTID NAME	DESCRIPTION	CREATIOND STATUS	PARTNEREMAIL
2 Analytics 3 OpsPledine 4 ModelTraining 5 Store 6 Optimization 7 DevOps 8 Networking	Cloud analytics Operations Pipeline At models on cloud GPU Cloud storage Optimizing all workflows Creating DevOps workflows Creating a cloud network	20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active	john.doe@dummy.com john.doe@dummy.com john.doe@dummy.com john.doe@dummy.com james.w@dummy.com james.w@dummy.com james.w@dummy.com

PartOf

EMAIL	PROJECTID
arjund122@dummy.com	3
james.w@dummy.com	2
james.w@dummy.com	6
james.w@dummy.com	7
james.w@dummy.com	8
jane.doe@dummy.com	5
jin.park@dummy.com	4
john.doe@dummy.com	6
john.doe@dummy.com	7
john.doe@dummy.com	8

ProjectSecurity

PROJECTID	SECURITYGROUPID
2	2
3	3
4	4
5	5
6	6
7	7
8	8

ProjectUsesService

UPDATE [Line: 99]

• UPDATE project SET Name = ?, Description = ?, CreationDate = ?, Status = ? WHERE ProjectID = ?

Before:

Project

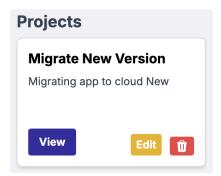
PROJECTID NAME	DESCRIPTION	CREATIOND STATUS	PARTNEREMAIL
2 Analytics 3 Opplipeline 4 ModelTraining 5 Store 6 Optimization 7 DevOps 8 Networking	Cloud analytics Operations Pipeline Ai models on cloud GPU Cloud storage Optimizing a workflows Creating DevOps workflows Creating DevOps workflows	20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active 20-JUL-24 Active	john.doe@dummy.com john.doe@dummy.com john.doe@dummy.com john.doe@dummy.com james.w@dummy.com james.w@dummy.com james.w@dummy.com

During (Editing project ID = 1):



After:

GUI:



Project

PROJECTID NAME	DESCRIPTION	CREATIOND	STATUS	PARTNEREMAIL
2 Analytics 3 OpsPipeline 4 ModelTraining 5 Store 6 Optimization 7 DevOps	Migrating app to cloud New Cloud analytics Operations Pipeline AI models on cloud GPU Cloud storage Optimizing all workflows Creating DevOps workflows Creating a cloud network	20-JUL-24 20-JUL-24 20-JUL-24 20-JUL-24 20-JUL-24 20-JUL-24 20-JUL-24 20-JUL-24	Active Active Active Active Active Active	james.w@dummy.com john.doe@dummy.com john.doe@dummy.com john.doe@dummy.com john.doe@dummy.com james.w@dummy.com james.w@dummy.com

Selection [Line: 538]

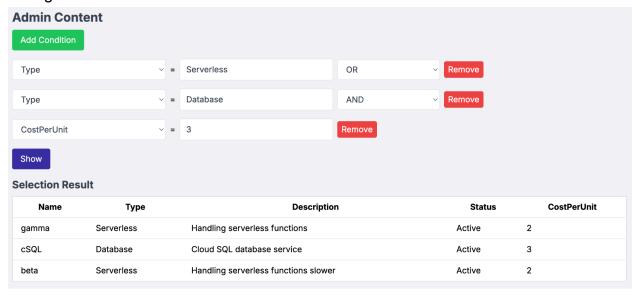
• SELECT * FROM ServiceDetails WHERE + whereQuery (whereQuery variable is passed from the front end)

Before:

Service Details

NAME	TYPE	DESCRIPTION	STATUS	COSTPERUNIT
gamma authIt GPUb RapidX cSQL dSQL veryFy gigaGPU mlGPU	Serverless Authentication GPUCompute HPC Database Database Authentication GPUCompute GPUCompute	Handling serverless functions Authentication with large capacity GPU-based rendering service High performance computing service Cloud SOL database service Dynamic SOL Faster authentication GPU-based gaming GPU-based H. training	Active	2 2 10 3 1 1 3 10
aSQL bSQL NAME	Database Database TYPE	Agile SQL Broadcom SQL DESCRIPTION	ACTIVE Active STATUS	1 1
eSQL RapidZ beta veryFyIt	Database HPC Serverless Authentication	DESCRIPTION Eth SQL Better High performance computing service Handling serverless functions slower Even faster authentication	Active Active Active Active	1 3 2 3

