

CPSC 304 Project Cover Page

Milestone #: 4

Date: 04/02/2024

Group Number: 90

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Desheng Lin	29605730	z3m1r	deshenglin127@gmail.com
James Wang	62824172	e0v7t	james.wang.0303@gmail.com
Xiran Wei	27327246	u5v5o	ommjfk@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

a. Description and outcome of the final project:

Our project is focused on building a game database system where users can create or log in to their own account in the system. On the main page, it displays the user's information and the language they chose to use in-game. The main page also displays buttons that are connected to other pages for users to manage their in-game data. The database, implemented using the Oracle database system, models entities such as user information, accounts, and characters, including the skills and weapons along with their relationships. The system allows users to store, modify and search in-game data, while game developers can make new updates to introduce new features.

b. How our final schema differed from the schema we turned in:

Some schema has changed due to illegal wording in Oracle Server. Like using the word 'User' as a table name, we changed it to 'UserTable' instead. The UserOwnsAccount table was separated into a UserTable and an Account table, because of the referencing conflict. The SaveCreateStoreData table is deleted because of the redundancy. We add a new attribute 'Wname' as Weapons' name in the Weapons table in order to better identify different weapons. We deleted the cname attribute in the Role table because it occurs a circular reference with the Characters_Info table. We added a new table called Map as the primary table of the LockedArea and UnlockedArea tables.

C. Schema:

- UserTable(Email: VARCHAR(50), Name: VARCHAR(50), Birthday: DATE)
Name: NOT NULL
- Account(UserID: INT, Password VANCHAR(50), Password VARCHAR(50), Language VARCHAR(50), **Email** VARCHAR(50))

Password: NOT NULL

Language: NOT NULL

Email: NOT NULL

- SavingData(DataID: CHAR(10), CreatingDate: DATE, **UserID**: INT)
CreatingDate: NOT NULL
- Characters_Stats(HP INTEGER, Playtime INT, charLevel INT)
- Characters_Info(Cname VARCHAR(50) PRIMARY KEY, charLevel INTEGER, Money INTEGER, **Rname** VARCHAR(50), **MapID** CHAR(10), currLoc CHAR(20))
- Roles(Rname: VARCHAR)
- Weapons(WeaponID: INT, wpDamage: INT, Price: INT, **Rname**: VARCHAR(50), Wname: VARCHAR(50))
- LearnSkills_Stats(SDamage: INT, Sname: VARCHAR(50), Requirement: VARCHAR(50))
- LearnSkills_Info(skillID CHAR(10), Sname VARCHAR(50), IsLearned CHAR(1), **Rname** VARCHAR(50))
- Map(MapID: CHAR(10), MapName: VARCHAR(50))
MapName: NOT NULL
- LockedArea(**MapID**: char(10), MapName: VARCHAR(50), FoggyArea: char(10))
MapName: NOT NULL
- UnlockArea(CheckPoint: char(10), **MapID**: char(10), MapName: VARCHAR(50))
MapName: NOT NULL & UNIQUE
- Coordinate(X_Coord: char(10), Y_Coord: char(10), **MapID**: char(10))
- BossInfo (Bname VARCHAR(50), Blevel INT, BossDMG INT, BossHP INT)
- DungeonInfo(dungeonID CHAR(10), dungeonName VARCHAR(50), Item VARCHAR(50), Bname VARCHAR(50), BossID CHAR(10), **MapID** CHAR(10))
- DungeonStats(dungeonName VARCHAR(50), clearStatus CHAR(1))

d. Code with corresponding queries can be found in the following lines:

1. Queries: INSERT, in DataBaseConnectionHandler, line 221
2. Queries: DELETE, in DataBaseConnectionHandler, line 310
3. Queries: UPDATE, in DataBaseConnectionHandler, line 330
4. Queries: Selection, in DataBaseConnectionHandler, line 681
5. Queries: Projection, in DataBaseConnectionHandler, line 724
6. Queries: Join, in DataBaseConnectionHandler, line 783
7. Queries: Aggregation with Group By,
in DataBaseConnectionHandler, line 611
8. Queries: Aggregation with Having, in DataBaseConnectionHandler,
line 659
9. Queries: Nested Aggregation with Group By,
in DataBaseConnectionHandler, line 824
10. Queries: Division, in DataBaseConnectionHandler, line 635

e.

