INDIVIDUAL ASSIGNMENT

TECHNOLOGY PARK MALAYSIA

CT010-3-1-FSD

FUNDAMENTALS OF SOFTWARE DEVELOPMENT

APU1F2006CS(IS)

SIM YOKE SHIN TP059851

HAND OUT DATE: 17TH AUGUST 2020

HAND IN DATE: 20TH SEPTEMBER 2020

WEIGHTAGE: 100%

INSTRUCTIONS TO CANDIDATES:

1. Submit your assignment online in MS Teams unless advised otherwise
2. Late submission will be awarded zero (0) unless Extenuating Circumstances (EC) are upheld
3. Cases of plagiarism will be penalized
4. You must obtain at least 50% in each component to pass this module

Table of Contents

No.	Title	Page Number
1.	Introduction	3
2.	Assumption	4
3.	Design of the program • Pseudocode • flowcharts	5
4.	Program source and explanation	33
5.	Screenshots of sample input/output and explanation	45
6.	Conclusion	55
7.	Reference	56

Introduction

This report is a documentation for the planning of an Automobile Parts Inventory Management System for an automobile manufacturing plant of Hondi Motor Company Limited Partnership (Hondi Co.) in Johor. Due the economic slowdown, the company has assembled all assembly division of cars into three modules named Bios (BS), Ambry (AY), and Barrier (BR).There have provide a warehouse for each module likes WBS warehouse for Bios, WAY warehouse for Ambry, and WBR warehouse for Barrier. Besides that, each warehouse that use to store 3 types of assembly sections named ES, AS, and SC. Thus, the main purpose of this report is designing the new automobile parts inventory management system and explant about all parts of pseudocode, flowcharts, and code of it.

Module (Code)	Division and Sections	Warehouse Code
Bios(BS)	Engine Section(ES), Air-conditioning Section(AS), Safety and Cockpit section(SC)	WBS
Ambry(AY)	Engine Section(ES), Air-conditioning Section(AS), Safety and Cockpit section(SC)	WAY
Barrier(BR)	Engine Section(ES), Air-conditioning Section(AS), Safety and Cockpit section(SC)	WBR

Assumption

Hondi Co. has stored all the assembly parts in different warehouse according with its section. All the assembly parts are supplied by some famous assembly parts supplier's company likes Robert Bosch GmbH, Aisin Seiki Corporation, Denso Corporation, and ZF Friedrichshafen AG. Hondi Co. has gotten the parts of ES section from Aisin Seiki Corporation, get the parts of AS section from Denso Corporation. The parts of SC section are provided by Robert Bosh GmbH and ZF Friedrichshafen AG.

The three main function of the automobile parts inventory management system is update parts inventory function, tracking parts inventory function, and searching function. Inside the update inventory function, the users can update the inventory for different warehouse. The company has extra stored a function for input the details of Suppliers. The update inventory function is design to input the addition parts for different warehouse's sections. The second main function is design for tracking the parts inventory. The users can choose either they want to print the details of whole parts that is store in the company or print the part's details which quantity is almost out of stock or just print the current warehouse's parts that they want. The last main function of this system is search function. It has provided three different ways for user to search likes search the parts, search the supplier details, and search the parts that supplied by supplier.

