

CPSC 304 Project Cover Page

Milestone #: 4

Date: November 30

Group Number: 46

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Vito Lin	76174770	x0n3w	vitolin2003@gmail.com
Yuang Chen	84122456	m9s0n	bryan1mageacal@gmail.com
Corina Dong	18129544	j5p6x	corina.dong@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

SQL Script

```
SET FOREIGN_KEY_CHECKS = 0;
DROP TABLE IF EXISTS Guild;
DROP TABLE IF EXISTS Status;
DROP TABLE IF EXISTS Player;
DROP TABLE IF EXISTS InventoryObject;
DROP TABLE IF EXISTS InventoryRecord;
DROP TABLE IF EXISTS Inventory;
DROP TABLE IF EXISTS Equipment;
DROP TABLE IF EXISTS Item;
DROP TABLE IF EXISTS Quest;
DROP TABLE IF EXISTS QuestRecord;
DROP TABLE IF EXISTS Passive;
SET FOREIGN_KEY_CHECKS = 1;
-- GRANT ALL ON db.* TO 'username'@'%';

CREATE TABLE Status (
    Lv INT PRIMARY KEY,
    HP INT,
    MY INT,
    ATK INT
);

-- add constraint that guilds must have at least one player
CREATE TABLE Guild (
    ID INT PRIMARY KEY,
    LV INT,
    Name VARCHAR(50)
);

CREATE TABLE Player (
    ID INT PRIMARY KEY,
    Name VARCHAR(20),
    Lv INT,
    GuildID INT,
    FOREIGN KEY (GuildID) REFERENCES Guild(ID) ON DELETE CASCADE,
    FOREIGN KEY (Lv) REFERENCES Status(Lv) ON DELETE CASCADE
);
```

```

CREATE TABLE InventoryObject (
    InventoryID INT PRIMARY KEY,
    Name VARCHAR(50)
);

CREATE TABLE InventoryRecord (
    InventoryID INT,
    PlayerID INT,
    Quantity INT,
    PRIMARY KEY (InventoryID, PlayerID),
    FOREIGN KEY (InventoryID) REFERENCES InventoryObject(InventoryID) ON
DELETE CASCADE,
    FOREIGN KEY (PlayerID) REFERENCES Player(ID) ON DELETE CASCADE
);

CREATE TABLE Equipment (
    EquipmentID INT PRIMARY KEY,
    FOREIGN KEY (EquipmentID) REFERENCES InventoryObject(InventoryID),
    HP_plus INT,
    MP_plus INT,
    ATK_plus INT
);

CREATE TABLE Item (
    ItemID INT PRIMARY KEY,
    FOREIGN KEY (ItemID) REFERENCES InventoryObject(InventoryID),
    Quality INT
);

CREATE TABLE Quest (
    ID INT PRIMARY KEY,
    Name VARCHAR(50)
);

-- this should have a number of times cleared column
CREATE TABLE QuestRecord (
    QuestID INT,

