

ItemMuW Project

About the Project

ItemMuw is a Java program that could help gamers sell their gaming-related products. This program ensures that gamers can sell their products safely because many scammers have lately. The program is created from scratch using Eclipse's Java IDE. Below is a more in-depth explanation of this project.

How does it Work?

The program consists of three menus, insert product, buy a product, and exit. The explanation of each menu goes as follows,

```
|
+=====+
|          MENU          |
+=====+
| 1. Insert Product      |
| 2. Buy Product         |
| 3. Exit                |
+=====+
>>
```

- **Insert Product (Menu 1)**

If the user chooses menu 1, firstly, the program will ask the user to input their product name. The program then will validate the program. The validation will be that the product name must be between 6 and 20 characters.

```
Input product name [6-20 characters] :
Input product name [6-20 characters] : abcd
Input product name [6-20 characters] : Dragon Lore
```

Next, the program will ask the user to input their product type and validate it. The validation is case-sensitive. The input must be either Currency, Item, or Account.

```
Input product type [Currency | Item | Account] :
Input product type [Currency | Item | Account] : weapon
Input product type [Currency | Item | Account] : currency
Input product type [Currency | Item | Account] : Currency
```

Next, the program will ask the user to input their product price. The price must be between 10.000 and 1.000.000.

```
Input product price [10000 - 1000000] : 1000
Input product price [10000 - 1000000] : 2000000
Input product price [10000 - 1000000] : 150000
```

Lastly, the program will ask for the product quantity. The quantity must be between 1 and 50. After the user adds the amount, the program will save the product data in an array.

```
Input product quantity [1 - 50] : 0
Input product quantity [1 - 50] : 55
Input product quantity [1 - 50] : 25
Product added !! Press enter to continue..
```

- **Buy Product (Menu 2)**

When the user chooses menu 2, the program will display a message “There’s no product” if there isn’t any product. Otherwise, the program will display all product data that consists of product name, product type, product price, product quantity, and subtotal.

```
>> 2
There's no product..
```

```
>> 2
+-----+-----+-----+-----+
| NO | Name                | Price    | Quantity | Sub Total |
+-----+-----+-----+-----+
| 1 | Dragon Lore          | 1000000  | 1         | 1000000   |
+-----+-----+-----+-----+
```

Next, the program will ask the user if they want to buy all products. The input is case-sensitive. The user must input either Y or N.

```
Buy All? [ Y/N ] :Y
```

Lastly, if the user chooses to buy all, the program displays a total price, discount, tax, and receipt.

```
+-----+
|          Receipt          |
+-----+
| Total                    : 1000000 |
| Discount [ 10% ]        : 100000  |
| Tax [ 5% ]               : 50000   |
| Grand Total              : 950000  |
+-----+
| Thank You for purchasing !! |
+-----+
```

- **Exit (Menu 3)**

If the user chooses menu 3, the program will be closed.

```
>> 3
-^ Thanks for using this program... ^-
```

My Role Within the Project

I create this program as an exercise for the upcoming quiz. Since this program is developed by myself from scratch, I play the developer role.

The Difficulties and How to Face it

I found understanding and solving the syntax and logic complicated because this was the first coding project I had. To overcome this issue, I look at the project's video explanation and continuously do trial and error to find the correct syntax for this project