

City University of Hong Kong
MS3111 Quantitative Business Analysis with Visual Basic
for Applications
2020/2021 Semester B



C03 – Group 60
Group Members:
Poon Tsz Chun 56312045
Wong Sai Kit 55208188
Chong Wing Pan 55696641
Lou Ho Lam 56303221

Table of contents

Introduction	2
Assumptions	2
General Flow of Application	3
Data Structure	3
4.1 New Customer Information worksheet	3
4.2 Product worksheet	3-4
4.3 Order worksheet	4
4.4 Sale information worksheet	4
Excel VBA Application	5
Userform (1): Start _ Step1	5
Userform (2) Welcome _ Step2	5-7
Userform (3): New_Customers_Step3	8-13
Userform (4): Make_Orders_Step4	123-16
Userform (5): Employees_login_Step5	167-19
Userform (6): Choose_Your_Role_Step6	19
Userform (7): Staff_Control_Panel_Step7	20-26
Userform (8): Staff_Control_Panel_Step8	26-29
Appendix	30-36

Introduction

Apple is an American electronic product brand established in 1976. The company provides various products in their online platform, such as iPhone and iPad. In hope of increasing the efficiency of their manager and offering a user-friendly platform to customers, we designed a new system to control various daily tasks.

we provide different roles to users: Customer or Staff (including staff or manager).

For “customer”, the registered customers may login by inputting the correct password and email. And new customers need to create an account. They can make an order by choosing products and indicating the quantity after logging in. Then, the staff will receive the confirmed order to finish the transaction.

For “Employee”, users also need to select their role, which are “staff” and “manager”. They can see various order details ranging from purchased items to shipping details through choosing “staff”. Moreover, they can create different reports such as profit and sale performance through choosing “Manager”. It is considered that the system can directly increase the efficiency of daily operation.

This system is designed for Apple in Hong Kong.

Assumptions

1. Employee account login information:

- a. ID == em1
- b. Password == apple

2. Customer login example:

amy@gmail.com
pw: amy

coco@gmail.com
pw: coco

or any accounts on the customer information list

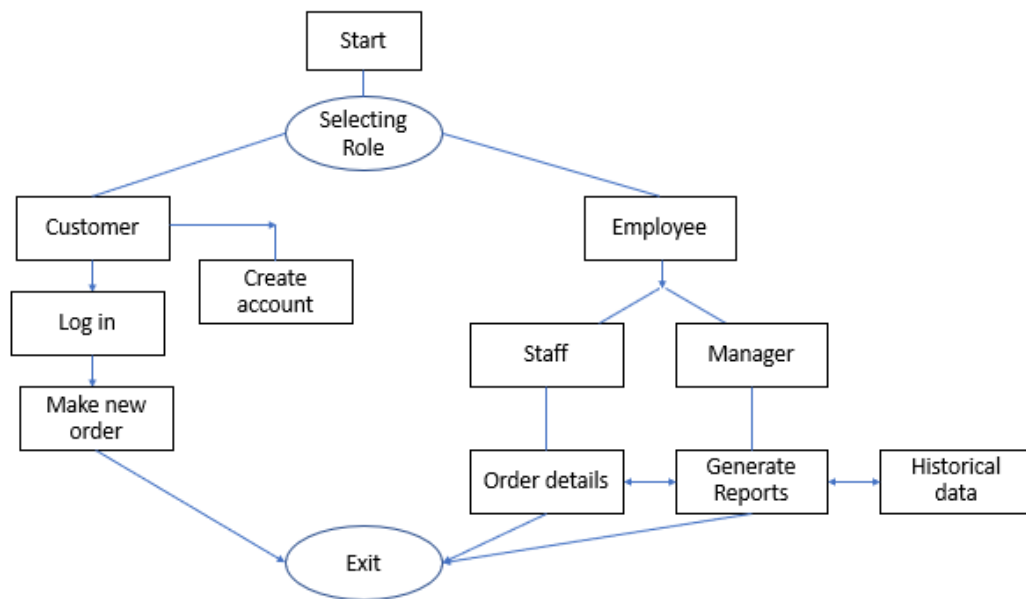
3. Userform4 placing order:

Once customers are being passed to page2 of the userform, we assume that our customers will return to the first page if they are going to add extra products to the shopping cart.

4. Viewing manager's reports:

The manager can only leave the report worksheet by pressing the button in the corner (Back to Manager Report) then click the “Exit” button” when they generate a chart.

General Flow of Application



When users click the “Start” Button, then the system will ask them to select a role who are “Customer” and “Employee”. For the “Customer” button, they will enter the customer interface to log in. If they do not have an account, they need to create an account firstly, and they can go back to the login page after the account is created. Next, users can make a new order through the system. For the “Employee” button, they also need to log in with an employee ID, then they can select the role of “Staff” or “Manager”. They can see the order details or generate a report using the “Employee” panel.

Data Structure

4.1 Worksheet(New Customer Information)

This worksheet is about customer information. It is notable that the number of rows will be increased when the user creates an account. Currently, we have more than 200 registered customers in the list, and more new customers will be added if accounts are created.

Customer ID	First Name	Last Name	Email	Gender	Phone Number
Address	Password	Cardholder's Name	Credit card number	Expiry Date	

4.2 Worksheet(Product)

This worksheet is about the product information that Apple will offer. The data on this worksheet will be used in other sections to do more calculation.e.g., manager reports.

Products Code	Product Name	Price(\$)	Cost(\$)
---------------	--------------	-----------	----------

Products	iWatch	iPhone	iPad	Macbook
----------	--------	--------	------	---------

4.3 Worksheets (for making and storing data of new Orders)

The “New order” worksheet, “packing” worksheet, and “shipped” worksheets are all correlated.

worksheet(new order):

Order Name	Date	Name	Phone Number	Address	Total Qty	Subtotal(\$)
------------	------	------	--------------	---------	-----------	--------------

worksheet(packing):

Order Name	Date	Name	Phone Number	Address	Total Qty	Subtotal(\$)
------------	------	------	--------------	---------	-----------	--------------

It is notable that the data from the worksheet(new order) will be transferred to the worksheet(packing) when the order is packed.

worksheet(shipped):

Order Name	Date	Name	Phone Number	Address	Total Qty	Subtotal(\$)	Expected Delivery Date
------------	------	------	--------------	---------	-----------	--------------	------------------------

For “Shipped” worksheet, there is almost the same as the other two worksheets, but it has more columns which is “Expected Delivery Date”. The data from the “Packing” worksheet will be transferred to the “Shipped” worksheet when the order has already shipped.

4.4 Sales related information worksheets

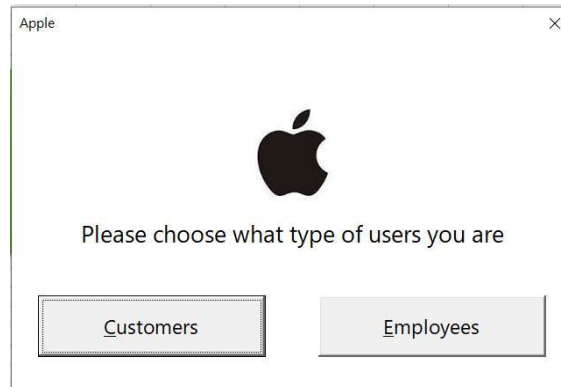
This worksheet is about “Manager” worksheet which includes “Order Name”, “Name”, “Qty”, “Product”, “Cost”, “Transaction Date”, “Revenue”, “Profit” and “Performance for Product”. It is notable that the rows will be increased by the order numbers.

Excel VBA Application

Userform 1: Start _ Step1



There is a command button of “Start”. The role userform will be shown after the user clicks this button.



The system will provide two roles for users to select which are “Customers” and “Employees”. If the user clicks on the “Customers” button, then he/she can make a new order under the customer panel. Besides, if the user clicks on the “Employees” button, the user can check the order details and generate various reports under the manager panel and staff panel.

Userform 2: Welcome _ Step2



In this step, there are different functions for existing customers and new customers. For existing customers who are already having an account, they can enter their email address and password to log in directly. For new customers, they can click on the “Create an Account” button to create an account in New_Customer_Step3.

In the login process, the system assumes that there are 4 possible situations that may happen.

Welcome to Apple!


Existing Customer

New Customer

Email Address:

Password:

Sign In Error

 Please enter your email address and password!

```

If txt_UID_EC = "" Then
    MsgBox "Please enter your email address and password!", vbOKOnly + vbExclamation, "Sign In Error"
    txt_PW_EC = ""
    txt_UID_EC.SetFocus
    txt_PW_EC.BackColor = rgbYellow
    txt_UID_EC.BackColor = rgbYellow
    Exit Sub
End If

If txt_PW_EC = "" Then
    MsgBox "Please enter your password!", vbOKOnly + vbExclamation, "Sign In Error"
    txt_UID_EC.SetFocus
    txt_PW_EC.BackColor = rgbYellow
    Exit Sub
End If

```

First, if the customer does not enter anything in the text box, then the system will show up a message box that says "Please enter your email address and password!", and the system will also highlight the text box to remind the user.

Innisfree

You have been successfully log in !

```

Rowcount = 2
Password = False
Do While Cells(Rowcount, 4).Value <> ""
    If Cells(Rowcount, 4).Value = txt_UID_EC.Value Then
        If Cells(Rowcount, 8) = txt_PW_EC.Value Then
            Password = True
            Welcome_Step2.Hide
            MsgBox ("You have been successfully log in !")
            Make_order_Step4.Show
            txt_UID_EC = ""
            txt_PW_EC = ""
            Exit Sub
        End If
    End If
    Rowcount = Rowcount + 1
End While

```

Next, if the customer enters both the available email address and password in the text box. The data of the email address will be matched with the data of password in the new customer information worksheet. Besides, in order to make our system more user-friendly, the system will transfer the email address to be the upper case. Thus, customers can log in successfully be it uppercase to lowercase.

Welcome

X

Welcome to Apple!


Existing Customer

New Customer

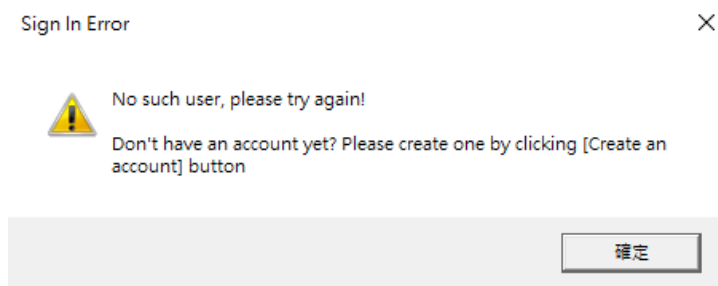
Email Address:

Password:

Sign In Error

 You entered a wrong password. Please try again!