Queries: INSERT:

- User can create account, which is using insert query;
- User can create new characters using insert;

Query location DataBaseConnectionHandler:290

Before:

The screenshot shows a window titled "Character Management" with a table containing headers: Character Name, Level, Money, Role, and Current Location. An "Insert New Character" dialog box is overlaid on top. The dialog box contains the following fields and controls:

- "Enter New Character Name:" followed by a text input field.
- "Select New Role:" followed by a dropdown menu showing "Warrior".
- "Select New Location:" followed by a dropdown menu showing "Town".
- An "Insert" button.

Below the dialog box, in the main window, there is an "Insert" button and a text input field labeled "Enter New Money Value Here:".

After:

The screenshot shows the same "Character Management" window after the insertion. The table now contains one row of data:

Character Name	Level	Money	Role	Current Location
Bobby	1	999	Warrior	Ocean

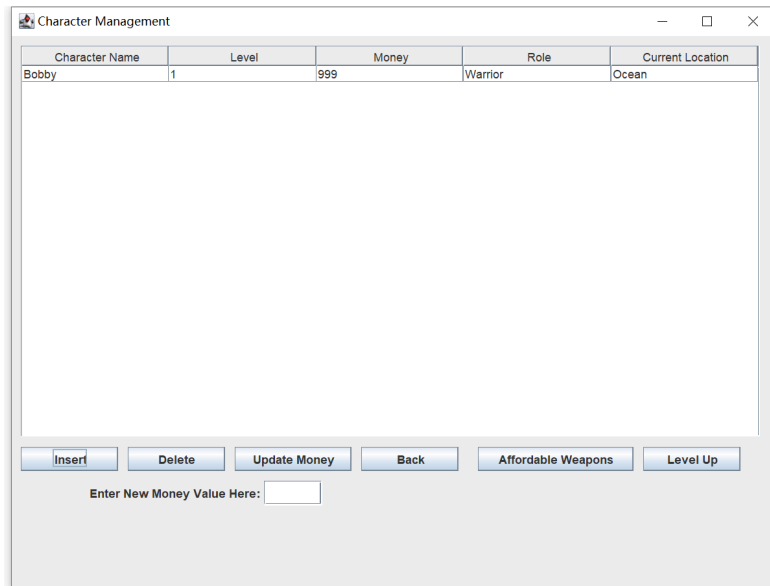
Below the table, there are several buttons: "Insert", "Delete", "Update Money", "Back", "Affordable Weapons", and "Level Up". At the bottom, there is a text input field labeled "Enter New Money Value Here:".

Queries: DELETE:

- User can delete characters;

Query locationDataBaseConnectionHandler:310

Before:



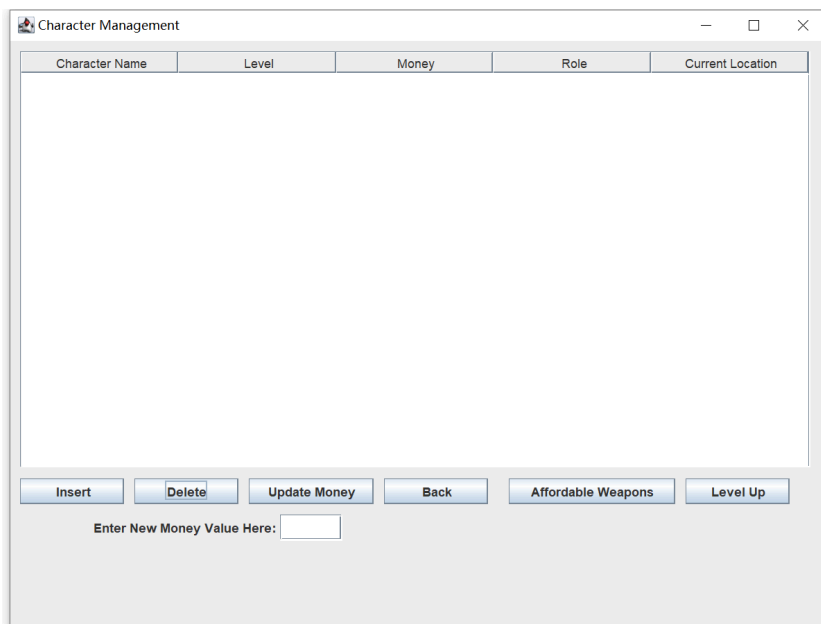
The screenshot shows a window titled "Character Management" with a table containing one character, Bobby. The table has columns for Character Name, Level, Money, Role, and Current Location. Below the table are buttons for Insert, Delete, Update Money, Back, Affordable Weapons, and Level Up. There is also a text input field labeled "Enter New Money Value Here:".

Character Name	Level	Money	Role	Current Location
Bobby	1	999	Warrior	Ocean

Buttons: Insert, Delete, Update Money, Back, Affordable Weapons, Level Up

Enter New Money Value Here:

After:



The screenshot shows the same "Character Management" window, but the table is now empty. The buttons and input field remain the same.

Character Name	Level	Money	Role	Current Location
----------------	-------	-------	------	------------------

Buttons: Insert, Delete, Update Money, Back, Affordable Weapons, Level Up

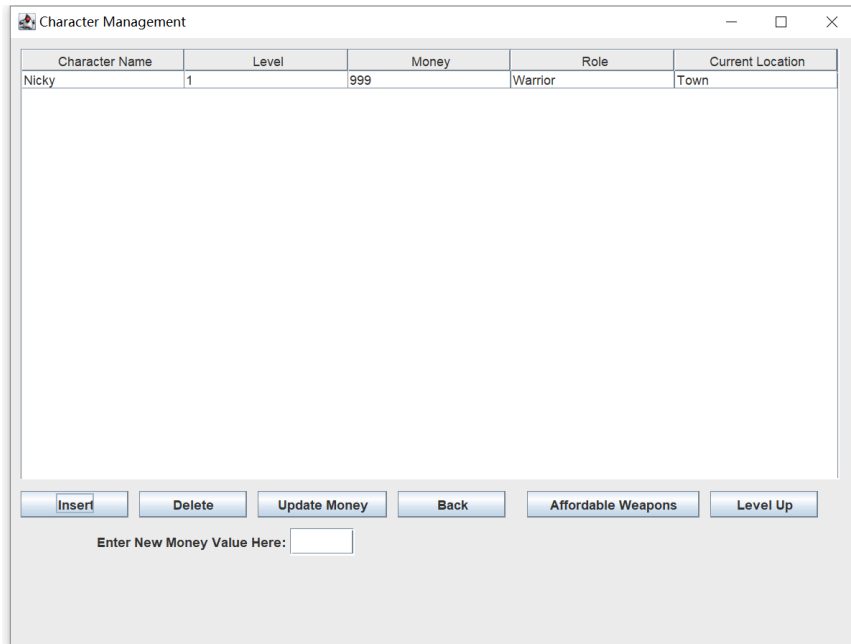
Enter New Money Value Here:

Queries: UPDATE:

- User can update character's money value;

Query location DataBaseConnectionHandler:330

Before:



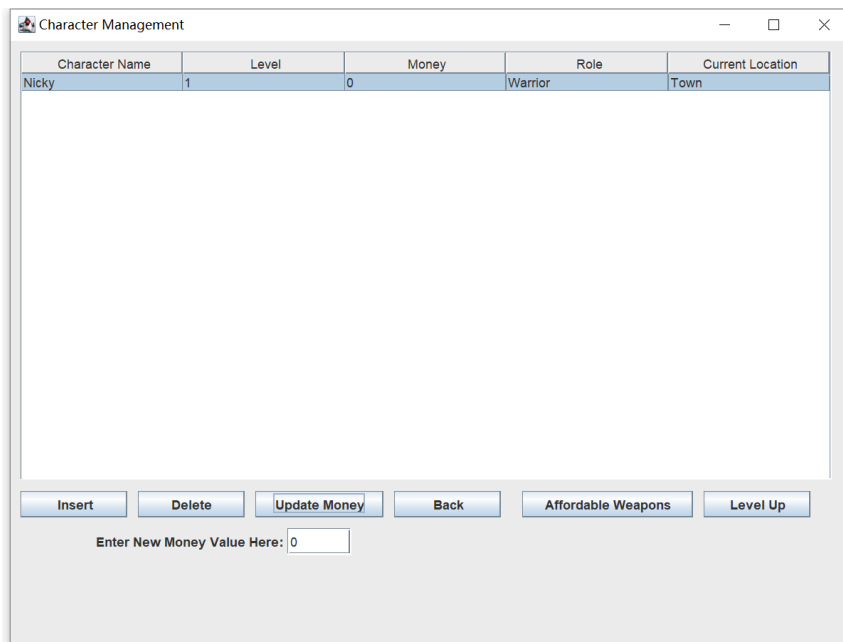
The screenshot shows a window titled "Character Management" with a table containing one row: Nicky, Level 1, Money 999, Role Warrior, Current Location Town. Below the table are buttons for Insert, Delete, Update Money, Back, Affordable Weapons, and Level Up. At the bottom, there is a label "Enter New Money Value Here:" followed by an empty text input field.

Character Name	Level	Money	Role	Current Location
Nicky	1	999	Warrior	Town

Buttons: Insert, Delete, Update Money, Back, Affordable Weapons, Level Up

Enter New Money Value Here:

After:



The screenshot shows the same "Character Management" window, but the money value for Nicky has been updated to 0. The text input field at the bottom now contains the value "0".

Character Name	Level	Money	Role	Current Location
Nicky	1	0	Warrior	Town

Buttons: Insert, Delete, Update Money, Back, Affordable Weapons, Level Up

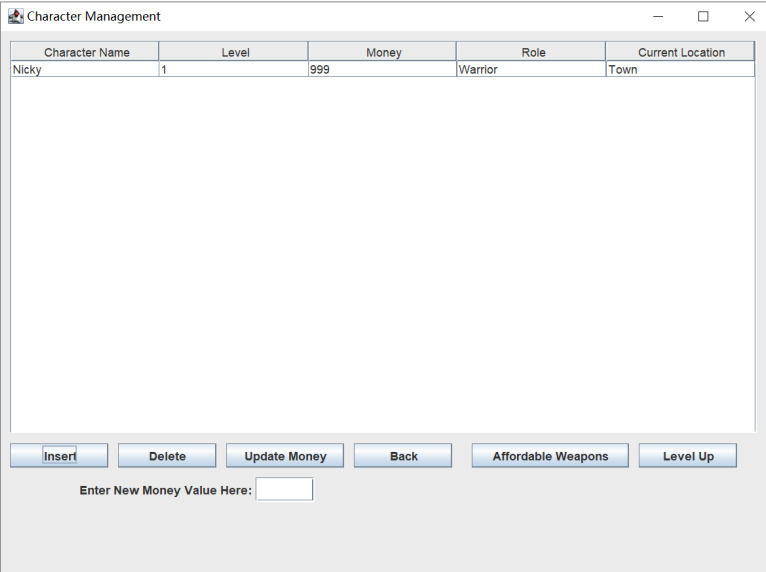
Enter New Money Value Here:

Queries: Selection:

- User can view the weapon that the character can afford to buy.
- select weaponName from weapon, character_info, where price < money.