Design of program

Parts Inventory Creation in Warehouses and Update

Pseudocode:

```
FUNCTION updateInventory()
        PRINT('WBS warehouse,WAY warehouse, WBR warehouse')
        PRINT('Which warehouse:[Enter code in Capital Letter]')
        READ warCode
        IF warCode is Integer THEN
                IF warCode equal to 'WBS' THEN
                        PRINT(1.Add part)
                        PRINT(2.Edit Stock of Parts)
                        PRINT(Which service you want:)
                        READ service
                        IF service is Integer THEN
                                IF service equal to 1 THEN
                                        content = CALL saveInventoryWBS()
                                ELIF service equal to 2 THEN
                                        content = CALL editQuantityWBS()
                                ENDIF
                        ELSE
                                PRINT ('Wrong Typing')
                        ENDIF
                ELIF warCode equal to 'WAY' THEN
                        PRINT(1.Add part)
                        PRINT(2.Edit Stock of Parts)
                        PRINT(Which service you want:)
                        READ service2
                        IF service2 is Integer THEN
                                IF service2 equal to 1 THEN
                                        content = CALL saveInventoryWAY()
                                ELIF service2 equal to 2 THEN
                                        content = CALL editQuantityWAY()
                                ENDIF
                        ELSE
                                PRINT ('Wrong Typing')
                        ENDIF
                ELIF warCode equal to 'WBR' THEN
                        PRINT(1.Add part)
                        PRINT(2.Edit Stock of Parts)
                        PRINT(Which service you want:)
                        READ service3
                        IF service3 is Integer THEN
                                IF service3 equal to 1 THEN
                                        content = CALL saveInventoryWBR()
                                ELIF service3 equal to 2 THEN
                                        content = CALL editQuantityWBR()
                                ENDIF
                        ELSE
                                PRINT ('Wrong Typing')
                        ENDIF
                ENDIF
        ELSE
                PRINT ('Wrong Typing')
        ENDIF
ENDFUNCTION
```

```
    FUNCTION saveSupplier()
            OPEN FILE 'supplierDetails.txt' as fileHandlerSup in APPEND MODE
            PRINT('-----Adding supplier details-----')
            supplierDetails = CALL supplier()
            updateSup = CALL update(supplierDetails , fileHandlerSup)
            CLOSE fileHandlerSup
    ENDFUNCTION


FUNCTION saveInventoryWBS()
        PRINT('-----Adding parts into WBS warehouse-----')
        PRINT('Add to which section? [1.Engine Section(ES), 2.Air-conditioning Section(AS), 3.Safety and Cockpit section(SC)]')
        READ sellectedSection
        IF sellectedSection is Integer
                IF sellectedSection equal to 1 THEN
                        OPEN FILE 'ES-WBS.txt' as fileHandlerWBS in APPEND mode
                ELIF sellectedSection equal to 2 THEN
                        OPEN FILE 'AS-WBS.txt' as fileHandlerWBS in APPEND mode
                ELIF sellectedSection equal to 3 THEN
                        OPEN FILE 'SC-WBS.txt' as fileHandlerWBS in APPEND mode
                ENDIF
                inventoryWBS = CALL inventory()
                updateWBS = CALL update(inventoryWBS, fileHandlerWBS)
                CLOSE fileHandlerWBS
        ELSE
                PRINT ('Wrong Typing')
        ENDIF
ENDFUNCTION
FUNCTION saveInventoryWAY()
        PRINT('-----Adding parts into WAY warehouse-----')
        PRINT('Add to which section? [1.Engine Section(ES), 2.Air-conditioning Section(AS), 3.Safety and Cockpit section(SC)]')
        READ sellectedSection
        IF sellectedSection is Integer THEN
                IF sellectedSection equal to 1 THEN
                        OPEN FILE 'ES-WAY.txt' as fileHandlerWAY in APPEND mode
                ELIF sellectedSection equal to 2 THEN
                        OPEN FILE 'AS-WAY.txt' as fileHandlerWAY in APPEND mode
                ELIF sellectedSection equal to 3 THEN
                        OPEN FILE 'SC-WAY.txt' as fileHandlerWAY in APPEND mode
                ENDIF
                wayInventory = CALL inventory()
                updateWAY = CALL update(wayInventory, fileHandlerWAY)
                CLOSE fileHandlerWAY
        ELSE
                PRINT ('Wrong Typing')
`       ENDIF
ENDFUNCTION
FUNCTION saveInventoryWBR()
        PRINT('-----Adding parts into WBR warehouse-----')
        PRINT('Add to which section? [1.Engine Section(ES), 2.Air-conditioning Section(AS), 3.Safety and Cockpit section(SC)]')
        READ sellectedSection
        IF sellectedSection is Integer THEN
                IF sellectedSection equal to 1 THEN