Then, if the customer enters the available email address but gives the wrong password, it will not sign in



successfully. Due to the email address being unable to match with password the system will show a message box that says “You entered the wrong password. Please try again!”, and the system also will clear and highlight the password textbox as yellow to remind the customer.

```

Do While Cells(Rowcount, 4).Value <> ""
    If Cells(Rowcount, 4).Value = txt_UID_EC.Value Then
        If Cells(Rowcount, 8) = txt_PW_EC.Value Then
            Password = True
            Welcome_Step2.Hide
            MsgBox ("You have been successfully log in !")
            Make_order_Step4.Show
            txt_UID_EC = ""
            txt_PW_EC = ""
            Exit Sub
            Exit Do
        Else
            Password = False
            MsgBox "You entered a wrong password. Please try again!", vbOKOnly + vbExclamation, "Sign In Error"
            txt_PW_EC = ""
            txt_UID_EC = ""
            Exit Sub
        End If
    Else
        Rowcount = Rowcount + 1
    End If
Loop

MsgBox "No such user, please try again!" & Chr(13) & Chr(13) & _
"Don't have an account yet? Please create one by clicking [Create an account] button", _
vbOKOnly + vbExclamation, "Sign In Error"
txt_UID_EC = ""
txt_PW_EC = ""

```

Finally, the customer enters both the wrong email address and password. As the email address cannot be found in the new customer information worksheet, the message box will show that “No such user, please try again...” and the textbox will be cleared.

Userform 3: New_Customers_Step3

For every new customer, we offer a new customer userform to save their personal information.

In this system, we used additem functions for customers in email, genders, and phone area code. they can choose the appropriate option in combo boxes rather than typing by themselves.

New Customer Registration



*All information is required.

First Name

Last Name

Password

*mix with letters & numbers

Email

Gender

Phone Number

Address

Credit Card Information

Cardholder Name

Credit Card Number

Expiry Date (YY/MM)

Clear All

Enter

Back To Login

Exit



```

Private Sub UserForm_Initialize()
    txtFirstName.Value = ""
    txtLastName.Value = ""
    txtPassword.Value = ""
    txtEmail.Value = ""
    txtPhoneNumber.Value = ""
    txtAddress.Value = ""
    txt_Cd_name = ""
    txt_Cd_num = ""
    txt_Cd_year = ""
    txt_Cd_month = ""

    With cboMail
        .Clear
        .AddItem "@gmail.com"
        .AddItem "@yahoo.com"
        .AddItem "@hotmail.com"
        .AddItem "@outlook.com"
        .AddItem "@live.com"
    End With

    With cboGender
        .Clear
        .AddItem "Male"
        .AddItem "Female"
    End With

    With cboArea
        .Clear
        .AddItem "852"
        .AddItem "886"
        .AddItem "82"
        .AddItem "81"
        .AddItem "66"
        .AddItem "86"
    End With

    txtFirstName.SetFocus
End Sub

```

```

Private Sub cmbBackToLogin_Click()
    Unload Me
    Welcome_Step2.Show
End Sub

```

We used additem functions for customers in email, genders and phone area code; they can choose the appropriate option in combo boxes rather than typing by themselves.

```

Private Sub cmbExit_Click()
    Unload Me
End Sub

```

```

Private Sub cmbClear_Click()
    cboMail.Value = ""
    cboArea.Value = ""
    Call UserForm_Initialize
End Sub

```

```

Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)
    If CloseMode = vbFormControlMenu Then
        MsgBox "Please click Exit button to close.", vbInformation, "Attention"
        Cancel = True
    End If
End Sub

```

This code is for the “Clear All” and “Exit” button. the customer is allowed to clear data or exit the userform through clicking these two buttons. If the customer types in something wrong, then they can click the “Clear All” to directly delete all the data. Besides, as we do not allow them to exit the userform through clicking the

```

Private Sub txt_Cd_name_AfterUpdate()
If txt_Cd_name.Value <> "" Then
txt_Cd_name.BackColor = rgbWhite
End If
End Sub

Private Sub txt_Cd_num_AfterUpdate()
If txt_Cd_num.Value <> "" Then
txt_Cd_num.BackColor = rgbWhite
End If
End Sub

Private Sub txt_Cd_year_AfterUpdate()
If txt_Cd_year.Value <> "" Then
txt_Cd_year.BackColor = rgbWhite
txt_Cd_month.BackColor = rgbWhite
End If
End Sub

Private Sub txtAddress_AfterUpdate()
If txtAddress.Value <> "" Then
txtAddress.BackColor = rgbWhite
End If
End Sub

Private Sub txtEmail_AfterUpdate()
If txtEmail.Value <> "" Then
txtEmail.BackColor = rgbWhite
cboMail.BackColor = rgbWhite
End If
End Sub

Private Sub txtFirstName_AfterUpdate()
If txtFirstName.Value <> "" Then
txtFirstName.BackColor = rgbWhite
End If
End Sub

Private Sub txtLastName_AfterUpdate()
If txtLastName.Value <> "" Then
txtLastName.BackColor = rgbWhite
End If
End Sub

Private Sub txtPassword_AfterUpdate()
If txtPassword.Value <> "" Then
txtPassword.BackColor = rgbWhite
End If
End Sub

Private Sub txtPhoneNumber_AfterUpdate()
If txtPhoneNumber.Value <> "" Then
cboArea.BackColor = rgbWhite
txtPhoneNumber.BackColor = rgbWhite
End If
End Sub

```

close button, they only can exit the userform through clicking the “Exit” button. Finally, we also offer a “Back To Login” button to help customers go back to the login panel after creating an account.

```

Private Sub cboArea_AfterUpdate()
If cboArea.Value <> "" Then
cboArea.BackColor = rgbWhite
txtPhoneNumber.BackColor = rgbWhite
End If
End Sub

Private Sub cboGender_AfterUpdate()
If cboGender.Value <> "" Then
cboGender.BackColor = rgbWhite
End If
End Sub

Private Sub cboMail_AfterUpdate()
If cboMail.Value <> "" Then
txtEmail.BackColor = rgbWhite
cboMail.BackColor = rgbWhite
End If
End Sub

Private Sub cmbBackToLogin_Click()
Unload Me
Welcome_Step2.Show
End Sub

```

These commands are just for checking whether they have input the data or not. if the customers enter something in the textboxes, the textbox will still be the original color which is white color.

```
If Trim(txt_Cd_name) = "" Then
    MsgBox "You did not enter cardholder name.", vbExclamation, "Attention"
    txt_Cd_month.SetFocus
    txt_Cd_month.BackColor = rgbYellow
Exit Sub
ElseIf IsNumeric(txt_Cd_name.Value) Then
    MsgBox "Invaild cardholder name."
    txt_Cd_month.SetFocus
    txt_Cd_month.BackColor = rgbYellow
Exit Sub
End If

If txt_Cd_num.Value = "" Then
    MsgBox "You did not enter credit card number.", vbExclamation, "Attention"
    txt_Cd_num.SetFocus
    txt_Cd_num.BackColor = rgbYellow
Exit Sub
ElseIf Not IsNumeric(txt_Cd_num.Value) Then
    MsgBox "You entered an invaild credit card number.", vbExclamation, "Attention"
    txt_Cd_num.SetFocus
    txt_Cd_num.BackColor = rgbYellow
Exit Sub
End If

If Len(txt_Cd_num) = 16 Then
Else
    MsgBox "You entered an invaild credit card number.", vbExclamation, "Attention"
    txt_Cd_num.SetFocus
    txt_Cd_num.BackColor = rgbYellow
Exit Sub
End If

If txt_Cd_year.Value < 21 Then
    MsgBox "You entered the wrong expiry date.", vbExclamation, "Attention"
    txt_Cd_year.SetFocus
    txt_Cd_year.BackColor = rgbYellow
Exit Sub
ElseIf txt_Cd_year.Value = "" Then
    MsgBox "You did not enter Expiry Date(YY).", vbExclamation, "Attention"
    txt_Cd_year.SetFocus
    txt_Cd_year.BackColor = rgbYellow
Exit Sub
End If

If txt_Cd_month.Value > 12 Then
    MsgBox "You entered the wrong expiry date.", vbExclamation, "Attention"
    txt_Cd_month.SetFocus
    txt_Cd_month.BackColor = rgbYellow
Exit Sub
ElseIf txt_Cd_month.Value = "" Then
    MsgBox "You did not enter Expiry Date(MM).", vbExclamation, "Attention"
    txt_Cd_month.SetFocus
    txt_Cd_month.BackColor = rgbYellow
Exit Sub
End If

If Not IsNumeric(txt_Cd_year.Value) Or Not IsNumeric(txt_Cd_month.Value) Then
    MsgBox "Invalid input of Expiry Date.", vbExclamation, "Attention"
    txt_Cd_year.SetFocus
    txt_Cd_year.BackColor = rgbYellow
    txt_Cd_month.BackColor = rgbYellow
Exit Sub
End If
```

These commands are used to avoid customers inputting invalid data. For Name, the user must type in letters. For Email addresses and passwords, the user can type in letters or numbers. For phone number and area code, the user only can type in numbers. Moreover, we also set some conditions for typing credit card information. For expiry year, the year must be larger than 21. For the month, the month must be smaller than 12. For the length of the credit card numbers, the numbers must be 16 digits. Thus, these commands can help customers to know that where they type in an invalid data directly.

```

With Sheets("New Customer Information")
.Cells(nextRec, 1).Value = nextRec + 9999
.Cells(nextRec, 2).Value = txtFirstName.Value
.Cells(nextRec, 3).Value = txtLastName.Value
.Cells(nextRec, 4).Value = txtEmail.Value & cboMail.Value
.Cells(nextRec, 5).Value = cboGender.Value
.Cells(nextRec, 6).Value = "(" & cboArea.Value & ")" & txtPhoneNumber.Value
.Cells(nextRec, 7).Value = txtAddress.Value
.Cells(nextRec, 8).Value = txtPassword.Value
.Cells(nextRec, 10).Value = txt_Cd_name.Value
.Cells(nextRec, 11).Value = "" & txt_Cd_num.Value
.Cells(nextRec, 12).Value = txt_Cd_year.Value & "/" & txt_Cd_month.Value
End With

MsgBox "You have been successfully registered! You can log in now!", , "Apple"
Unload Me
Welcome_Step2.Show