During:



After: No tables affected

Projection [Line: 565]

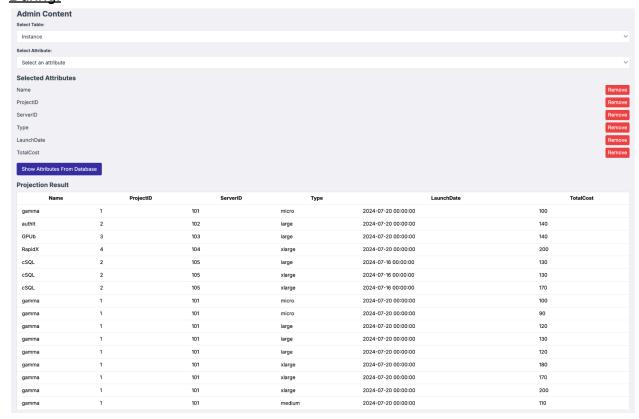
• SELECT + attributes + FROM + relation (attributes and relation variables are passed from the front end)

Before:

Instance:

INSTANCEID N	NAME	SERVERID	PROJECTID	TYPE	T0TALC0ST	STATUS	LAUNCHDAT STOPDATE
1 (101	1	micro	100	Running	20-JUL-24
2 8	authIt	102	2	large	140	Running	20-JUL-24
3 (GPUb	103	3	large	140	Running	20-JUL-24
4 F	RapidX	104	4	xlarge	200	Running	20-JUL-24
5 (cSQL	105	2	large	130	Running	16-JUL-24
15 (cSQL	105	2	xlarge	130	Running	16-JUL-24
16 (cSQL	105	2	xlarge	170	Running	16-JUL-24
6 (gamma	101	1	micro			20-JUL-24
7 (gamma	101	1	micro			20-JUL-24
	gamma	101		large			20-JUL-24
9 (gamma	101	1	large	130	Running	20-JUL-24
INSTANCEID N	NAME	SERVERID	PROJECTID	TYPE	T0TALC0ST	STATUS	LAUNCHDAT STOPDATE
10	 gamma	101	1	large	120	Running	20-JUL-24
11 (gamma	101	1	xlarge	180	Running	20-JUL-24
12 (gamma	101	1	xlarge	170		20-JUL-24
13 (gamma	101		xlarge			20-JUL-24
14 (gamma	101	1	medium	110	Running	20-JUL-24

During:



After: Does not affect any tables

Join [Line: 268 - 270]

 SELECT ServiceDetails.Name, ServiceDetails.Description, ServiceDetails.Status, ServiceDetails.CostPerUnit, ServiceType.Type, ServiceType.CostType FROM ProjectUsesService JOIN ServiceDetails ON ProjectUsesService.Name = ServiceDetails.Name JOIN ServiceType ON ServiceDetails.Type = ServiceType.Type WHERE ProjectUsesService.ProjectID = ?

Before:

ProjectUsesService:



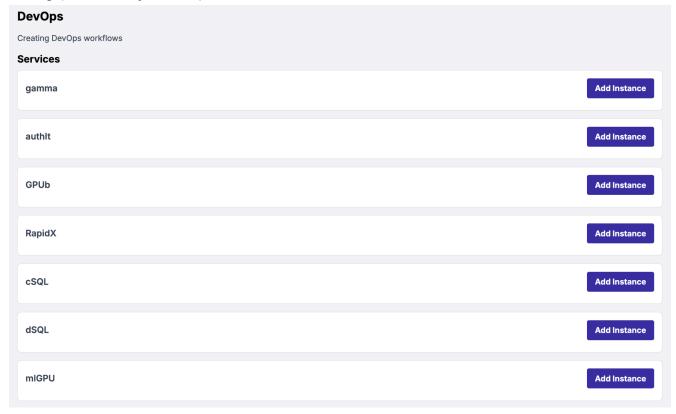
ServiceDetails

NAME	TYPE	DESCRIPTION	STATUS	COSTPERUNIT
gamma	Serverless	Handling serverless functions	Active	2
authIt	Authentication	Authentication with large capacity	Active	2
GPUb	GPUCompute	GPU-based rendering service	Active	10
RapidX	HPC	High performance computing service	Active	3
cSQL	Database	Cloud SQL database service	Active	3
dSQL	Database	Dynamic SQL	Active	1
veryFy	Authentication	Faster authentication	Active	3
gigaGPU	GPUCompute	GPU-based gaming	Active	10
mlGPU	GPUCompute	GPU-based ML training	Active	9
aSQL	Database	Agile SQL	Active	1
bSQL	Database	Broadcom SQL	Active	1
NAME	TYPE	DESCRIPTION	STATUS	COSTPERUNIT
eSQL	Database	== ===================================	Active	1
RapidZ	HPC	Better High performance computing service	Active	3
beta	Serverless	Handling serverless functions slower	Active	2
veryFyIt	Authentication	Even faster authentication	Active	3