Query locationDataBaseConnectionHandler:683

Before:



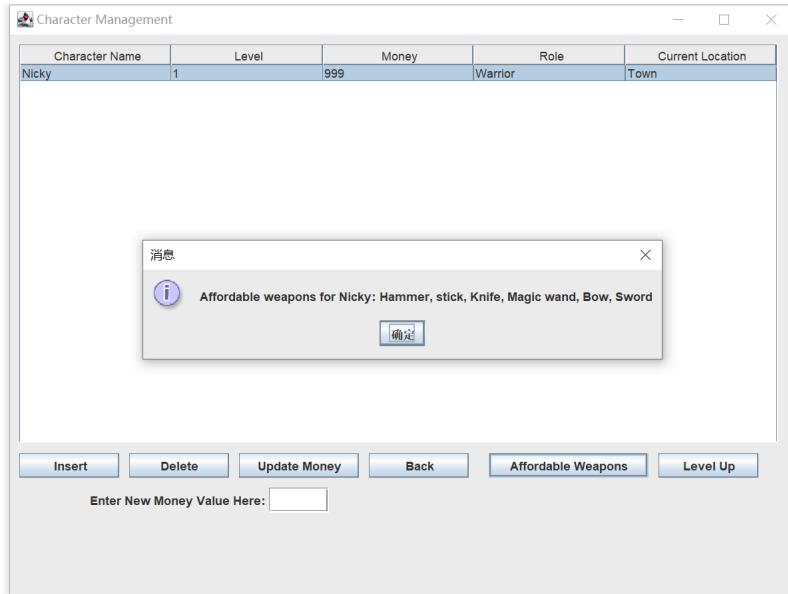
The screenshot shows a window titled "Character Management" with a table containing one character. The table has five columns: Character Name, Level, Money, Role, and Current Location. The data row shows "Nicky" with Level 1, Money 999, Role Warrior, and Current Location Town. Below the table are several buttons: Insert, Delete, Update Money, Back, Affordable Weapons, and Level Up. At the bottom, there is a text input field labeled "Enter New Money Value Here:".

Character Name	Level	Money	Role	Current Location
Nicky	1	999	Warrior	Town

Buttons: Insert, Delete, Update Money, Back, Affordable Weapons, Level Up

Enter New Money Value Here:

After:

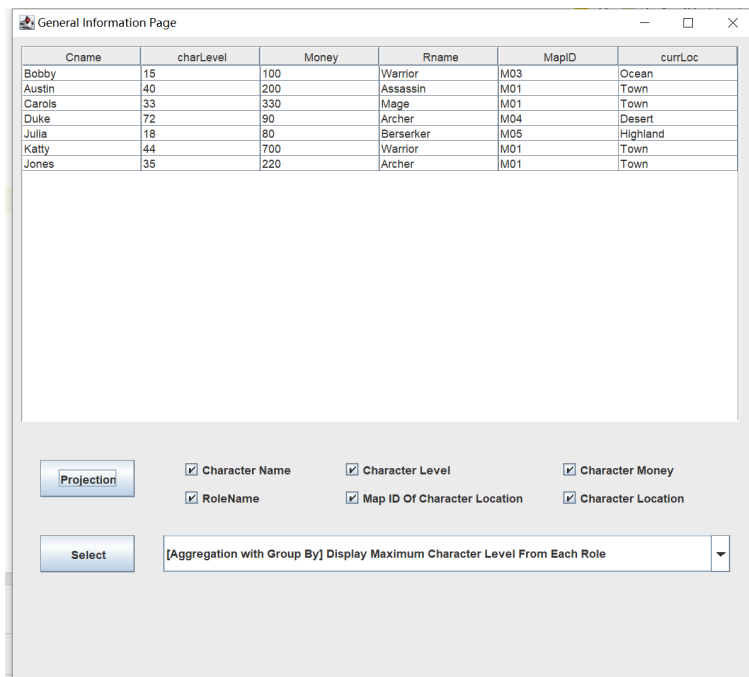


Queries: Projection:

- User can choose to view any attributes in character_info

Query locationDataBaseConnectionHandler:724

Before:



After:

General Information Page

Cname	Rname
Bobby	Warrior
Austin	Assassin
Carols	Mage
Duke	Archer
Julia	Berserker
Katty	Warrior
Jones	Archer

☒ Character Name
 ☐ Character Level
 ☐ Character Money
☒ RoleName
 ☐ Map ID Of Character Location
 ☐ Character Location

[Aggregation with Group By] Display Maximum Character Level From Each Role

Queries: Join:

- User can view a new table with character table join by weapons table.

Query locationDataBaseConnectionHandler:783

Before:

General Information Page

输入

?

Please enter the character name:

☐ Character Name
 ☐ Character Level
 ☐ Character Money
☐ RoleName
 ☐ Map ID Of Character Location
 ☐ Character Location

[Join] Display Character's Role And Weapon Information

Character_Info table

	CNAME	CHARLEVEL	MONEY	RNAME	MAPID	CURRLOC
1	Bobby	15	100	Warrior	M03	Ocean
2	Austin	40	200	Assassin	M01	Town
3	Carols	33	330	Mage	M01	Town
4	Duke	72	90	Archer	M04	Desert
5	Julia	18	80	Berserker	M05	Highland
6	Katty	44	700	Warrior	M01	Town
7	Jones	35	220	Archer	M01	Town

Weapons table

	WEAPONID	WPDAMAGE	PRICE	RNAME	WNAME
1	1000000001	100	150	Warrior	Hammer
2	1000000001	1	150	Warrior	stick
3	1000000002	20	200	Assassin	Knife
4	1000000002	51	250	Mage	Magic wand
5	1000000002	20	300	Archer	Bow
6	1000000002	20	400	Berserker	Sword

After:

Character Name	Level	Role	Weapon Name	Weapon Damage	Weapon Price
Bobby	15	Warrior	Hammer	100	150
Bobby	15	Warrior	stick	1	150

Projection

☐ Character Name
 ☐ Character Level
 ☐ Character Money
 ☐ RoleName
 ☐ Map ID Of Character Location
 ☐ Character Location

Select

[Join] Display Character's Role And Weapon Information

Queries: Aggregation with Group By:

- Find character names with the highest level for each role

Query locationDataBaseConnectionHandler:611

Before:

General Information Page

Projection

☐ Character Name
 ☐ Character Level
 ☐ Character Money
 ☐ RoleName
 ☐ Map ID Of Character Location
 ☐ Character Location

Select

[Aggregation with Group By] Display Maximum Character Level From Each Role

Character_Info table

	CNAME	CHARLEVEL	MONEY	RNAME	MAPID	CURRLOC
1	Bobby	15	100	Warrior	M03	Ocean
2	Austin	40	200	Assassin	M01	Town
3	Carols	33	330	Mage	M01	Town
4	Duke	72	90	Archer	M04	Desert
5	Julia	18	80	Berserker	M05	Highland
6	Katty	44	700	Warrior	M01	Town
7	Jones	35	220	Archer	M01	Town

Roles table

	RNAME
1	Archer
2	Assassin
3	Berserker
4	Mage
5	Warrior

After:

General Information Page

Role Name	Max Character Level
Assassin	40
Warrior	44
Archer	72
Berserker	18
Mage	33