```

	A	B	C	D	E	F	G	H	I	J	K	L
1	Customer ID	First Name	Last Name	Email	Gender	Phone Number	Address	Password		Cardholder's Name	Credit Card Number	Expiry Date (YY/MM)
2	10001	Sarge	Wife	swife0@zdnnet.com	Male	(688) 7221101	5812 Sloan Trail	iWGB0XnXL		Sarge Wife	1234567800003320	25/05
3	10002	Monique	Fairweather	mfairweather1@webn	Male	(690) 9380293	35677 Muir Alley	DR8QcZ5bWrch		Monique Fairweather	1234567890123450	24/03
4	10003	Kania	Mattheis	kmattheis2@goo.ne.jp	Male	(877) 3639874	00 Division Trail	Xw7Pluy7JLJ		Kania Mattheis	1234123412341234	22/07
5	10004	Kaile	Shatford	kshatford3@yahoo.co	Female	(153) 1865518	30732 Sutteridge Way	tcZDIYHv		Kaile Shatford	1234567800003320	21/07
6	10005	Jenica	Rozenbaum	jrozenbaum4@com.co	Male	(793) 4358981	680 Oak Crossing	xHIng5PMg5s		Jenica Rozenbaum	7777888899996666	25/07
7	10006	Enid	Ferronier	eferronier5@phoca.cz	Female	(339) 3614767	4 Bluestem Park	OS0z5QgRt		Enid Ferronier	1234567800003320	21/04
8	10007	Rozina	Grigoire	rgrigoire6@google.de	Male	(643) 2327886	08676 Loomis Alley	GMEQkpa7		Rozina Grigoire	1234567890123450	26/08
9	10008	Jacquelin	McKechnie	jmckechnie7@trellian	Female	(993) 2201208	96 Spaight Trail	mazy0SU0765		Jacquelin McKechnie	1234123412341234	23/09
10	10009	Rona	Le Teve	rleteve8@eepurl.com	Female	(458) 3181981	6 Esch Junction	z4VWOxMNS		Rona Le Teve	1234567800003320	26/01
11	10010	Casi	Ucceli	cucceli9@desdev.cn	Female	(325) 4125476	9144 Hollow Ridge Pass	epH6q2nXTEN		Casi Ucceli	7777888899996666	26/04
12	10011	Kristoffer	L'Episcopio	klepiscopioa@reuters	Male	(850) 3263843	61 Ruskin Terrace	gPrTgD		Kristoffer L'Episcopio	1234567800003320	24/07
13	10012	Brittany	Gresswood	bgresswoodb@dagonc	Male	(512) 7032025	79 Bluejay Park	5Ab8HL4Pb		Brittany Gresswood	1234567890123450	21/06
14	10013	Kris	Georg	kgeorgc@state.gov	Female	(436) 1709299	5974 Messerschmidt Dri	InYTCPz		Kris Georg	1234123412341234	23/03
15	10014	Cordie	Katte	ckatted@state.tx.us	Male	(120) 2778795	24 Butternut Center	wszyq8B76fQ6		Cordie Katte	1234567800003320	24/06
16	10015	Lorry	Crosseland	lcrosselande@daily	Female	(650) 4725846	8 Eagle Crest Drive	tOYx41gi		Lorry Crosseland	7777888899996666	22/03
17	10016	Renate	Holborn	rholbornf@zimbio.con	Female	(879) 6927995	6740 Corry Drive	tFWIA0		Renate Holborn	1234567800003320	25/08
18	10017	Sylvan	Thacke	sthackeg@hhs.gov	Female	(332) 4239329	49614 Grayhawk Plaza	niaEbqQst		Sylvan Thacke	1234567890123450	24/09
19	10018	Jonie	Fraczak	jfraczakh@uol.com.br	Male	(875) 4445193	299 Stephen Crossing	bkdyxIHTWX		Jonie Fraczak	1234123412341234	26/03
20	10019	Meryl	Zoephel	mzoepheli@edublogs	Male	(186) 6972354	81898 Rusk Crossing	eH4J2jm2gNLFh		Meryl Zoephel	1234567800003320	25/09
21	10020	Kerk	Paike	kpaikex@netlog.com	Male	(344) 2622901	4 Mockingbird Street	AZJWEei		Kerk Paike	7777888899996666	22/05
22	10021	Bridget	Edgerton	bedgertonk@webnodi	Female	(966) 2531419	13 Heath Drive	yiaSI0jtT1Y		Bridget Edgerton	1234567800003320	21/09
23	10022	Barbette	McGettigan	bmcgettigan@smugm	Female	(890) 8485257	5 Mitchell Street	h2nhejFbX		Barbette McGettigan	1234567890123450	25/01
24	10023	Si	Nancarrow	snancarrowm@creativ	Male	(161) 8422057	12390 Emmet Terrace	IHL1ZxqQgo1p		Si Nancarrow	1234123412341234	24/01
25	10024	Ginnie	Skippon	gskipponn@geocities	Female	(394) 7805392	4 Hansons Terrace	F1X1Tmhm4B		Ginnie Skippon	1234567800003320	21/08
26	10025	Pearla	Normabell	pnormabello@examin	Female	(361) 2174476	95671 Dottie Junction	6WaxUfgbpr7e		Pearla Normabell	7777888899996666	22/01
27	10026	Darby	Alf	dalfp@altervista.org	Male	(576) 3342994	546 Bowman Way	08jydb		Darby Alf	1234567800003320	22/04
28	10027	Buddy	Sherwill	bsherwillq@chron.con	Female	(124) 6624103	46512 Westport Parkwa	3MESdbS1wd		Buddy Sherwill	1234567890123450	22/09
29	10028	Breana	McAuslene	bmcauslener@cdc.gov	Female	(845) 1439402	2 Johnson Parkway	4PhmmXC7XU8m		Breana McAuslene	1234123412341234	26/04
30	10029	Port	Goneau	pgoneaus@salon.com	Female	(529) 3026673	7 Meadow Valley Street	6FeNoOc3		Port Goneau	1234567800003320	22/02
31	10030	Rosita	Hucknall	rhucknallt@mlb.com	Male	(379) 5060377	5 Pepper Wood Pass	kf9DU0		Rosita Hucknall	7777888899996666	21/06

The worksheet containing existing customer information will be updated. If the customers press the "Enter" button for the second input, the data will be automatically modified in the next row. The Welcome panel will then appear, allowing them to log in.

Userform 4: Make_Orders_Step4

This userform has two pages, one is for selecting the products and the other one is for confirming the orders that have been selected. One the first page, customers can select the products and indicate the quantity they want and then add them to the cart. As we can see, Apple offers 4 products to choose from. For the products that have been added to the shopping cart, customers can click the remove button to remove those items they do not want. And the total price of the order will be shown on the right corner.



Product Information



Iphone \$6000



Ipad \$4500



MacBook \$ 8400



Iwatch \$3000

Order Information

Select a product and indicate the quantity. Click
Add To Chart button to process.

Product	Price(\$)
iWatch	3000
iPad	4500
iPhone	6000
Macbook	8400

Quantity



1

Add To Cart

Shopping Cart

Select one or more items. Then click Remove button to
remove the selected items from the shopping cart.

Product	Qty	Price (\$)
iWatch	5	15000
iPad	3	13500
iPhone	2	12000
Macbook	1	8400

Total: \$48,900.0

Remove

Proceed

Quit

Option Explicit

Dim dblTotal As Double

Private Sub CommandButton1_Click()

Gift_Step4continue.Show

End Sub

Private Sub UserForm_Initialize()

MultiPage1.Value = 0

With lbxProducts

.AddItem "Product"

.List(0, 1) = "Price(\$)"

.AddItem "iWatch"

.List(1, 1) = "3000"

.AddItem "iPad"

.List(2, 1) = "4500"

.AddItem "iPhone"

.List(3, 1) = "6000"

.AddItem "Macbook"

.List(4, 1) = "8400"

End With

With lbxCart

.AddItem "Product"

.List(0, 1) = "Qty"

.List(0, 2) = "Price (\$)"

End With

End Sub

Private Sub lbxProducts_Click()

If lbxProducts.Selected(0) Then

fmeQuantity.Enabled = False

Else

fmeQuantity.Enabled = True

End If

End Sub

Private Sub spnQuantity_Change()

lblQuantity.Caption = spnQuantity.Value

End Sub

Private Sub lbxCart_Change()

Dim intIndex As Integer, intSelect As Integer

For intIndex = 1 To lbxCart.ListCount - 1

If lbxCart.Selected(intIndex) = True Then

btnRemove.Enabled = True

intSelect = intSelect + 1

End If

Next intIndex

If intSelect = 0 Then btnRemove.Enabled = False

End Sub

Private Sub btnRemove_Click()

Dim intIndex As Integer

With lbxCart

For intIndex = lbxCart.ListCount - 1 To 1 Step -1

If .Selected(intIndex) = True Then

dblTotal = dblTotal - VBA.Val(.List(intIndex, 2))

lblTotal.Caption = VBA.Format(dblTotal, "\$#,##0.0")

.RemoveItem (intIndex)

End If

Next intIndex

End With

btnRemove.Enabled = False

If lbxCart.ListCount < 2 Then btnProceed.Enabled = False

End Sub

```

Private Sub btnCart_Click()
    Dim intMatch As Integer, intIndex As Integer
    intMatch = 0
    For intIndex = 1 To lbxCart.ListCount - 1
        If lbxProducts.List(lbxProducts.ListIndex) = lbxCart.List(intIndex) Then
            intMatch = 1
            lbxCart.List(intIndex, 1) = VBA.Val(lbxCart.List(intIndex, 1)) + VBA.Val(lblQuantity.Caption)
            lbxCart.List(intIndex, 2) = VBA.Val(lbxCart.List(intIndex, 2)) + VBA.Val(lbxProducts.List(lbxProducts.ListIndex, 1)) * spnQuantity.Value
            Exit For
        End If
    Next intIndex
    If intMatch = 0 Then
        With lbxCart
            .AddItem lbxProducts.List(lbxProducts.ListIndex, 0)
            .List(intIndex, 1) = lblQuantity.Caption
            .List(intIndex, 2) = VBA.Val(lbxProducts.List(lbxProducts.ListIndex, 1)) * spnQuantity.Value
        End With
    End If
    dblTotal = dblTotal + VBA.Val(lbxProducts.List(lbxProducts.ListIndex, 1)) * spnQuantity.Value
    lblTotal.Caption = VBA.Format(dblTotal, "$#,##0.0")
    lbxProducts.ListIndex = -1
    spnQuantity.Value = 1
    fmeQuantity.Enabled = False
    btnProceed.Enabled = True
End Sub

Private Sub lbxCart_Change()
    Dim intIndex As Integer, intSelect As Integer
    For intIndex = 1 To lbxCart.ListCount - 1
        If lbxCart.Selected(intIndex) = True Then
            btnRemove.Enabled = True
            intSelect = intSelect + 1
        End If
    Next intIndex
    If intSelect = 0 Then btnRemove.Enabled = False
End Sub

```

Once our customers choose to move to the next step, they can click the proceed button and the userform will turn to page2, which is hidden.