                        OPEN FILE 'ES-WBR.txt' as fileHandlerWBR in APPEND mode
                ELIF sellectedSection equal to 2 THEN
                        OPEN FILE 'AS-WBR.txt' as fileHandlerWBR in APPEND mode
                ELIF sellectedSection equal to 3 THEN
                        OPEN FILE 'SC-WBR.txt' as fileHandlerWBR in APPEND mode
                ENDIF
                wbrInventory = CALL inventory()
                updateWBR = CALL update(wbrInventory, fileHandlerWBR)
                CLOSE fileHandlerWBR
        ELSE
                PRINT ('Wrong Typing')
        ENDIF
ENDFUNCTION
```

```
FUNCTION editQuantityWBS()
        PRINT('Engine Section(ES), Air-conditioning Section(AS), Safety and Cockpit section(SC)')
        PRINT('which sections:[Enter Code]')
        READ secCode
        IF secCode is Integer THEN
                IF secCode equal to 'ES' THEN
                        OPEN FILE 'ES-WBS.txt' as fileHandlerWBS in READ mode
                        PRINT('Which part you need to edit?[Enter Part Id]')
                        READ pardIdForSearch
                        PRINT('What is the new quantity of it')
                        READ newQuantity
                        DECLARE newData as emptyARRAY
                        readLine to return a line of text from fileHandlerWBS INTO data
                        FOR line IN data
                                CONVERT line into newLine as ARREY using tab as a delimiter
                                IF newLine[1] equal to partIdForSearch THEN
                                        newLine[2] replace by newQuantity
                                ENDIF
                                APPEND newLine INTO newData
                                PRINT(newLine)
                        CLOSE fileHandlerWBS
                        OPEN FILE 'ES-WBS.txt' as fileHandlerWBS1 in WRITE mode
                        FOR data IN newData
                                FOR item IN data
                                        WRITE item INTO fileHandlerWBS1
                                        WRITE TAB INTO fileHandlerWBS1
                                WRITE new line INTO fileHandlerWBS1
                        CLOSE fileHandlerWBS1
                ELIF secCode equal to 'AS' THEN
                        OPEN FILE 'AS-WBS.txt' as fileHandlerWBS in READ mode
                        PRINT('Which part you need to edit?[Enter Part Id]')
                        READ pardIdForSearch
                        PRINT('What is the new quantity of it')
                        READ newQuantity
                        DECLARE newData as emptyARRAY
                        readLine to return a line of text from fileHandlerWBS INTO data
                        FOR line IN data
                                CONVERT line into newLine as ARREY using tab as a delimiter
                                IF newLine[1] equal to partIdForSearch THEN
                                        newLine[2] replace by newQuantity
                                ENDIF
                                APPEND newLine INTO newData
                                PRINT(newLine)
                        CLOSE fileHandlerWBS
                        OPEN FILE 'AS-WBS.txt' as fileHandlerWBS1 in WRITE mode
                        FOR data IN newData
                                FOR item IN data
                                        WRITE item INTO fileHandlerWBS1
                                        WRITE TAB INTO fileHandlerWBS1
                                WRITE new line INTO fileHandlerWBS1
                        CLOSE fileHandlerWBS1
                ELIF secCode equal to 'SC' THEN
                        OPEN FILE 'SC-WBS.txt' as fileHandlerWBS in READ mode
                        PRINT('Which part you need to edit?[Enter Part Id]')
                        READ pardIdForSearch
                        PRINT('What is the new quantity of it')
                        READ newQuantity
                        DECLARE newData as emptyARRAY
                        readLine to return a line of text from fileHandlerWBS INTO data
                        FOR line IN data
                                CONVERT line into newLine as ARREY using tab as a delimiter
                                IF newLine[1] equal to partIdForSearch THEN
                                        newLine[2] replace by newQuantity
                                ENDIF
                                APPEND newLine INTO newData
                                PRINT(newLine)
                        CLOSE fileHandlerWBS
                        OPEN FILE 'SC-WBS.txt' as fileHandlerWBS1 in WRITE mode
                        FOR data IN newData
                                FOR item IN data
                                        WRITE item INTO fileHandlerWBS1
                                        WRITE TAB INTO fileHandlerWBS1
                                WRITE new line INTO fileHandlerWBS1
                        CLOSE fileHandlerWBS1
        ELSE
                PRINT ('Wrong Typing')
        ENDIF
ENDFUNCTION
```