Projection

☐ Character Name
☐ Character Level
☐ Character Money

☐ RoleName
☐ Map ID Of Character Location
☐ Character Location

Select

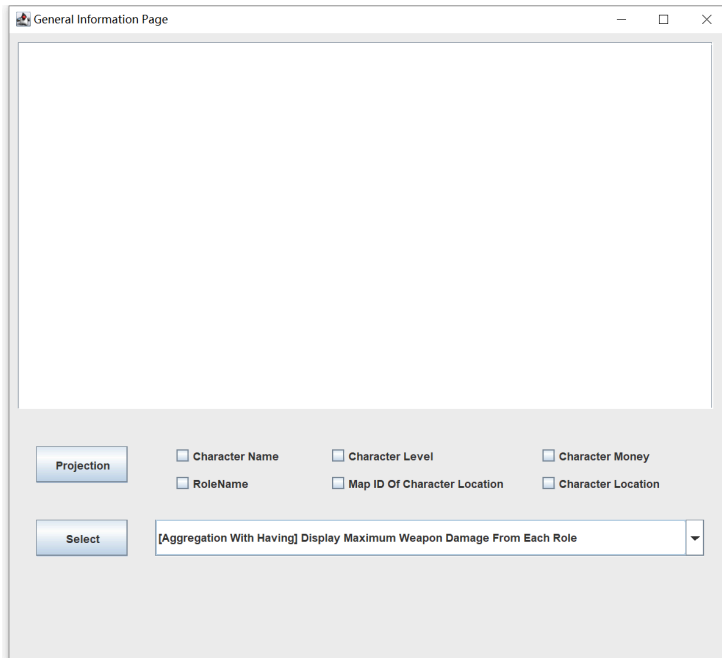
[Aggregation with Group By] Display Maximum Character Level From Each Role

Queries: Aggregation with Having:

- Find the maximum damage of weapons with maximum weapon damage larger than 100 for each role

Query locationDataBaseConnectionHandler:659

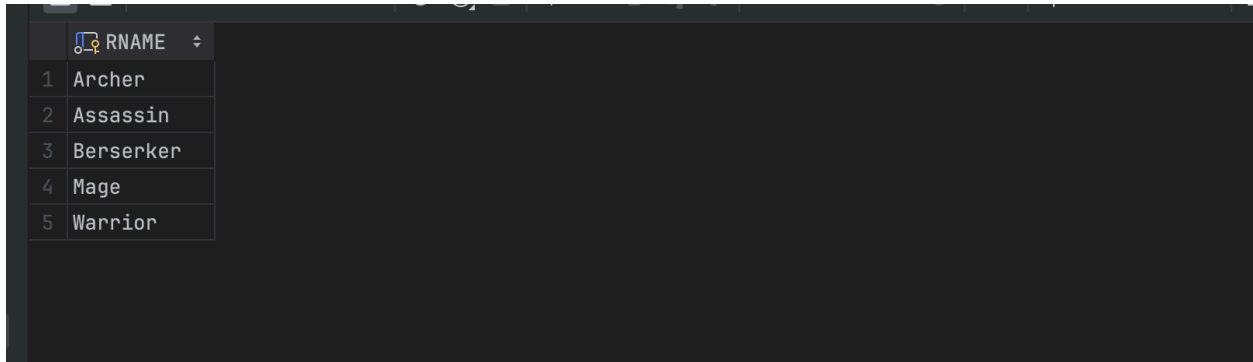
Before:



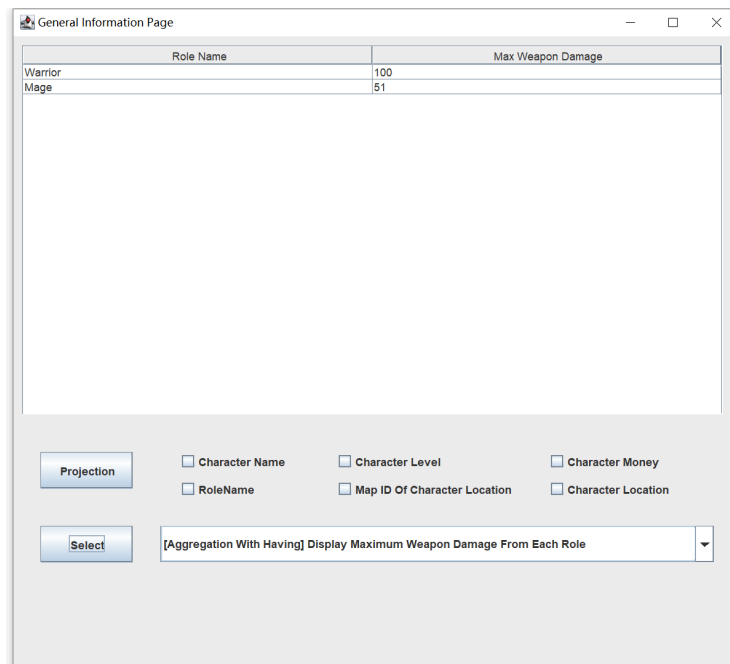
Weapons table

	<input type="checkbox"/> WEAPONID	<input type="checkbox"/> WPDAMAGE	<input type="checkbox"/> PRICE	<input checked="" type="checkbox"/> RNAME	<input type="checkbox"/> WNAME
1	1000000001	100	150	Warrior	Hammer
2	1000000001	1	150	Warrior	stick
3	1000000002	20	200	Assassin	Knife
4	1000000002	51	250	Mage	Magic wand
5	1000000002	20	300	Archer	Bow
6	1000000002	20	400	Berserker	Sword

Roles table



After:



Queries: Nested Aggregation with Group By:

- For each role, find the average weapon damage that is less than other roles' corresponding weapon damage
- Query locationDataBaseConnectionHandler:824

Before:

General Information Page

Projection

☐ Character Name

☐ RoleName

☐ Character Level

☐ Map ID Of Character Location

☐ Character Money

☐ Character Location

Select

[Nested aggregation] Display roles with the minimum average weapon damage

Weapons table

	WEAPONID	WPDAMAGE	PRICE	RNAME	WNAME
1	1000000001	100	150	Warrior	Hammer
2	1000000001	1	150	Warrior	stick
3	1000000002	20	200	Assassin	Knife
4	1000000002	51	250	Mage	Magic wand
5	1000000002	20	300	Archer	Bow
6	1000000002	20	400	Berserker	Sword

Roles table

	RNAME
1	Archer
2	Assassin
3	Berserker
4	Mage
5	Warrior

After:

Role Name	Average Damage
Assassin	20
Archer	20
Berserker	20

Projection

☐ Character Name
 ☐ Character Level
 ☐ Character Money

☐ RoleName
 ☐ Map ID Of Character Location
 ☐ Character Location

Select

[Nested aggregation] Display roles with the minimum average weapon damage

Queries: Division:

- Find the maps that have all the characters with money amount larger than 100
- Query locationDataBaseConnectionHandler:635

Before:

General Information Page

Projection

☐ Character Name
 ☐ Character Level
 ☐ Character Money

☐ RoleName
 ☐ Map ID Of Character Location
 ☐ Character Location

Select

[Division] Show Map That Includes ALL Characters With Money Value Over 100

Character Info table

	CNAME	CHARLEVEL	MONEY	RNAME	MAPID	CURRLOC
1	Bobby	15	100	Warrior	M03	Ocean
2	Austin	40	200	Assassin	M01	Town
3	Carols	33	330	Mage	M01	Town
4	Duke	72	90	Archer	M04	Desert
5	Julia	18	80	Berserker	M05	Highland
6	Katty	44	700	Warrior	M01	Town
7	Jones	35	220	Archer	M01	Town

Map table

	MAPID	MAPNAME
1	M01	Town
2	M02	Forest
3	M03	Ocean
4	M04	Desert
5	M05	Highland

After:

General Information Page

Map ID

Map Name

M01

Town

Projection

☐ Character Name

☐ Character Level

☐ Character Money

☐ RoleName

☐ Map ID Of Character Location

☐ Character Location

Select

[Division] Show Map That Includes ALL Characters With Money Value Over 100