Order Products

✕

[Back](#)



[Click me and choose the gift before placing orders](#)

Customer Information

Customer ID: 10201 Phone Number: (852)1234567 Address: HK island
 First Name: Amy Email: amy@gmail.com
 Last Name: Chan Credit Card: 1234345454645650

Order Summary

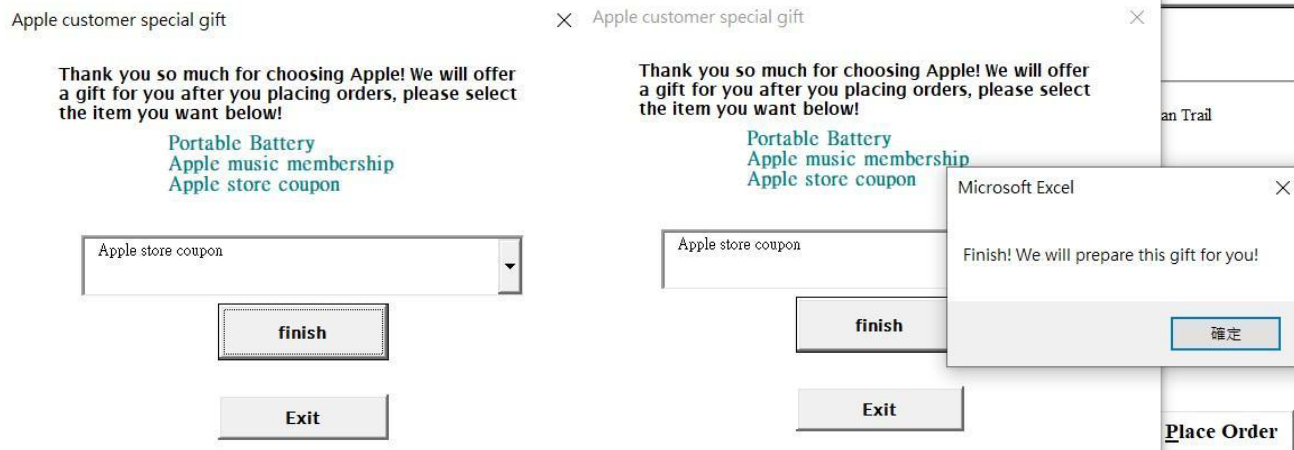
Code	Product	Qty	Price(\$)
10101	iWatch	5	15000
20202	iPhone	2	12000
40404	Macbook	1	8400
30303	iPad	3	13500

Order Total: \$48,900.0

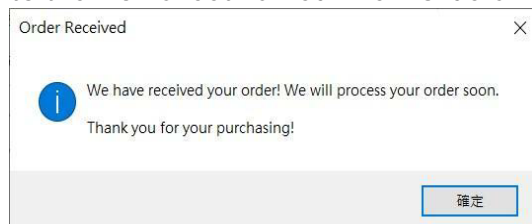
[Cancel Order](#)

[Place Order](#)

On this page, we can check the customer data again, and the data is obtained from the existing customer data worksheet by matching with the login information. The summary shows the order again. Before we really place the order, we can click the button on the right corner to choose a free gift from 3 options.



After checking if everything is correct, the customer may choose to place an order and the data will be passed to the worksheet named "New Orders".



And new orders will start with a specific order number starting with E. The order date will be the same date that order occurs.

	A	B	C	D	E	F	G	H	I	J	K	L
1	Order Number	Order Date	Name (First Name Last Name)	Phone Number	Address	10101	20202	30303	40404	Total Qty	Subtotal (\$)	
2	E1	28/4/2021	Amy Chan	(852)1234567	HK island	5	2	3	1	11	48,900.00	
3												
4												

```
Private Sub btnQuit_Click()
    VBA.Unload Make_order_Step4
End Sub

Private Sub btnOrder_Click()
    Dim wksheetOrder As Worksheet, intRow As Long, intNum As Integer, intCol As Long
    Dim ShippedDataCount As Long

    Set wksheetOrder = Sheets("New Orders")
    intRow = wksheetOrder.Cells(Rows.Count, 1).End(xlUp).Row
    ShippedDataCount = Sheets("Shipped").Cells(Rows.Count, 1).End(xlUp).Row - 1

    With wksheetOrder
        .Range("A1").Offset(intRow, 0).Value = "E" & intRow + ShippedDataCount
        .Range("A1").Offset(intRow, 1).Value = Date
        .Range("A1").Offset(intRow, 2).Value = lblFirstName & " " & lblLastName
        .Range("A1").Offset(intRow, 3).Value = lblPhoneNum
        .Range("A1").Offset(intRow, 4).Value = lblAddress
        For intNum = 1 To lblSummary.ListCount - 1
            For intCol = 6 To 9
                If lblSummary.List(intNum, 0) = CStr(.Cells(1, intCol).Value) Then
                    .Range("A1").Offset(intRow, intCol - 1) = lblSummary.List(intNum, 2)
                    Exit For
                End If
            Next intCol
        Next intNum
        .Range("A1").Offset(intRow, 9).Value = "=SUM(F" & intRow + 1 & ":I" & intRow + 1 & ")"
        .Range("A1").Offset(intRow, 9).Copy
        .Range("A1").Offset(intRow, 9).PasteSpecial Paste:=xlPasteValuesAndNumberFormats
        Application.CutCopyMode = False
        .Range("A1").Offset(intRow, 10).Value = VBA.Right(lblGrandTotal.Caption, VBA.Len(lblGrandTotal.Caption) - 1)
    End With

    MsgBox "We have received your order! We will process your order soon." & Chr(10) & _
        Chr(10) & "Thank you for your purchasing!", vbInformation, "Order Received"
End Sub

Private Sub btnCancel_Click()
    MsgBox "Thank you for your browsing.", vbInformation, "Leave Application"
End Sub
```

In case the customer wants to leave the system, we would like them to use the quit button to leave.


```

Private Sub btnBack_Click()
    MultiPage1.Value = 0
End Sub

Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)
    If CloseMode <> 1 Then
        If MultiPage1.Value = 1 Then
            Cancel = 1
            MsgBox "Please cancel order to leave.", vbExclamation, "Order Products"
        Else
            Cancel = 1
            MsgBox "Use the Quit button on the form to quit the application.", vbCritical, "Order Products"
        End If
    End If
End Sub

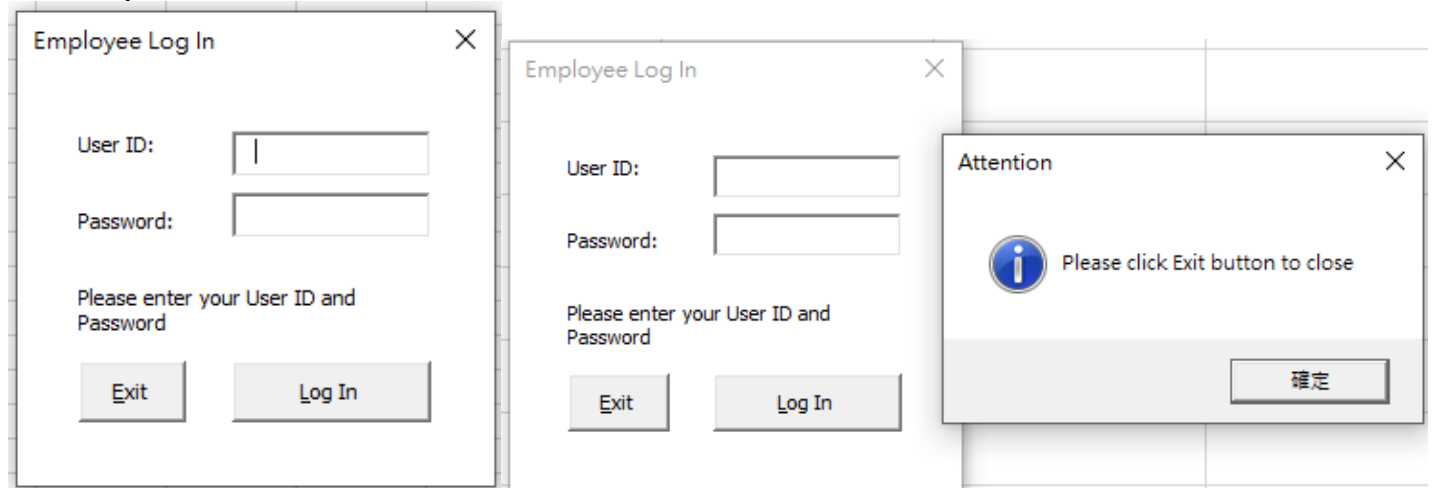
```

Userform 5: Employees_login_Step5

By clicking the button “btnStaff”, which is shown as “Staff” in the form Userform 1: Start_Step1, the employee users will come to Userform5: Employees_login_Step5.

All employees must click the Exit button when they quit Userform5: Employees_login_Step5.

If the users want to exit by clicking the exit at the upper right corner, a message box warning will be shown up with a caption “Please click the Exit button to close” as a VBA attention.