```

```
PlayerID INT,  
TimesCleared INT,  
FOREIGN KEY (QuestID) REFERENCES Quest(ID) ON DELETE CASCADE,  
FOREIGN KEY (PlayerID) REFERENCES Player(ID) ON DELETE CASCADE  
);  
  
CREATE TABLE Passive (  
EquipmentID INT,  
PassiveID INT,  
PassiveName VARCHAR(50),  
PRIMARY KEY (EquipmentID, PassiveID),  
FOREIGN KEY (EquipmentID) REFERENCES Equipment(EquipmentID)  
);  
  
INSERT INTO InventoryObject VALUES (1, 'Aquila Favonia');  
INSERT INTO Equipment VALUES (1, 1000, 0, 0);  
  
INSERT INTO InventoryObject VALUES (2, 'Dull Blade');  
INSERT INTO Equipment VALUES (2, 0, 0, 0);  
  
INSERT INTO InventoryObject VALUES (3, 'Hanatsuki Paddle');  
INSERT INTO Equipment VALUES (3, 100, 0, 0);  
  
INSERT INTO InventoryObject VALUES (4, 'Rusty Sickle');  
INSERT INTO Equipment VALUES (4, 10, 0, 0);  
  
INSERT INTO InventoryObject VALUES (5, 'Jotunheim');  
INSERT INTO Equipment VALUES (5, 5000, 0, 0);  
  
INSERT INTO InventoryObject VALUES (13, 'Diamond Shield');  
INSERT INTO Equipment VALUES (13, 213, 12, 1);  
  
INSERT INTO InventoryObject VALUES (7, 'Stick');  
INSERT INTO Equipment VALUES (7, 500, 100, 30);  
  
INSERT INTO InventoryObject VALUES (8, 'Excalibur');  
INSERT INTO Equipment VALUES (8, 5001, 1010, 310);
```

```
INSERT INTO InventoryObject VALUES (10, 'Thunder Staff');
INSERT INTO Equipment VALUES (10, 2000, 0, 20);

INSERT INTO InventoryObject VALUES (9, 'Healing Potion');
INSERT INTO Item VALUES (9, 12);

INSERT INTO InventoryObject VALUES (6, 'Jar');
INSERT INTO Item VALUES (6, 1500);

INSERT INTO InventoryObject VALUES (11, 'Invisibility Cloak');
INSERT INTO Item VALUES (11, 123);

INSERT INTO InventoryObject VALUES (12, 'Fireball Scroll');
INSERT INTO Item VALUES (12, 756);

INSERT INTO InventoryObject VALUES (14, 'Health Elixir');
INSERT INTO Item VALUES (14, 4);

INSERT INTO Quest VALUES (1, 'Echoes of the Forgotten Realm');
INSERT INTO Quest VALUES (2, 'Whispers in the Enchanted Woods');
INSERT INTO Quest VALUES (3, 'Serpent Lair Mystery');
INSERT INTO Quest VALUES (4, 'The Crystal Cavern Expedition');
INSERT INTO Quest VALUES (5, 'Shadows Over Sunhaven');
INSERT INTO Quest VALUES (6, 'Riddle of the Lost Crypt');
INSERT INTO Quest VALUES (7, 'Moonlit Mirage Conundrum');
INSERT INTO Quest VALUES (8, 'Secrets of the Celestial Garden');
INSERT INTO Quest VALUES (9, 'The Ebonfire Ritual');
INSERT INTO Quest VALUES (10, 'Enigma of the Astral Isles');

INSERT INTO Guild VALUES (1, 10, 'Eternal Eclipse Syndicate');
INSERT INTO Guild VALUES (2, 10, 'Whispering Shadows Alliance');
INSERT INTO Guild VALUES (3, 15, 'Sapphire Serpent Society');
INSERT INTO Guild VALUES (4, 25, 'Celestial Vanguard');
INSERT INTO Guild VALUES (5, 18, 'Golden Griffin Brotherhood');
INSERT INTO Guild VALUES (6, 30, 'Radiant Rose Coalition');
```

```