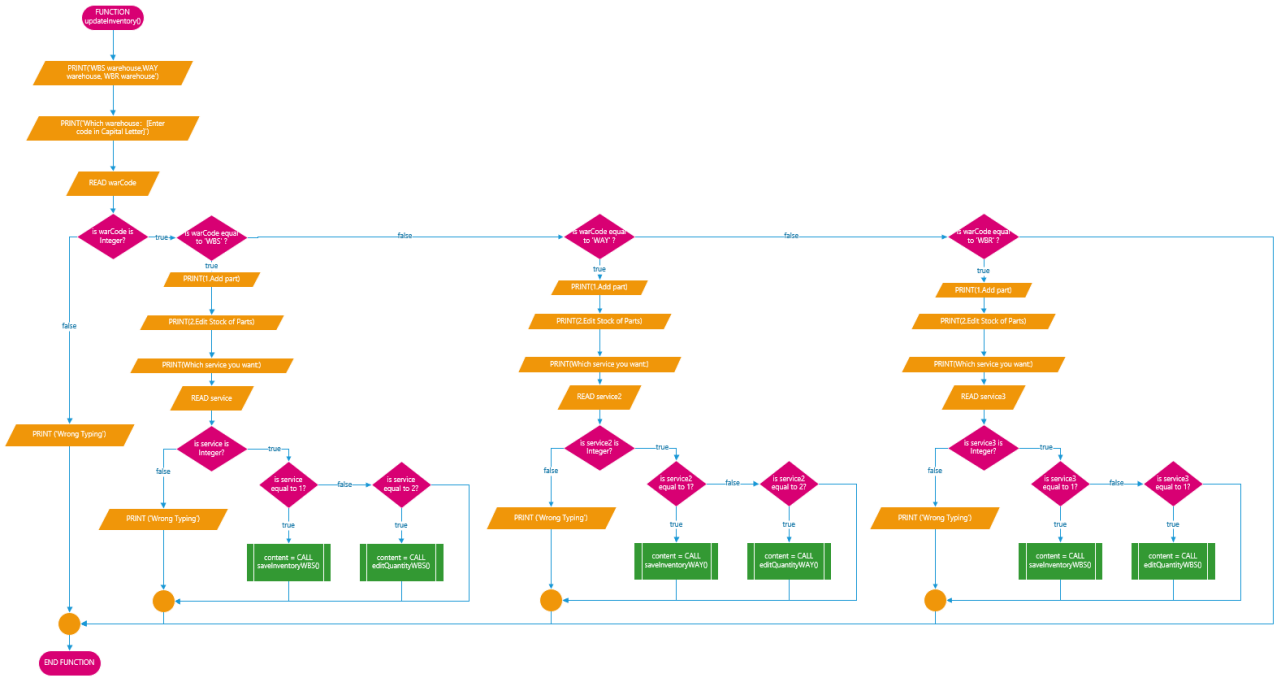
```
FUNCTION editQuantityWAY()
        PRINT('Engine Section(ES), Air-conditioning Section(AS), Safety and Cockpit section(SC)')
        PRINT('which sections:[Enter Code]')
        READ secCode
        IF secCode is Integer THEN
                IF secCode equal to 'ES' THEN
                        OPEN FILE 'ES-WAY.txt' as fileHandlerWAY in READ mode
                        PRINT('Which part you need to edit?[Enter Part Id]')
                        READ pardIdForSearch
                        PRINT('What is the new quantity of it')
                        READ newQuantity
                        DECLARE newData as emptyARRAY
                        readLine to return a line of text from fileHandlerWAY INTO data
                        FOR line IN data
                                CONVERT line into newLine as ARREY using tab as a delimiter
                                IF newLine[1] equal to partIdForSearch THEN
                                        newLine[2] replace by newQuantity
                                ENDIF
                                APPEND newLine INTO newData
                                PRINT(newLine)
                        CLOSE fileHandlerWAY
                        OPEN FILE 'ES-WAY.txt' as fileHandlerWAY1 in WRITE mode
                        FOR data IN newData
                                FOR item IN data
                                        WRITE item INTO fileHandlerWAY1
                                        WRITE TAB INTO fileHandlerWAY1
                                WRITE new line INTO fileHandlerWAY1
                        CLOSE fileHandlerWAY1
                ELIF secCode equal to 'AS' THEN
                        OPEN FILE 'AS-WAY.txt' as fileHandlerWAY in READ mode
                        PRINT('Which part you need to edit?[Enter Part Id]')
                        READ pardIdForSearch
                        PRINT('What is the new quantity of it')
                        READ newQuantity
                        DECLARE newData as emptyARRAY
                        readLine to return a line of text from fileHandlerWAY INTO data
                        FOR line IN data
                                CONVERT line into newLine as ARREY using tab as a delimiter
                                IF newLine[1] equal to partIdForSearch THEN
                                        newLine[2] replace by newQuantity
                                ENDIF
                                APPEND newLine INTO newData
                                PRINT(newLine)
                        CLOSE fileHandlerWAY
                        OPEN FILE 'AS-WAY.txt' as fileHandlerWAY1 in WRITE mode
                        FOR data IN newData
                                FOR item IN data
                                        WRITE item INTO fileHandlerWAY1
                                        WRITE TAB INTO fileHandlerWAY1
                                WRITE new line INTO fileHandlerWAY1
                        CLOSE fileHandlerWAY1
                ELIF secCode equal to 'SC' THEN
                        OPEN FILE 'SC-WAY.txt' as fileHandlerWAY in READ mode
                        PRINT('Which part you need to edit?[Enter Part Id]')
                        READ pardIdForSearch
                        PRINT('What is the new quantity of it')
                        READ newQuantity
                        DECLARE newData as emptyARRAY
                        readLine to return a line of text from fileHandlerWAY INTO data
                        FOR line IN data
                                CONVERT line into newLine as ARREY using tab as a delimiter
                                IF newLine[1] equal to partIdForSearch THEN
                                        newLine[2] replace by newQuantity
                                ENDIF
                                APPEND newLine INTO newData
                                PRINT(newLine)
                        CLOSE fileHandlerWAY
                        OPEN FILE 'SC-WAY.txt' as fileHandlerWAY1 in WRITE mode
                        FOR data IN newData
                                FOR item IN data
                                        WRITE item INTO fileHandlerWAY1
                                        WRITE TAB INTO fileHandlerWAY1
                                WRITE new line INTO fileHandlerWAY1
                        CLOSE fileHandlerWAY1
        ELSE
                PRINT ('Wrong Typing')
        ENDIF