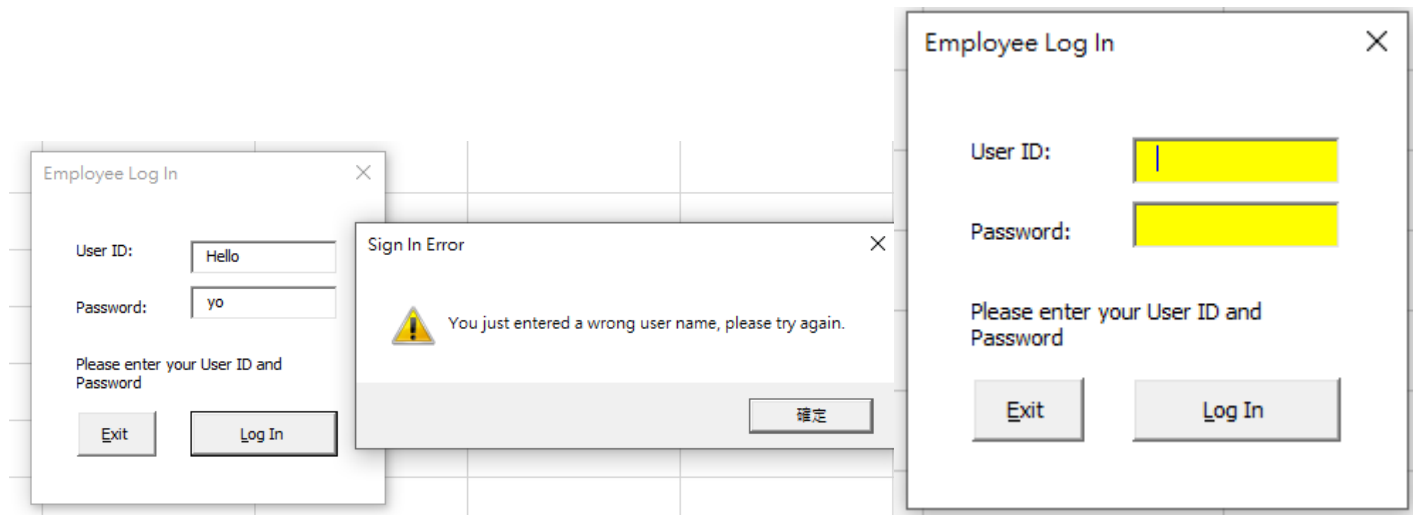
```

Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)
    If CloseMode = vbFormControlMenu Then
        MsgBox "Please click Exit button to close", vbInformation, "Attention"
        Cancel = True
    End If
End Sub

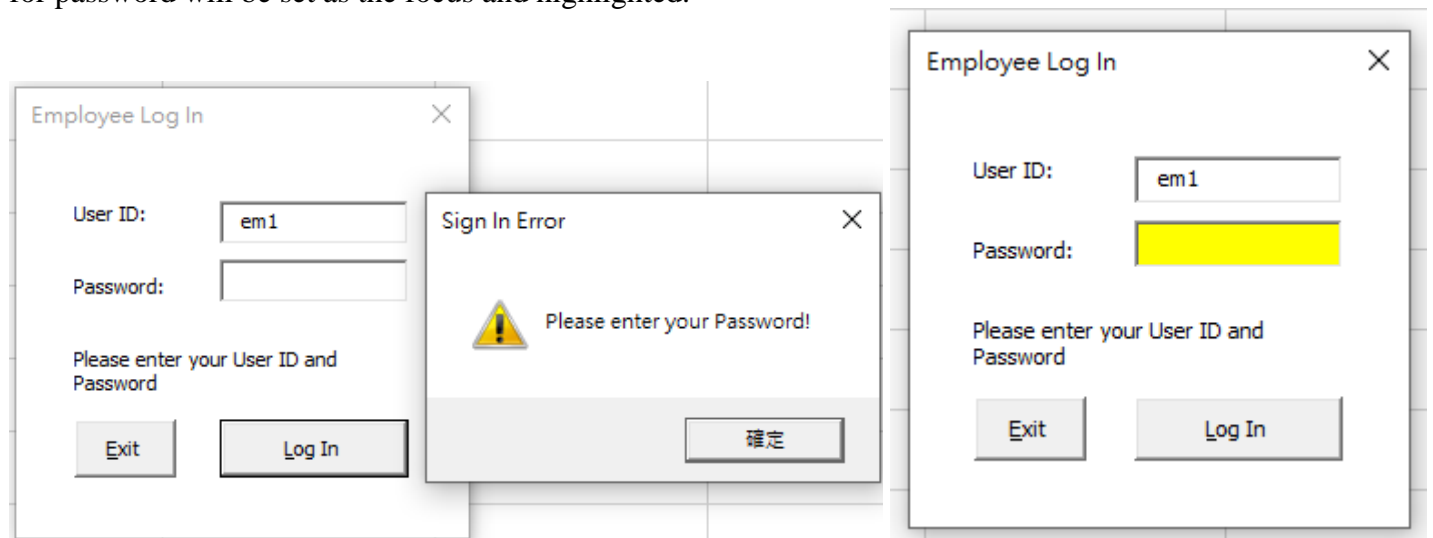
```

In Userform5: Employees_login_Step5, the employees need to enter an existing User ID in the textbox for UserID”, which is “em1”. Then it is needed to enter the correct password in the Textbox below, which is “apple”.

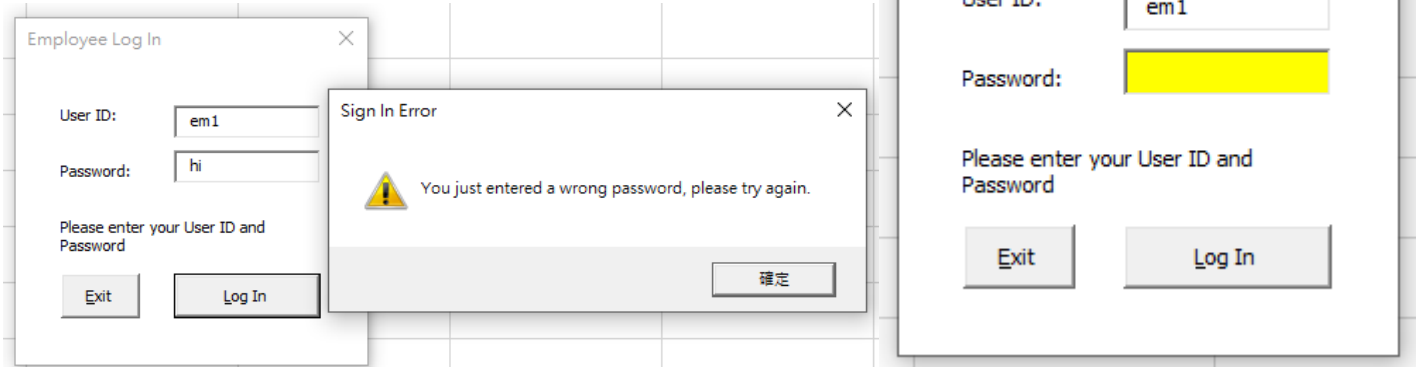
If the users enter the wrong User ID, there will be a message box shown up with a caption “You just entered a wrong user ID, please try again.” as a VBA sign in error. And both textboxes will be set as yellow.



Afterwards, there will be a warning if the users do not enter any password as a VBA sign in error. A message box will show up with a caption “Please enter your password” to tell users to enter something. And the TextBox for password will be set as the focus and highlighted.



On the other hand, if the users enter the wrong password, a message box will show up with a caption “You just entered a wrong user ID, please try again.” to tell users to enter the correct password as a VBA sign in error. And the textbox will also be set as the focus and the colour of its background will be changed.



After the User ID and the password are entered correctly, there will be a message box shown as “You have been log in!”. Then Userform 6: Choose_Your_Role_Step6 will be shown.

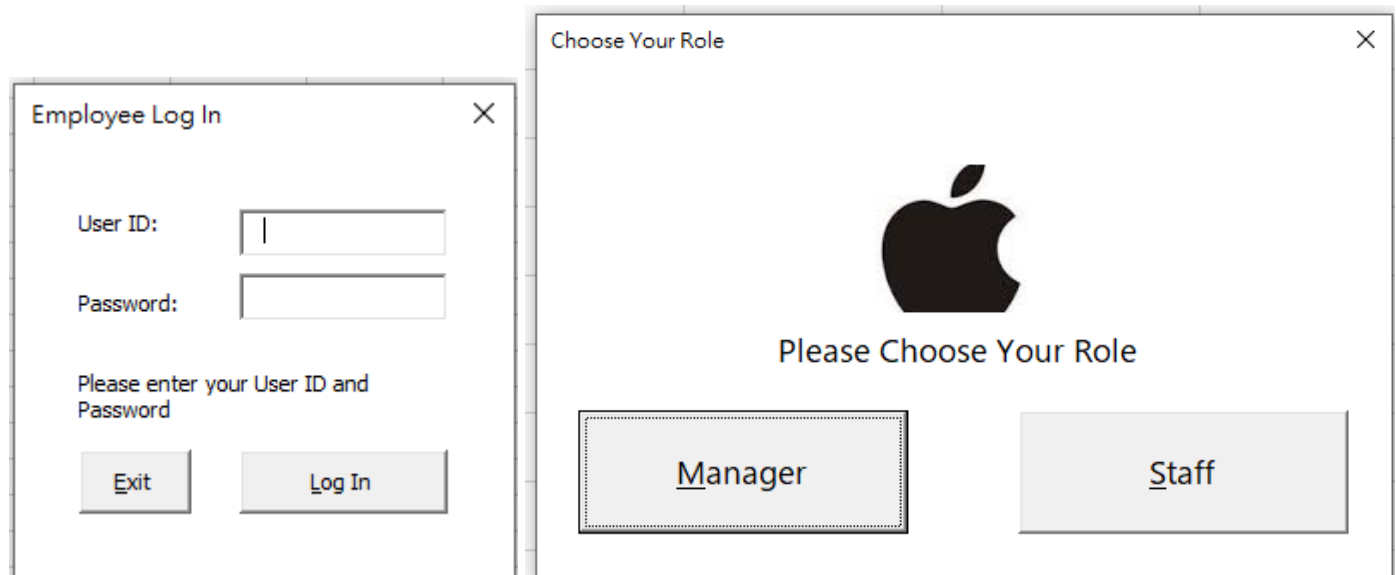
```
Private Sub btnEmployee_Login_Click()

    If txt_UID_Employee = "" Then
        MsgBox "Please enter your user ID and Password!", vbOKOnly + vbExclamation, "Sign In Error"
        txt_PW_Employee = ""
        txt_UID_Employee.SetFocus
        txt_UID_Employee.BackColor = rgbYellow
        txt_PW_Employee.BackColor = rgbYellow
    Exit Sub
    End If

    If txt_PW_Employee = "" Then
        MsgBox "Please enter your Password!", vbOKOnly + vbExclamation, "Sign In Error"
        txt_PW_Employee.SetFocus
        txt_PW_Employee.BackColor = rgbYellow
    Exit Sub
    End If

    Password_Employee = False
    If txt_UID_Employee = "em1" Then
        If txt_PW_Employee = "apple" Then
            Password_Employee = True
            Employee_Login_Step5.Hide
            MsgBox ("You have been successfully log in !"), , "Innisfree"
            Choose_Your_Role_Step6.Show
            txt_UID_Employee = ""
            txt_PW_Employee = ""
        Exit Sub
        Else
            Password_Employee = False
            MsgBox "You just entered a wrong password, please try again.", vbOKOnly + vbExclamation, "Sign In Error"
            txt_PW_Employee = ""
            txt_PW_Employee.SetFocus
            txt_PW_Employee.BackColor = rgbYellow
        Exit Sub
        End If
    Else
        Password_Employee = False
        MsgBox "You just entered a wrong user name, please try again.", vbOKOnly + vbExclamation, "Sign In Error"
        txt_UID_Employee = ""
        txt_PW_Employee = ""
        txt_UID_Employee.SetFocus
        txt_UID_Employee.BackColor = rgbYellow
        txt_PW_Employee.BackColor = rgbYellow
    Exit Sub
    End If
End Sub
```

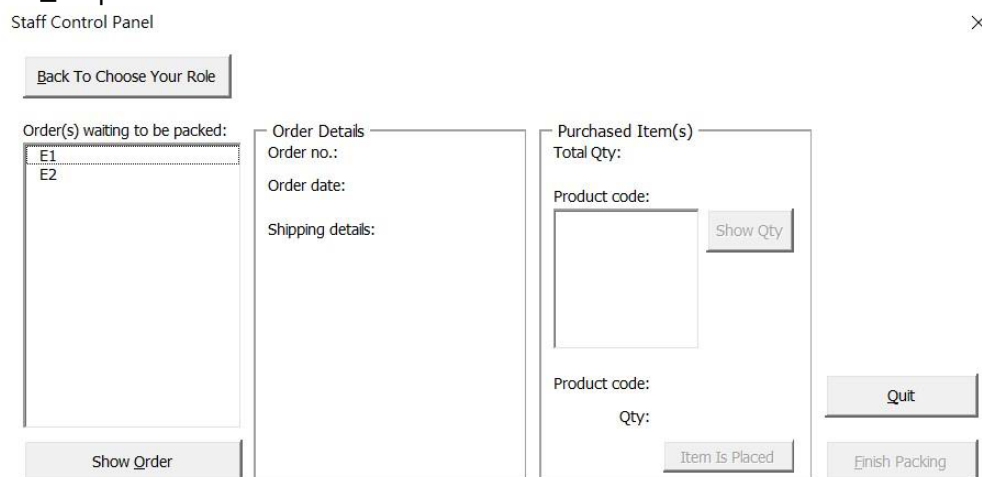
Userform 6: Choose_Your_Role_Step6



The employees can press either the button “Manager” or “Staff” in Userform 6: Choose_Your_Role_Step6 to choose the user role, which represents manager and staff respectively. They will use their control panel depending on their roles.