ServiceType:

ТҮРЕ	COSTTYPE
Serverless Authentication GPUCompute HPC Database	PerJob PerUser PerDay PerDay PerQuery

During (ran for ProjectID=7):



After: Does not affect any tables.

Aggregation with GROUP BY [Line: 591]

SELECT Type, SUM(TotalCost) AS TotalCost FROM Instance WHERE ProjectID
 = ? GROUP BY Type

Before:

INSTANCEID NAME	SERVERID	PROJECTID	TYPE	T0TALC0ST	STATUS	LAUNCHDAT STOPDATE
 1 gamma	101	1	micro	 100	Running	20-JUL-24
2 authIt	102	2	large	140	Running	20-JUL-24
3 GPUb	103	3	large	140	Running	20-JUL-24
4 RapidX	104	4	xlarge	200	Running	20-JUL-24
5 cSQL	105	2	large	130	Running	16-JUL-24
15 cSQL	105	2	xlarge	130	Running	16-JUL-24
16 cSQL	105	2	xlarge	170	Running	16-JUL-24
6 gamma	101	1	micro	100	Running	20-JUL-24
7 gamma	101	1	micro	90	Running	20-JUL-24
8 gamma	101	1	large	120	Running	20-JUL-24
9 gamma	101	1	large	130	Running	20-JUL-24
INSTANCEID NAME	SERVERID	PROJECTID	TYPE	T0TALC0ST	STATUS	LAUNCHDAT STOPDATE
	101	1	large	 120	Running	20-JUL-24
11 gamma	101		xlarge		Running	20-JUL-24
12 gamma	101		xlarge		Running	20-JUL-24
13 gamma	101	1	xlarge	200	Running	20-JUL-24
14 gamma	101	1	medium	110	Running	20-JUL-24

<u>During</u> (run for ProjectID = 1):

Get Cost Per Instance Type						
Туре		Cost				
	370					
	290					
	550					
	110					
	Туре	370 290 550				

After: Does not affect any tables.

Aggregation with HAVING [Line: 609 - 611]

 SELECT ServiceDetails.Type, COUNT(*) AS Count FROM ServiceDetails, ProjectUsesService WHERE ServiceDetails.name = ProjectUsesService.name AND ProjectUsesService.ProjectID = ? GROUP BY ServiceDetails.Type HAVING COUNT(*) > 1;

Before:

ServiceDetails:

NAME	TYPE	DESCRIPTION	STATUS	COSTPERUNIT
gamma authIt	Serverless Authentication	Handling serverless functions Authentication with large capacity	Active Active	2
GPUb	GPUCompute	GPU-based rendering service	Active	10
RapidX cSOL	HPC Database	High performance computing service Cloud SOL database service	Active Active	3
dSQL veryFy	Database Authentication	Dynamic SQL Faster authentication	Active Active	1
gigaGPU	GPUCompute	GPU-based gaming	Active	10
mlGPU aSQL	GPUCompute Database	GPU-based ML training Agile SQL	Active Active	1
bSQL	Database	Broadcom SQL	Active	1
NAME	TYPE	DESCRIPTION	STATUS	COSTPERUNIT
eSQL	Database	Eth SQL	Active	1
RapidZ beta	HPC Serverless	Better High performance computing service Handling serverless functions slower	Active Active	2
veryFyIt	Authentication	Even faster authentication	Active	3

ProjectUsesService:

riojeci	Usesselvice.
PROJECTID	NAME
1	GPUb
	RapidX
1	RapidZ
	aSQL
	authIt
	bSQL
	beta
	cSQL dSQL
	eS0L
	gamma
_	5
PROJECTID	NAME
1	gigaGPU
	mlGPU
1	veryFy
	veryFyIt
	RapidX
	RapidX
	RapidX RapidX
	GPUb
	RapidX
	RapidZ
PROJECTID	NAME
	aSQL
	authIt
	bSQL
	beta
	cSQL dSQL
	eSOL
	gamma
	gigaGPU
	mlGPU
6	veryFy
PROJECTID	NAME
	veryFyIt GPUb
	RapidX
	authIt
	cSQL
	dSQL
7	gamma
7	
8	RapidX

<u>During (run for ProjectID = 1):</u>

Get Duplicate Service Counts						
Туре	Count					
Authentication	3					
GPUCompute	3					
Database	5					
Serverless	2					
HPC	2					

After: Does not affect any tables.