ENDFUNCTION
```

```
FUNCTION editQuantityWBR()
        PRINT('Engine Section(ES), Air-conditioning Section(AS), Safety and Cockpit section(SC)')
        PRINT('which sections:[Enter Code]')
        READ secCode
        IF secCode is Integer THEN
                IF secCode equal to 'ES' THEN
                        OPEN FILE 'ES-WBR.txt' as fileHandlerWBR in READ mode
                        PRINT('Which part you need to edit?[Enter Part Id]')
                        READ pardIdForSearch
                        PRINT('What is the new quantity of it')
                        READ newQuantity
                        DECLARE newData as emptyARRAY
                        readLine to return a line of text from fileHandlerWBR INTO data
                        FOR line IN data
                                CONVERT line into newLine as ARREY using tab as a delimiter
                                IF newLine[1] equal to partIdForSearch THEN
                                        newLine[2] replace by newQuantity
                                ENDIF
                                APPEND newLine INTO newData
                                PRINT(newLine)
                        CLOSE fileHandlerWBR
                        OPEN FILE 'ES-WBR.txt' as fileHandlerWBR1 in WRITE mode
                        FOR data IN newData
                                FOR item IN data
                                        WRITE item INTO fileHandlerWBR1
                                        WRITE TAB INTO fileHandlerWBR1
                                WRITE new line INTO fileHandlerWBR1
                        CLOSE fileHandlerWBR1
                ELIF secCode equal to 'AS' THEN
                        OPEN FILE 'AS-WBR.txt' as fileHandlerWBR in READ mode
                        PRINT('Which part you need to edit?[Enter Part Id]')
                        READ pardIdForSearch
                        PRINT('What is the new quantity of it')
                        READ newQuantity
                        DECLARE newData as emptyARRAY
                        readLine to return a line of text from fileHandlerWBR INTO data
                        FOR line IN data
                                CONVERT line into newLine as ARREY using tab as a delimiter
                                IF newLine[1] equal to partIdForSearch THEN
                                        newLine[2] replace by newQuantity
                                ENDIF
                                APPEND newLine INTO newData
                                PRINT(newLine)
                        CLOSE fileHandlerWBR
                        OPEN FILE 'AS-WBR.txt' as fileHandlerWBR1 in WRITE mode
                        FOR data IN newData
                                FOR item IN data
                                        WRITE item INTO fileHandlerWBR1
                                        WRITE TAB INTO fileHandlerWBR1
                                WRITE new line INTO fileHandlerWBR1
                        CLOSE fileHandlerWBR1
                ELIF secCode equal to 'SC' THEN
                        OPEN FILE 'SC-WBR.txt' as fileHandlerWBR in READ mode
                        PRINT('Which part you need to edit?[Enter Part Id]')
                        READ pardIdForSearch
                        PRINT('What is the new quantity of it')
                        READ newQuantity
                        DECLARE newData as emptyARRAY
                        readLine to return a line of text from fileHandlerWBR INTO data
                        FOR line IN data
                                CONVERT line into newLine as ARREY using tab as a delimiter
                                IF newLine[1] equal to partIdForSearch THEN
                                        newLine[2] replace by newQuantity
                                ENDIF
                                APPEND newLine INTO newData
                                PRINT(newLine)
                        CLOSE fileHandlerWBR
                        OPEN FILE 'SC-WBR.txt' as fileHandlerWBR in WRITE mode
                        FOR data IN newData
                                FOR item IN data
                                        WRITE item INTO fileHandlerWBR1
                                        WRITE TAB INTO fileHandlerWBR1
                                WRITE new line INTO fileHandlerWBR1
                        CLOSE fileHandlerWBR1
        ELSE
                PRINT ('Wrong Typing')
        ENDIF
ENDFUNCTION
```