Userform 7: Staff_Control_Panel_Step7

If the user has chosen the role “Staff” in step 6, the program will lead the user to UserForm7: Staff_Control_Panel_Step7.



When the user clicks into Userform 7: Staff_Control_Panel_Step7 wrong accidentally, the users can go back to Userform 6: Choose_Your_Role_Step6 to choose the appropriate role by clicking the button on the left corner.

```
Private Sub btnBack_Click()  
    Unload Me  
    Choose_Your_Role_Step6.Show  
End Sub
```

A Quit button is also provided for the user to end the application.

```
Private Sub btnQuit_Click()  
    Unload Me  
End Sub
```

To prevent closing the application accidentally, closing the application by clicking the close button in the top right corner is prohibited. The user can only click the Quit button to close the application.

```

Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)
    If CloseMode = vbFormControlMenu Then
        MsgBox "Please click Quit button to close.", vbInformation, "Attention"
        Cancel = True
    End If
End Sub

```

The Staff Control Panel is designed for the staff to pack and ship orders. If a customer has placed a new order, the order information would be added to the "New Orders" Sheet of the workbook.

	A	B	C	D	E	F	G	H	I	J	K	L
1	Order Number	Order Date	Name (First Name Last Name)	Phone Number	Address	10101	20202	30303	40404	Total Qty	Subtotal (\$)	
2	E1	28/4/2021	Amy Chan	(852)1234567	HK island	5	2	3	1	11	48,900.00	
3												
4												

The order information will then be displayed in the Staff Control Panel for the staff to handle the order. The staff can only select one order at a time.

```

Private Sub UserForm_Initialize()
    Dim FinalRow As Long

    FinalRow = Sheets("New Orders").Cells(Rows.Count, 1).End(xlUp).Row

    lstOrderNum.Clear
    If FinalRow = 2 Then
        lstOrderNum.AddItem Sheets("New Orders").Range("A2").Value
    ElseIf FinalRow > 2 Then
        lstOrderNum.List = Sheets("New Orders").Range("A2", "A" & FinalRow).Value
    End If
    If FinalRow = 1 Then lstOrderNum.Clear
    lblOrderNum.Caption = ""
    lblOrderDate.Caption = ""
    lblName.Caption = ""
    lblPhoneNum.Caption = ""
    lblAddress.Caption = ""
    lblSumQty.Caption = ""
    lstProductCode.Clear
    lblProductCode.Caption = ""
    lblQty.Caption = ""

    If lstOrderNum.ListCount = 0 Then
        btnShowOrder.Enabled = False
    Else
        btnShowOrder.Enabled = True
    End If
    btnShowQty.Enabled = False
    btnItemPlaced.Enabled = False
    btnFinishPacking.Enabled = False
    btnQuit.Enabled = True
    btnBack.Enabled = True
End Sub

```

Staff Control Panel

×

Back To Choose Your Role

Order(s) waiting to be packed:

E1

E2

Show Order

Order Details

Order no.: E1

Order date: 28/4/2021

Shipping details:

Amy Chan

(852)1234567

HK island

Purchased Item(s)

Total Qty: 11

Product code:

10101

20202

30303

40404

Show Qty

Product code:

Qty:

Item Is Placed

Quit

Finish Packing

Once the order has been selected and the Show Order button has been clicked, the staff can then view the order details that contains all the information of the selected order in the Order Details frame. In the Purchased Item(s) frame, it will display the Total Qty and the all the product codes of the specific order. Same as the order number list box, only one product code could be selected each time. At the same time, button Show Qty will be enabled.

```
Private Sub btnShowOrder_Click()
    Dim i As Long
    Dim j As Integer
    Dim FinalRow As Long

    FinalRow = Sheets("New Orders").Cells(Rows.Count, 1).End(xlUp).Row
    If FinalRow = 2 Then

        'show that order's details
        lblOrderNum.Caption = Sheets("New Orders").Cells(FinalRow, 1).Value
        lblOrderDate.Caption = Sheets("New Orders").Cells(FinalRow, 2).Value
        lblName.Caption = Sheets("New Orders").Cells(FinalRow, 3).Value
        lblPhoneNum.Caption = Sheets("New Orders").Cells(FinalRow, 4).Value
        lblAddress.Caption = Sheets("New Orders").Cells(FinalRow, 5).Value
        lblSumQty.Caption = Sheets("New Orders").Cells(FinalRow, 10).Value
        lstProductCode.Clear
        For j = 6 To 9
            If Sheets("New Orders").Cells(FinalRow, j).Value > 0 Then
                lstProductCode.AddItem Sheets("New Orders").Cells(1, j).Value
            End If
        Next j
        lblProductCode.Caption = ""
        lblQty.Caption = ""
        btnItemPlaced.Enabled = False

    ElseIf FinalRow > 2 Then
        For i = 2 To FinalRow
            If Sheets("New Orders").Cells(i, 1).Value = lstOrderNum Then

                'show that order's details
                lblOrderNum.Caption = Sheets("New Orders").Cells(i, 1).Value
                lblOrderDate.Caption = Sheets("New Orders").Cells(i, 2).Value
                lblName.Caption = Sheets("New Orders").Cells(i, 3).Value
                lblPhoneNum.Caption = Sheets("New Orders").Cells(i, 4).Value
                lblAddress.Caption = Sheets("New Orders").Cells(i, 5).Value
                lblSumQty.Caption = Sheets("New Orders").Cells(i, 10).Value
                lstProductCode.Clear
                For j = 6 To 9
                    If Sheets("New Orders").Cells(i, j).Value > 0 Then
                        lstProductCode.AddItem Sheets("New Orders").Cells(1, j).Value
                    End If
                Next j
                lblProductCode.Caption = ""
                lblQty.Caption = ""
                btnItemPlaced.Enabled = False

            Exit For

        End If
    Next i

    End If

    If lstProductCode.ListCount > 0 Then btnShowQty.Enabled = True
End Sub
```

With the Show Qty button clicked, the label will display the product code again with the total quantity of the specific product from the "New Orders" worksheet. When the product code is being selected, the "Item Is Placed" button will be enabled for staff to place the order for the particular product.

Order(s) waiting to be packed:

E1
E2

Order Details
Order no.: E1
Order date: 28/4/2021
Shipping details:
Amy Chan
(852)1234567
HK island

Purchased Item(s)
Total Qty: 11
Product code:

10101
20202
30303
40404

Product code: 10101
Qty: 5

```

Private Sub btnShowQty_Click()
    Dim i As Long
    Dim j As Integer
    Dim FinalRow As Long
    Dim C As Long

    FinalRow = Sheets("New Orders").Cells(Rows.Count, 1).End(xlUp).Row
    For i = 2 To FinalRow
        If Sheets("New Orders").Cells(i, 1).Value = lblOrderNum.Caption Then

            'show that item's details
            For j = 6 To 9
                C = Sheets("New Orders").Cells(1, j).Value
                If C = lstProductCode.Value Then
                    lblProductCode.Caption = Sheets("New Orders").Cells(1, j).Value
                    lblQty.Caption = Sheets("New Orders").Cells(i, j).Value
                    Exit For
                End If
            Next j

            End If
        Next i

        If lstProductCode.Value <> Empty Then btnItemPlaced.Enabled = True
    End Sub

```

With the "Item Is Placed" button being clicked, the product code will be removed from the list box and the quantity of the product will be removed from the "New Orders" sheet. In this stage, we assume the product will never run out of stock, therefore the quantity of the product will be added into the "Packing" Sheet temporarily to undergo packing procedure. On the "packing" worksheet, the picture belows shows where the item is transferred one by one after being placed but not yet finish packing.

	A	B	C	D	E	F	G	H	I	J	K	L
1	Order Number	Order Date	Name (First Name Last Name)	Phone Number	Address	10101	20202	30303	40404	Total Qty	Subtotal (\$)	
2	E1	44314	Amy Chan	(852)1234567	HK island		2			2	12000	
3												

Once the button "finish packing" is clicked, the quantity of the product with its order number, customer name, and order date will also be updated in the "Manager" sheet, where the transaction date, revenue, and profit will be calculated automatically for the manager to generate reports regarding the sales performance.

On the right hand side, the total quantity of the products in this worksheet will be calculated.