INSERT INTO Guild VALUES (7, 22, 'Silent Storm Union');
INSERT INTO Guild VALUES (8, 12, 'Lunar Phoenix Clan');
INSERT INTO Guild VALUES (9, 28, 'Iron Crown Syndicate');
INSERT INTO Guild VALUES (10, 17, 'Ember Wolf Pack');

INSERT INTO Status VALUES (1, 10, 10, 10);
INSERT INTO Status VALUES (2, 15, 15, 15);
INSERT INTO Status VALUES (3, 20, 20, 20);
INSERT INTO Status VALUES (4, 25, 25, 25);
INSERT INTO Status VALUES (5, 30, 30, 30);
INSERT INTO Status VALUES (6, 35, 35, 35);
INSERT INTO Status VALUES (7, 40, 40, 40);
INSERT INTO Status VALUES (8, 45, 45, 45);
INSERT INTO Status VALUES (9, 50, 50, 50);
INSERT INTO Status VALUES (10, 55, 55, 55);

INSERT INTO Player (ID, Name, Lv, GuildID) VALUES (1, 'DragonSlayer01', 1, 1);
INSERT INTO Player (ID, Name, Lv, GuildID) VALUES (2, 'PixelKnight', 2, 1);
INSERT INTO Player (ID, Name, Lv, GuildID) VALUES (3, 'StealthArcher', 3, 2);
INSERT INTO Player (ID, Name, Lv, GuildID) VALUES (4, 'ArcaneWizard', 4, 2);
INSERT INTO Player (ID, Name, Lv, GuildID) VALUES (5, '727', 5, 3);

INSERT INTO InventoryRecord (InventoryID, PlayerID, Quantity) VALUES (1, 1, 10);
INSERT INTO InventoryRecord (InventoryID, PlayerID, Quantity) VALUES (2, 1, 5);
INSERT INTO InventoryRecord (InventoryID, PlayerID, Quantity) VALUES (3, 1, 20);
INSERT INTO InventoryRecord (InventoryID, PlayerID, Quantity) VALUES (4, 1, 1);
INSERT INTO InventoryRecord (InventoryID, PlayerID, Quantity) VALUES (5, 1, 3);
INSERT INTO InventoryRecord (InventoryID, PlayerID, Quantity) VALUES (6, 1, 9);
```

```
INSERT INTO InventoryRecord (InventoryID, PlayerID, Quantity) VALUES (2, 2, 15);
INSERT INTO InventoryRecord (InventoryID, PlayerID, Quantity) VALUES (3, 3, 20);
INSERT INTO InventoryRecord (InventoryID, PlayerID, Quantity) VALUES (4, 4, 25);
INSERT INTO InventoryRecord (InventoryID, PlayerID, Quantity) VALUES (5, 5, 30);

INSERT INTO QuestRecord VALUES (1, 1, 1);
INSERT INTO QuestRecord VALUES (2, 1, 1);
INSERT INTO QuestRecord VALUES (3, 1, 1);
INSERT INTO QuestRecord VALUES (4, 1, 1);
INSERT INTO QuestRecord VALUES (5, 1, 1);
INSERT INTO QuestRecord VALUES (6, 1, 1);
INSERT INTO QuestRecord VALUES (7, 1, 1);
INSERT INTO QuestRecord VALUES (8, 1, 1);
INSERT INTO QuestRecord VALUES (9, 1, 1);
INSERT INTO QuestRecord VALUES (10, 1, 1);

INSERT INTO QuestRecord VALUES (2, 2, 1);
INSERT INTO QuestRecord VALUES (7, 2, 2);

INSERT INTO QuestRecord VALUES (1, 3, 7);

INSERT INTO QuestRecord VALUES (10, 4, 1);

INSERT INTO Passive VALUES (1, 1, 'Guardians Shield');
INSERT INTO Passive VALUES (1, 2, 'Aura of Fortitude');
INSERT INTO Passive VALUES (2, 2, 'Aura of Fortitude');
INSERT INTO Passive VALUES (3, 3, 'Rapid Strikes');
INSERT INTO Passive VALUES (4, 4, 'Precision Mastery');
INSERT INTO Passive VALUES (4, 5, 'Elemental Fusion');
INSERT INTO Passive VALUES (5, 6, 'Arcane Ward');
```