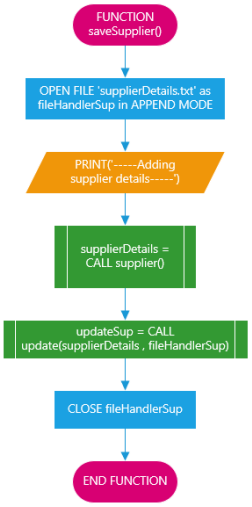
```
FUNCTION inventory()
        DECLARE inventoryLocal as emptyARRAY
        FOR i FROM 1 TO 1
                DECLARE parts as emptyARRAY
                PRINT('Enter name of part: ')
                READ partName
                WRITE partName INTO parts
                PRINT('Enter Part ID: ')
                READ partId
                WRITE partId INTO parts
                PRINT("Enter Quantity of part: ")
                READ partQuantity
                WRITE partQuantity INTO parts
                PRINT("Enter the company of supplier : ")
                READ partSupplier
                WRITE partSupplier INTO parts
                WRITE partS INTO inventoryLocal
        RETURN inventoryLocal
END FUNCTION

FUNCTION supplier()
        DECLARE supplierLocal as emptyARRAY
        FOR i FROM 1 TO 1
                DECLARE supplierParts as emptyARRAY
                PRINT('Enter name of company of supplier: ')
                READ supplierName
                WRITE supplierName INTO supplierParts
                PRINT('Enter company's contact number: ')
                READ supplierContact
                WRITE supplierContact INTO supplierParts
                PRINT("Enter partsID of supplier supplied: ")
                READ supplierSupplied
                WRITE supplierSupplied INTO supplierParts
                WRITE supplierParts INTO supplierLocal
        RETURN supplierLocal
ENDFUNCTION

FUNCTION update(newData, fileHandler)
        FOR data IN newData
                FOR item IN data
                        WRITE item INTO fileHandler
                        WRITE tab INTO fileHandler
                WRITE new line INTO fileHandler
ENDFUNCTION
```