K3

✕

✓

*f*_x

=SUMIF(D3:D100,"iWatch", C3:D100)

	A	B	C	D	E	F	G	H	I	J	K
1	Manager										
2	Order Number	Name	Qty	Product	Costs	Transaction date	Revenue	Profit		product	total quantity
3	E1	Amy Chan	5	iWatch	4500	28/4/2021	15000	10500		iWatch	6
4	E1	Amy Chan	2	iPhone	3600	28/4/2021	12000	8400		iPhone	5
5	E1	Amy Chan	3	iPad	3600	28/4/2021	13500	9900		iPad	5
6	E1	Amy Chan	1	Macbook	3300	28/4/2021	8400	5100		Macbook	6
7	E2	Coco Wong	1	iWatch	900	28/4/2021	3000	2100			
8	E2	Coco Wong	3	iPhone	5400	28/4/2021	18000	12600			
9	E2	Coco Wong	2	iPad	2400	28/4/2021	9000	6600			
10	E2	Coco Wong	5	Macbook	16500	28/4/2021	42000	25500			

```
Private Sub btnItemPlaced_Click()
    Dim i As Long
    Dim j As Integer
    Dim FinalRow As Long
    Dim NewRow As Long
    Dim nr As Long
    Dim C As Long

    lstProductCode.Clear
    FinalRow = Sheets("New Orders").Cells(Rows.Count, 1).End(xlUp).Row
    For i = 2 To FinalRow
        If Sheets("New Orders").Cells(i, 1).Value = lblOrderNum.Caption Then

            'use Sheets("Packing") to store that item qty (col6-9)
            For j = 6 To 9
                C = Sheets("New Orders").Cells(1, j).Value
                If C = lblProductCode.Caption Then
                    NewRow = Sheets("Packing").Cells(Rows.Count, 1).End(xlUp).Row + 1
                    Sheets("New Orders").Cells(i, j).Cut Sheets("Packing").Cells(NewRow, j)
                    'tran that item qty to Sheets("Manager")
                    nr = Sheets("Manager").Cells(Rows.Count, 1).End(xlUp).Row + 1
                    With Sheets("Manager")
                        .Cells(nr, 1).Value = lblOrderNum.Caption
                        .Cells(nr, 2).Value = lblName.Caption
                        .Cells(nr, 3).Value = lblQty.Caption
                        If j = 6 Then
                            .Cells(nr, 4).Value = "iWatch"
                        ElseIf j = 7 Then
                            .Cells(nr, 4).Value = "iPhone"
                        ElseIf j = 8 Then
                            .Cells(nr, 4).Value = "iPad"
                        ElseIf j = 9 Then
                            .Cells(nr, 4).Value = "Macbook"
                        End If
                        .Cells(nr, 5).Value = "=VLOOKUP(D" & nr & ",Products!$B$1:$E$5,3,FALSE)*C" & nr
                        .Cells(nr, 6).Value = lblOrderDate.Caption
                        .Cells(nr, 7).Value = "=VLOOKUP(D" & nr & ",Products!$B$1:$E$5,2,FALSE)*C" & nr
                        .Cells(nr, 8).Value = "=G" & nr & "-E" & nr
                    End With
                    ElseIf Sheets("New Orders").Cells(i, j).Value > 0 Then
                        lstProductCode.AddItem Sheets("New Orders").Cells(1, j).Value
                    End If
                Next j
                lblProductCode.Caption = "-"
                lblQty.Caption = "-"
            End If
        Next i

        btnShowOrder.Enabled = False
        btnQuit.Enabled = False
        btnBack.Enabled = False
        btnFinishPacking.Enabled = True
        If lstProductCode.ListCount = 0 Then
            btnItemPlaced.Enabled = False
            btnShowQty.Enabled = False
        ElseIf lblQty.Caption = "-" Then
            btnItemPlaced.Enabled = False
        End If
    End If
End Sub
```

During the packing stage, if the staff has not packed all the product code of the order, the system will prompt an attention message in order to remind the staff to pack all items. The list box with unpacked products will then be highlighted in yellow to make sure they have all the products packed. When all products are well packed, the background color of the list box will eventually turn back to white.

[Back To Choose Your Role](#)

Order(s) waiting to be packed:

E1

Order Details

Order no.: E1

Order date: 28/4/2021

Shipping details:

Amy Chan

Purchased Item(s)

Total Qty: 11

Product code:

30303
40404

Show Qty

Attention



You haven't finished the packing. Please ensure you have pack all the item(s).

✕

Show Order

確定

Quit

Finish Packing

Staff Control Panel

✕

[Back To Choose Your Role](#)

Order(s) waiting to be packed:

E1

Order Details

Order no.: E1

Order date: 28/4/2021

Shipping details:

Amy Chan

(852)1234567

HK island

Purchased Item(s)

Total Qty: 11

Product code:

30303
40404

Show Qty

Product code: -

Qty: -

Item Is Placed

Quit

Finish Packing

Show Order

```
Private Sub 1stProductCode_AfterUpdate()
    If 1stProductCode.Value <> "" Then
        1stProductCode.BackColor = rgbWhite
    End If
End Sub
```

After packing all the products, the staff can then click the "Finish Packing" button. All the related data will then be moved to the "Shipped" worksheet and the expected delivery date has been set as 3 days from the packing date.

	A	B	C	D	E	F	G	H	I	J	K	L
1	Order Number	Order Date	Name (First Name Last Name)	Phone Number	Address	10101	20202	30303	40404	Total Qty	Subtotal (\$)	Expected Delivery Date
2	E1	2021/4/28	Amy Chan	(852)1234567	HK island	5	2	3	1	11	48,900.00	2021/5/1
3	E2	2021/4/28	Coco Wong	(852)9876567	Kowloon	1	3	2	5	11	72,000.00	2021/5/1
4	E3	2021/4/28	Steve Jobs	(852) 12123434	California, US	0	1	0	0	1	6,000.00	2021/5/1

```

Private Sub btnFinishPacking_Click()
    Dim i As Long
    Dim j As Integer
    Dim FinalRow As Long
    Dim NewRow As Long
    Dim NewRowRow As Long
    Dim C As Long

    If lstProductCode.ListCount <> 0 Then
        MsgBox "You haven't finished the packing. Please ensure you have pack all the item(s).", vbCritical, "Attention"
        lstProductCode.BackColor = rgbYellow
        Exit Sub
    End If

    'transfer that order to Sheets("Shipped")
    FinalRow = Sheets("New Orders").Cells(Rows.Count, 1).End(xlUp).Row
    NewRow = Sheets("Packing").Cells(Rows.Count, 1).End(xlUp).Row + 1
    NewRowRow = Sheets("Shipped").Cells(Rows.Count, 1).End(xlUp).Row + 1
    For i = 2 To FinalRow
        If Sheets("New Orders").Cells(i, 1).Value = lblOrderNum.Caption Then

            Sheets("New Orders").Cells(i, 1).Resize(1, 5).Cut Sheets("Packing").Cells(NewRow, 1)
            Sheets("New Orders").Cells(i, 10).Resize(1, 2).Cut Sheets("Packing").Cells(NewRow, 10)
            Sheets("New Orders").Rows(i).EntireRow.Delete
            Sheets("Packing").Cells(NewRow, 1).Resize(1, 11).Cut Sheets("Shipped").Cells(NewRowRow, 1)
            Sheets("Shipped").Cells(NewRowRow, 12).Value = Date + 3
            For j = 6 To 9
                If Sheets("Shipped").Cells(NewRowRow, j) = Empty Then
                    Sheets("Shipped").Cells(NewRowRow, j).Value = "0"
                End If
            Next j

        End If
    Next i

    Call UserForm_Initialize
    lstProductCode.BackColor = rgbWhite
End Sub

```

With all the procedure done, the staff can continue packing other orders, choose the role, or simply quit the program. When there is no order that needs to be packed, the “Show Order” button will be prohibited to click.

Userform 8: Staff_Control_Panel_Step8

If the user chooses “Manager” in “Choose_Your_Role_Step6”, the system will locate the user to Userform8: Staff_Control_Panel_Step8, then it will provide a multipage with two pages to the user.

The user clicks into Userform8: Staff_Control_Panel_Step8 wrong accidently, the users can go back to Userform 6: Choose_Your_Role_Step6 to choose the appropriate role by clicking the button “bthBack”, shown as “Back To Choose Your Role”.

```

Private Sub btnBack_Click()
    Unload Me
    Choose_Your_Role_Step6.Show
End Sub

```

The users can choose to view page 1, the manager can click on the three option buttons, which includes “Revenue”, “Performance of Each Product”, and “Profit”, to select which chart the system should generate. Then, the user can click on the “Generate” button to create a chart.

X

Back To Choose Your Role

Page1 Page2

A. Choose which report you want to generate

☐ Revenue
☐ Performance of Each Product
☐ Profit

Generate

Exit

```

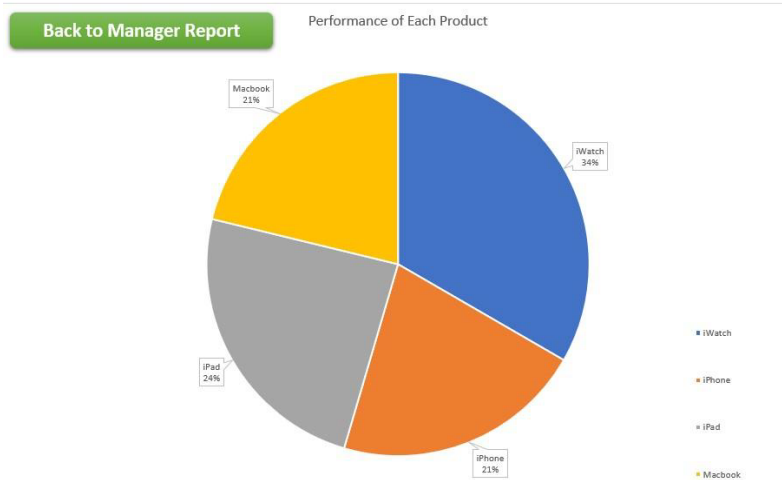
Private Sub cmb_generate_Click()
    Dim FinalRow As Long

    FinalRow = Range(Sheets("Manager").Range("A1"), Sheets("Manager").Range("A1").End(xlDown)).Rows.Count

    If opt1.Value = True Then
        Charts("Revenue").Visible = True
        Charts("Revenue").Activate
        ActiveChart.ChartArea.Select
        With Sheets("Manager")
            ActiveChart.SetSourceData Source:=.Range("G3", "G" & FinalRow), PlotBy:=xlColumns
            ActiveChart.SeriesCollection(1).Name = "=Manager!R2C7"
            ActiveChart.SeriesCollection(1).XValues = "=Manager!R3C6:R" & FinalRow & "C6"
        End With
        Sheets("Start").Visible = xlSheetVeryHidden
        Staff_Control_Panel_Step8.Hide
        btnBack.Enabled = False
    ElseIf opt2.Value = True Then
        Charts("Profit").Visible = True
        Charts("Profit").Activate
        ActiveChart.ChartArea.Select
        With Sheets("Manager")
            ActiveChart.SetSourceData Source:=.Range("H3", "H" & FinalRow), PlotBy:=xlColumns
            ActiveChart.SeriesCollection(1).Name = "=Manager!R2C8"
            ActiveChart.SeriesCollection(1).XValues = "=Manager!R3C6:R" & FinalRow & "C6"
        End With
        Sheets("Start").Visible = xlSheetVeryHidden
        Staff_Control_Panel_Step8.Hide
        btnBack.Enabled = False
    ElseIf opt3.Value = True Then
        Charts("Performance of Each Product").Visible = True
        Charts("Performance of Each Product").Activate
        Sheets("Start").Visible = xlSheetVeryHidden
        Staff_Control_Panel_Step8.Hide
        btnBack.Enabled = False
    End If
End Sub

```

For example, we can see the distribution of the products on the diagram below.



On the other hand, the users can choose to use page 2 of the multipage. At page2, the users can click the button "Browse" to select an excel file with a format as ".xlsx" and then open the selected file. As a reminder, the available year, 2018, 2019 and 2020, will be shown above the button "Browse".