A. Description of Project

Our project is a game simulator where users can create characters, equip them with various items, and record their quest progress. Abstract functions enable the display of player-related information and support advanced searches, like identifying players who have completed all quests in the game. Besides, each player and equipment has their own status and players are allowed join one and only one guild.

B. Difference in Schema

Since the original ER diagram and dataset is too complicated to implement in one month. We have to simplify it a bit and only keep 8 different entities. We still keep the core entities including Player, Equipment, item, and Quest and the schema now still fulfill the requirement.

C. Schema & Screenshots of SQL Tables

Schema:

Status (Lv , HP, MP, ATK)

Guild (ID, LV, Name)

Player (ID, Name, **Lv**, **GuildID**)

InventoryObject (InventoryID, Name)

InventoryRecord (**InventoryID**, **PlayerID**, Quantity)

Equipment (**EquipmentID**, HP_plus, MP_plus, ATK_plus)

Item (**ItemID**, Quality)

Quest (ID, Name)

QuestRecord (**QuestID**, **PlayerID**, TimesCleared)

Passive (**EquipmentID**, **PassiveID**, PassiveName)

					Status					
Lv	1	2	3	4	5	6	7	8	9	10
HP	10	15	20	25	30	35	40	45	50	55
MP	10	15	20	25	30	35	40	45	50	55
ATK	10	15	20	25	30	35	40	45	50	55
Player										
ID	1	2	3	4	5					
Name	DragonSlayer01	PixelKnight	StealthArcher	ArcaneWizard	727					
Lv	1	2	3	4	5					
GuildID	1	1	2	2	3					
InventoryRecord										
InventoryID	1	2	2	3	3	4	4	5	5	6
PlayerID	1	1	2	1	3	1	4	1	5	1
Quantity	10	5	15	20	20	1	25	3	30	9
Equipment										
EquipmentID	1	2	3	4	5	7	8	10	13	
HP_plus	1000	0	100	10	5000	500	5001	2000	213	
MP_plus	0	0	0	0	0	100	1010	0	12	
ATK_plus	0	0	0	0	0	30	310	20	1	

QuestRecord										
QuestID	1	1	2	2	3	4	5	6	7	7
PlayerID	1	3	1	2	1	1	1	1	1	2
TimesCleared	1	7	1	1	1	1	1	1	1	2

Guild										
ID	1	2	3	4	5	6	7	8	9	10
LV	10	10	15	25	18	30	22	12	28	17
Name	Eternal Eclipse Syndicate	Whispering Shadows Alliance	Sapphire Serpent Society	Celestial Vanguard	Golden Griffin Brotherhood	Radiant Rose Coalition	Silent Storm Union	Lunar Phoenix Clan	Iron Crown Syndicate	Ember Wolf Pack

Quest										
ID	1	2	3	4	5	6	7	8	9	10
Name	shoes of the Forgotten Realm	Whispers in the Enchanted Woods	Serpent Lair Mystery	The Crystal Cavern Expedition	Shadows Over Sunhaven	Riddle of the Lost Crypt	Moonlit Mirage Conundrum	Secrets of the Celestial Garden	The Ebonfire Ritual	Enigma of the Astral Isles

Passive							
EquipmentID	1	1	2	3	4	4	5
PassiveID	1	2	2	3	4	5	6
PassiveName	Guardians Shield	Aura of Fortitude	Aura of Fortitude	Rapid Strikes	Precision Mastery	Elemental Fusion	Arcane Ward

Item			
ItemID	6	9	11
Quality	1500	12	123

InventoryObject													
InventoryID	1	2	3	4	5	6	7	8	9	10	11	12	13
Name	Aquila Favonia	Dull Blade	Hanatsuki Paddle	Rusty Sickle	Jotunheim	Jar	Stick	Excalibur	Healing Potion	Thunder Staff	Invisibility Cloak	Fireball Scroll	Diamond Shield