Flowcharts:

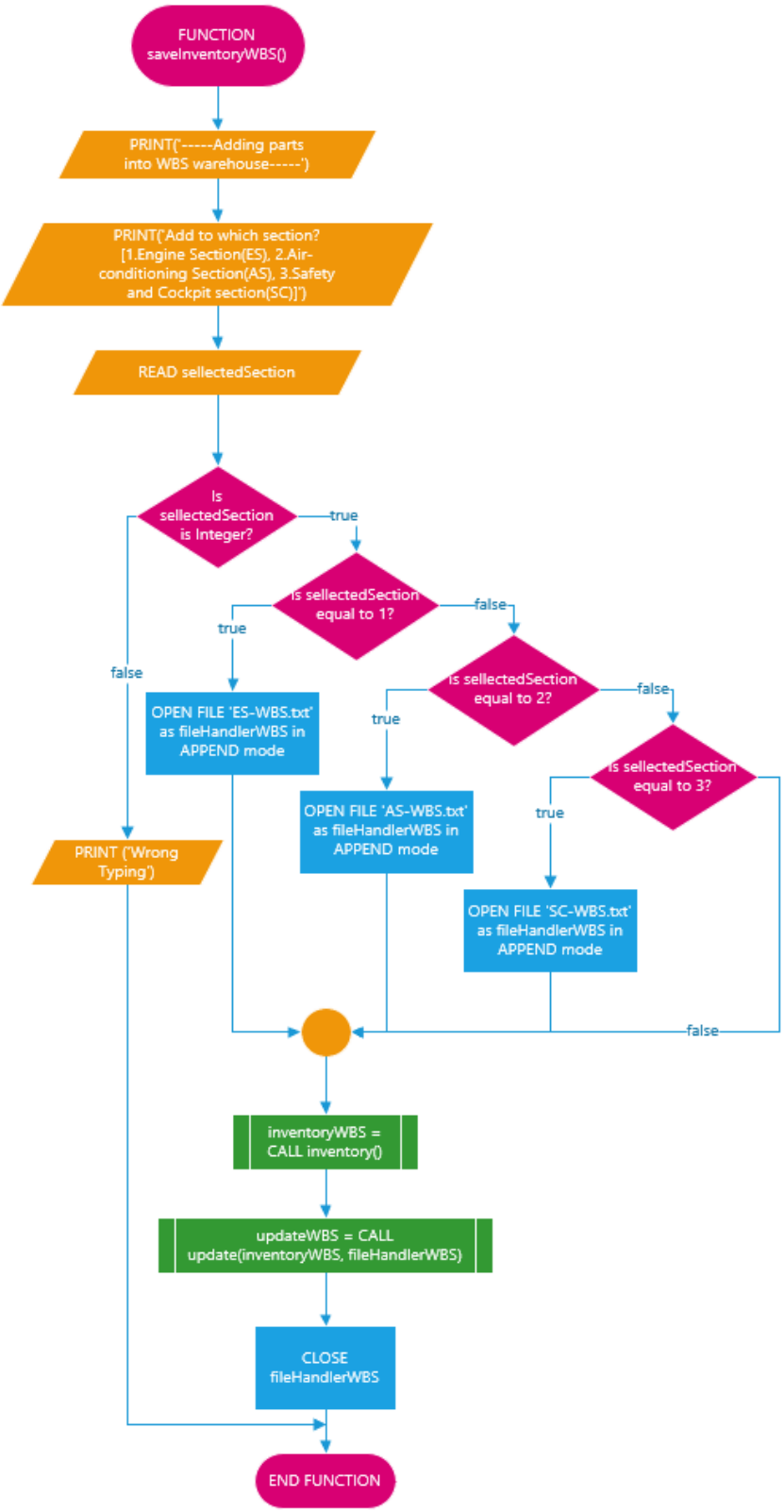
updateInventory():