Manager Report

Back To Choose Your Role

Page1 Page2

B. choose the past sales data you want to check :
available year :2018, 2019, 2020

Browse

ew folder

Name	Date modified	Type	Size
今天 (3)			
2019	28/4/2021 3:11	Microsoft Excel ...	25 KB
2018	28/4/2021 3:11	Microsoft Excel ...	26 KB
2020	28/4/2021 2:46	Microsoft Excel ...	25 KB

File name: Excel

Tools Open Cancel

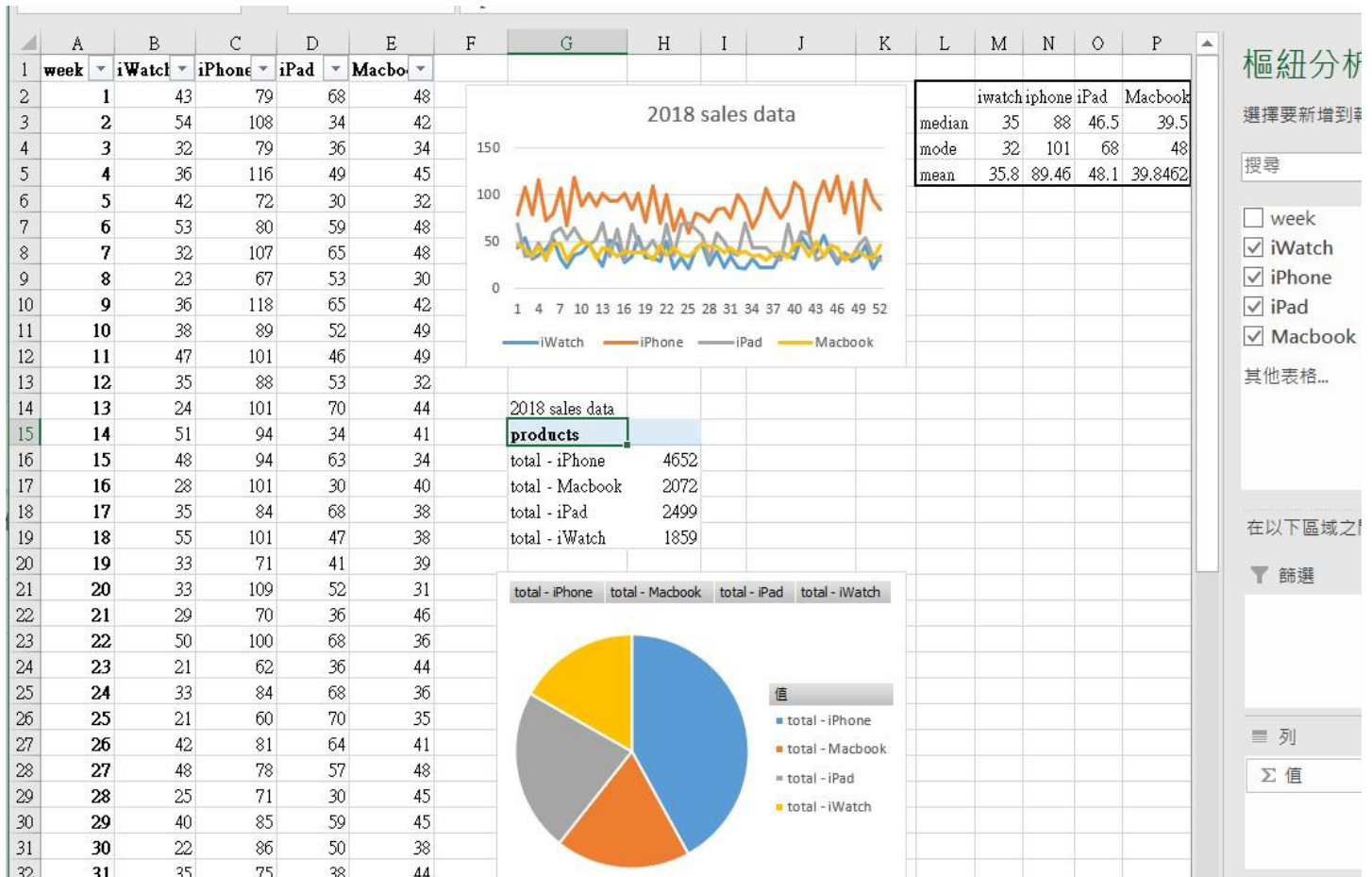
```
Private Sub Browse_Click()
```

```
Dim stringFileName As String
```

```
stringFileName = Application.GetOpenFilename(filefilter:="Excel,*.xlsx")
```

```
Workbooks.Open FileName:=stringFileName
```

```
End Sub
```



After browsing, the new workbook will be opened. In the historical data, we also include different diagrams and inserted pivot tables. And we also calculated the mean, mode, median of the product sales quantity in each week of that year.

Finally, the user must click on the "Exit" button to end the stage.

```
Private Sub cmb_Exit_Click()
```

```
Sheets("Start").Visible = True
```

```
Charts("Revenue").Visible = xlSheetVeryHidden
```

```
Charts("Profit").Visible = xlSheetVeryHidden
```

```
Charts("Performance of Each Product").Visible = xlSheetVeryHidden
```

```
Unload Me
```

```
End Sub
```

It is notable that all charts will be hidden until the user clicks on the "Generate" button. Each report will be created on a new worksheet.

This is the end of the report, thank you so much.

Appendix:

Appendix 1.0 Start_Step1

Controls	Caption	Name of Control
UserForm	Apple	Start_Step1
Image	/	Image1
Label	Please choose what type of users you are	Label1
Command Button	Customers	btnCustomers
Command Button	Employee	btnEmployees

Appendix 2.0 Welcome_Step2


Controls	Caption	Name of Control
UserForm	Welcome	Welcome_Step2
Image	/	Image1
Label	Existing Customer	Label2
Label	New Customer	Label3
Label	Email Address:	Label5
Label	Password:	Label6
TextBox	/	txt_UID_EC
TextBox	/	Txt_PW_EC
Command Button	Log in	btnLogIn_EC
Command Button	Create an Account	btnCreateAccount
Command Button	Exit	EC_NC_quit

Controls	Caption	Name of Control
UserForm	New Customer Registration	New_Customers_Step3
Label	First Name	Label1
TextBox	/	txtFirstName
Label	Last Name	Label2
TextBox	/	txtLastName
Label	Password	Label3
TextBox	/	txtPassword
Label	*mix with letters & numbers	Label15
Label	Email	Label7
TextBox	/	txtEmail
ComboBox	/	cboMail
Label	Gender	Label6
ComboBox	/	cboGender
Label	Phone number	Label8
ComboBox	/	cboArea
TextBox	/	txtPhoneNumber
Label	Address	Label9
TextBox	/	txtAddress

Label	Credit Card Information	Label11
Label	Cardholder Name	Label12
TextBox	/	txt_Cd_name
Label	Credit Card Number	Label13
TextBox	/	txt_Cd_num
Label	Expiry Date (YY/MM)	Label14
TextBox	/	txt_Cd_year
TextBox	/	txt_Cd_month
CommandButton	Clear	cmbClear
CommandButton	Enter	cmbEnter
CommandButton	Back To Login	cmbBackToLogin
CommandButton	Exit	cmbExit


Appendix 4.0 Make_order_Step4

Controls	Caption	Name of Control
UserForm	Order Products	Make_Order_Step4
Image	/	Image12
Frame	Product Information	Frame5
Image	/	Image13
Image	/	Image14

Image	/	Image17
Image	/	Image16
Label	Iphone \$6000 Ipad \$4500 MacBook \$8400 Iwatch \$3000	Label28
Frame	Order Information	Frame2
Label	Select a product and indicate the quantity. Click Add To Cart button to process	Label9 
ListBox	/	lbxProducts
Frame	Quantity	fmeQuantity
SpinButton	/	spnQuantity
Label	1	lblQuantity
Command Button	Add to Cart	btnCart
Frame	Shopping Cart	Frame3
Label	Select one or more items. Then click the Remove button to remove the selected items from the shopping cart.	Label10
ListBox	/	lbxCart
Command Button	Remove	btnRemove
Label	Total:	Label11
Label	\$0.0	lblTotal

Command Button	Proceed	btnProceed
Command Button	Quit	btnQuit

Appendix 4.1 Gift_Step4continue

UserForm	Apple customer special gift 	Gift_Step4continue
Label	Thank you so much for choosing Apple! We will offer a gift for you after you <u>placing</u> orders, please select the item you want below!	Label1
Label	Portable Battery Apple music membership Apple store coupon	Label2
ComboBox	/	ComboBox1
Command Button	finish	CommandButton1
Command Button	Exit	CommandButton2

Appendix 5.0 Employee_Login_Step5

Controls	Caption	Name of Control
UserForm	Employee Log In	Emplyee_Login_Step5
Label	User ID:	Label5
Text Box	/	txt_UID_Employee
Label	Password:	Label6
Text Box	/	txt_PW_Employee
Command Button	Exit	btnELI_exit
Command Button	Log In	btnEmployee_login

Appendix 6.0 Choose_Your_Role_Step6

Controls	Caption	Name of Control
Userform	Choose Your Role	Chooes_Your_Role_Step6
Image	/	Image1
Label	Please Choose Your Role	Label1
Command Button	Manager	btnManager
Command Button	Staff	btnStaff

Appendix 7.0 Staff_Control_Panel_Step7

Controls	Caption	Name of Control
UserForm	Staff Control Panel	Staff_Control_Panel_Step7
CommandButton	Back To Choose Your Role	btnBack
Label	Order(s) waiting to be packed:	Label1
ListBox	/	lstOrderNum
Command Button	Show Order	btnShowOrder
Frame	Order Details	Frame1
Label	Order no.:	Label3
Label	N/A	lblOrderNum
Label	Order date:	Label4
Label	N/A	lblOrderDate
Label	Shipping details:	Label2
Label	N/A	lblName
Label	N/A	lblPhoneNum
Label	N/A	lblAddress
Frame	Purchased Item(s)	Frame2
Label	Total Qty:	Label5
Label	N/A	lblSumQty

Controls	Caption	Name of Control
Userform	Manager Report	Staff_Control_Panel_Step8
Command Button	Back To Choose Your Role	btnBack
MultiPage	Page1	Page1
Label	A. Choose which report you want to generate	Label1
Option Button	Revenue	opt1
Option Button	Performance of Each Product	opt3
Option Button	Profit	opt2
Command Button	Generate	cmb_generate
Command Button	Exit	cmb_Exit
MultiPage	Page2	Page2
Label	B. choose the past sales data you want to check:	Label2
Label	Available year: 2018, 2019, 2020	Label3
Command Button	Browse	Browse
Command Button	Exit	Exit2