D. SQL Queries

	appService.js	appController.js	scripts.js
INSERT <pre>INSERT INTO Player (id, name) VALUES (?, ?)</pre>	Line 82	Line 59	Line 109
DELETE <pre>DELETE FROM Player WHERE ID = ?</pre>	Line 146	Line 120	Line 223
UPDATE <pre>UPDATE Player SET GuildID = ? where ID = ?</pre>	Line 92 & 102 & 125	Line 60	Line 133
Selection <pre>SELECT * FROM Player WHERE ?? = ?</pre>	Line 193	Line 132	Line 494
Projection <pre>SELECT ?? FROM ??</pre>	Line 113	Line 139	Line 391
Join <pre>SELECT Player.Name FROM Player INNER JOIN InventoryRecord ON Player.ID = InventoryRecord.PlayerID INNER JOIN InventoryObject ON InventoryRecord.InventoryID = InventoryObject.InventoryID WHERE InventoryObject.Name = ?;</pre>	Line 178	Line 189	Line 344
Aggregation with GROUP BY <pre>SELECT g.Lv as GuildLevel, sum(p.Lv) as TotalLv FROM Player p inner Join Guild g on p.GuildID = g.ID GRoup By g.Lv;</pre>	Line 169	Line 147	Line 323
Aggregation with HAVING	Line 169	Line 146 & 183	Line 335

<pre>SELECT ir.PlayerID, p.Name FROM InventoryRecord AS ir JOIN Player AS p ON ir.PlayerID = p.ID GROUP BY ir.PlayerID, p.Name HAVING COUNT(DISTINCT ir.InventoryID) > 5;</pre>			
<p>Nested Aggregation with GROUP BY</p> <pre>SELECT GuildID, Guild.Name, COUNT(*), avg(Player.Lv) FROM Player INNER JOIN Guild ON Player.GuildID = Guild.ID GROUP BY GuildID having sum(Player.Lv) > (SELECT max(Player.Lv) FROM Player);</pre>	Line 169	Line 146 & 183	Line 328
<p>Division</p> <pre>SELECT p.ID, p.Name FROM Player AS p WHERE NOT EXISTS (SELECT q.ID FROM Quest AS q WHERE NOT EXISTS (SELECT qr.QuestID FROM QuestRecord AS qr WHERE qr.PlayerID = p.ID AND qr.QuestID = q.ID))</pre>	Line 169	Line 146 & 183	Line 332

E. Screenshots to Demonstrate

1. INSERT Operation

Before:

Show PlayerTable

ID	Name	Lv	GuildID
1	DragonSlayer01	1	1
2	PixelKnight	2	1
3	StealthArcher	3	2
4	ArcaneWizard	4	2
5	727	5	3

Insert Values into PlayerTable

ID:

Enter ID

Name:

Enter Name

insert

During:

Show PlayerTable

ID	Name	Lv	GuildID
1	DragonSlayer01	1	1
2	PixelKnight	2	1
3	StealthArcher	3	2
4	ArcaneWizard	4	2
5	727	5	3

Insert Values into PlayerTable

ID:

6

Name:

aaaaaa

insert

After:

Show PlayerTable

ID	Name	Lv	GuildID
1	DragonSlayer01	1	1
2	PixelKnight	2	1
3	StealthArcher	3	2
4	ArcaneWizard	4	2
5	727	5	3
6	aaaaaa		

Insert Values into PlayerTable

ID:

6

Name:

aaaaaa

insert

Data inserted successfully!

2. DELETE Operation

Before:

Show PlayerTable

ID	Name	Lv	GuildID
1	DragonSlayer01	1	1
2	PixelKnight	2	1
3	StealthArcher	3	2
4	ArcaneWizard	4	2
5	727	5	3
6	bbbbbb		

Show PlayerTable

ID	Name	Lv	GuildID
1	DragonSlayer01	1	1
2	PixelKnight	2	1
3	StealthArcher	3	2
4	ArcaneWizard	4	2
5	727	5	3
6	bbbbbb		

Delete Name in PlayerTable

The values are case sensitive and if you enter in the wrong case, the update statement will not do anything.

ID:

Delete

Delete Name in PlayerTable

The values are case sensitive and if you enter in the wrong case, the update statement will not do anything.

ID:

Delete

After:

Show PlayerTable

ID	Name	Lv	GuildID
1	DragonSlayer01	1	1
2	PixelKnight	2	1
3	StealthArcher	3	2
4	ArcaneWizard	4	2
6	bbbbbb		

Delete Name in PlayerTable

The values are case sensitive and if you enter in the wrong case, the update statement will not do anything.