Nested Aggregation with GROUP BY [Line: 629 - 635]

```
    SELECT I.Type
        FROM Instance I
        WHERE I.ProjectID = ?
        GROUP BY I.Type
        HAVING AVG(I.TotalCost) >= ALL (
            SELECT AVG(I2.TotalCost)
        FROM Instance I2
        WHERE I2.ProjectID = ?
        );
```

Before:

NSTANCEID	NAME	SERVERID	PROJECTID	TYPE	TOTALCOST	STATUS	LAUNCHDAT STOPDATE
1	gamma	101	1	micro	100	Running	20-JUL-24
2	authIt	102	2	large	140	Running	20-JUL-24
3	GPUb	103	3	large	140	Running	20-JUL-24
4	RapidX	104	4	xlarge		Running	20-JUL-24
5	cS0L	105		large		Running	16-JUL-24
15	cSQL	105	2	xlarge	130	Running	16-JUL-24
16	cSQL	105	2	xlarge		Running	16-JUL-24
6	gamma	101	1	micro	100	Running	20-JUL-24
7	gamma	101	1	micro	90	Running	20-JUL-24
8	gamma	101	1	large	120	Running	20-JUL-24
9	gamma	101	1	large	130	Running	20-JUL-24
ISTANCEID	NAME	SERVERID	PROJECTID	TYPE	TOTALCOST	STATUS	LAUNCHDAT STOPDATE
10	gamma	101	1	large	120	Running	
11	gamma	101	1	xlarge	180	Running	20-JUL-24
12	gamma	101	1	xlarge	170	Running	20-JUL-24
13	gamma	101	1	xlarge	200	Running	20-JUL-24
	gamma	101		medium		Running	20-JUL-24

<u>During (run for ProjectID = 1):</u>

```
Get Instance Type With Maximum Average Cost
xlarge
```

After: Does not affect any tables.

```
Division [Line: 653 - 658]
```

Before:

Project:

```
PROJECTID NAME

DESCRIPTION

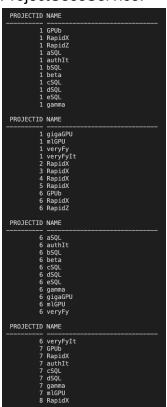
Rigrate Migrating app to cloud

Analytics Cloud analytics Cloud analytics 28–JUL-24 Active james.w@dunmy.com
3 OpsPipeline Operations Pipeline 28–JUL-24 Active john.doe@dunmy.com
4 ModelTraining AI models on cloud GPU 28–JUL-24 Active john.doe@dunmy.com
5 Store Cloud storage 28–JUL-24 Active john.doe@dunmy.com
6 Optimization Optimizing all workflows 28–JUL-24 Active john.doe@dunmy.com
7 DevOps Creating DevOps workflows 28–JUL-24 Active james.w@dunmy.com
8 Networking Creating a cloud network 28–JUL-24 Active james.w@dunmy.com
```

ServiceDetails

NAME	TYPE	DESCRIPTION	STATUS	COSTPERUNIT
gamma	Serverless	Handling serverless functions	Active	2
authIt	Authentication	Authentication with large capacity	Active	2
GPUb	GPUCompute	GPU-based rendering service	Active	10
RapidX	HPC	High performance computing service	Active	3
cSQL	Database	Cloud SQL database service	Active	3
dSQL	Database	Dynamic SQL	Active	1
veryFy	Authentication	Faster authentication	Active	3
gigaGPU	GPUCompute	GPU-based gaming	Active	10
mlGPU	GPUCompute	GPU-based ML training	Active	9
aSQL	Database	Agile SQL	Active	1
bSQL	Database	Broadcom SQL	Active	1
NAME	TYPE	DESCRIPTION	STATUS	COSTPERUNIT
eS0L	Database			1
RapidZ	HPC	Better High performance computing service	Active	3
beta	Serverless	Handling serverless functions slower	Active	2
veryFyIt	Authentication	Even faster authentication	Active	3

ProjectUsesService:



<u>During (ran for Email=john.doe@dummy.com):</u>



After: Does not affect any table.