ID:

Delete

Name deleted successfully!

3. UPDATE Operation

Before:

Show PlayerTable

ID	Name	Lv	GuildID
1	DragonSlayer01	1	1
2	PixelKnight	2	1
3	StealthArcher	3	2
4	ArcaneWizard	4	2
5	727	5	3
6	aaaaaa		

Update Name in PlayerTable

The values are case sensitive and if you enter in the wrong case, the update statement will not do anything.

Player ID:

New Name:

update

During:

Show PlayerTable

ID	Name	Lv	GuildID
1	DragonSlayer01	1	1
2	PixelKnight	2	1
3	StealthArcher	3	2
4	ArcaneWizard	4	2
5	727	5	3
6	aaaaaa		

Update Name in PlayerTable

The values are case sensitive and if you enter in the wrong case, the update statement will not do anything.

Player ID:

New Name:

update

After:

Show PlayerTable

ID	Name	Lv	GuildID
1	DragonSlayer01	1	1
2	PixelKnight	2	1
3	StealthArcher	3	2
4	ArcaneWizard	4	2
5	727	5	3
6	bbbb		

Update Name in PlayerTable

The values are case sensitive and if you enter in the wrong case, the update statement will not do anything.

Player ID:

New Name:

update

Name updated successfully!

4. Selection Operation

Before:

Select Table

Find the players that have these attributes:

ID is equal to

During:

Select Table

Find the players that have these attributes:

ID is equal to

OR Name is equal to

After:

Select Table

Find the players that have these attributes:

ID is equal to

OR Name is equal to

ID	Name	Lv	GuildID
1	DragonSlayer01	1	1
3	StealthArcher	3	2

5. Projection Operation

Before:

Projection Table

Select Table: Select Attribute:

During:

Projection Table

Select Table: Select Attribute:

Nested / Join with HAVING: Finds the player level greater than the max player friendship

Projection Table

Select Table: Select Attribute:

After:

Projection Table

Select Table: Select Attribute:

EquipmentID	MP_plus	ATK_plus
1	0	0
2	0	0
3	0	0
4	0	0
5	0	0
7	100	30
8	1010	310
10	0	20
13	12	1

6. Join Operation

Before:

Find the names of all players who have a specific item in their inventory

ID:

Search

During:

Find the names of all players who have a specific item in their inventory

ID:

Search

After:

Find the names of all players who have a specific item in their inventory

ID:

Search

Name
DragonSlayer01

7. Aggregation with GROUP BY Operation

Before:

Aggregation with GROUP BY: Find the cumulative player level of all guilds that are the same level

Display!

During:

Aggregation with GROUP BY: Find the cumulative player level of all guilds that are the same level

Display!

After:

Aggregation with GROUP BY: Find the cumulative player level of all guilds that are the same level

Display!

Lv	sum(p.Lv)
10	10

8. Aggregation with HAVING Operation

Before:

Aggregation with HAVING: Find the players who have more than 5 different items in their inventory

Display!

During:

Aggregation with HAVING: Find the players who have more than 5 different items in their inventory

Display!

After:

Aggregation with HAVING: Find the players who have more than 5 different items in their inventory

Display!

PlayerID	Name
1	DragonSlayer01

9. Nested Aggregation with GROUP BY Operation

Before:

Nested Aggregation with GROUP BY: Finds the guilds whose sum of player levels are greater than the max player level (power of friendship)

Display!

During:

Nested Aggregation with GROUP BY: Finds the guilds whose sum of player levels are greater than the max player level (power of friendship)

Display!

After:

Nested Aggregation with GROUP BY: Finds the guilds whose sum of player levels are greater than the max player level (power of friendship)

Display!

GuildID	COUNT(*)	avg(Player.Lv)
2	2	3.5000

10. Division Operation

Before:

Division: find players who have completed all quests:

Display!

During:

Division: find players who have completed all quests:

Display!

After:

Division: find players who have completed all quests:

Display!

ID	Name
1	DragonSlayer01