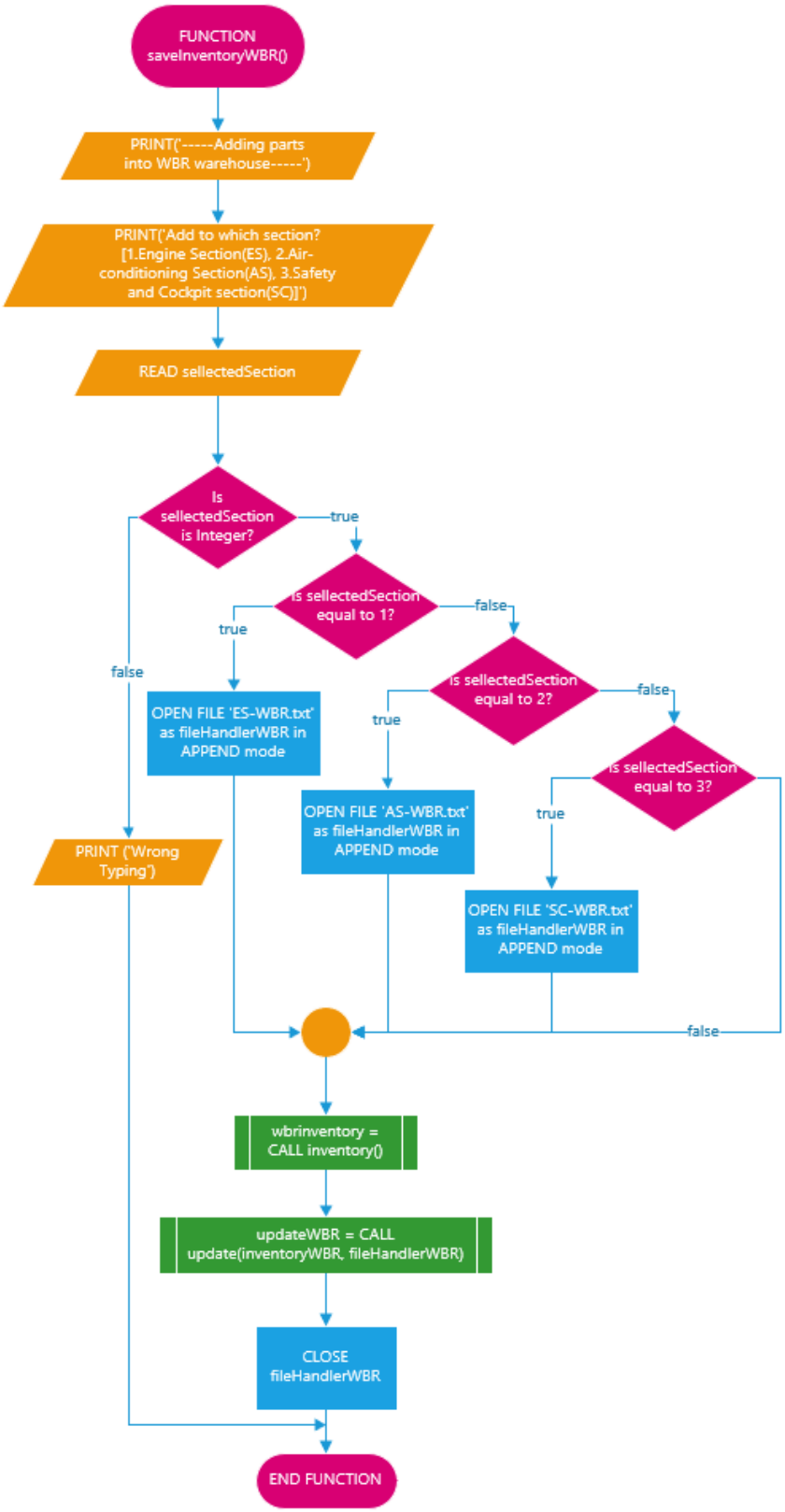
saveSupplier():



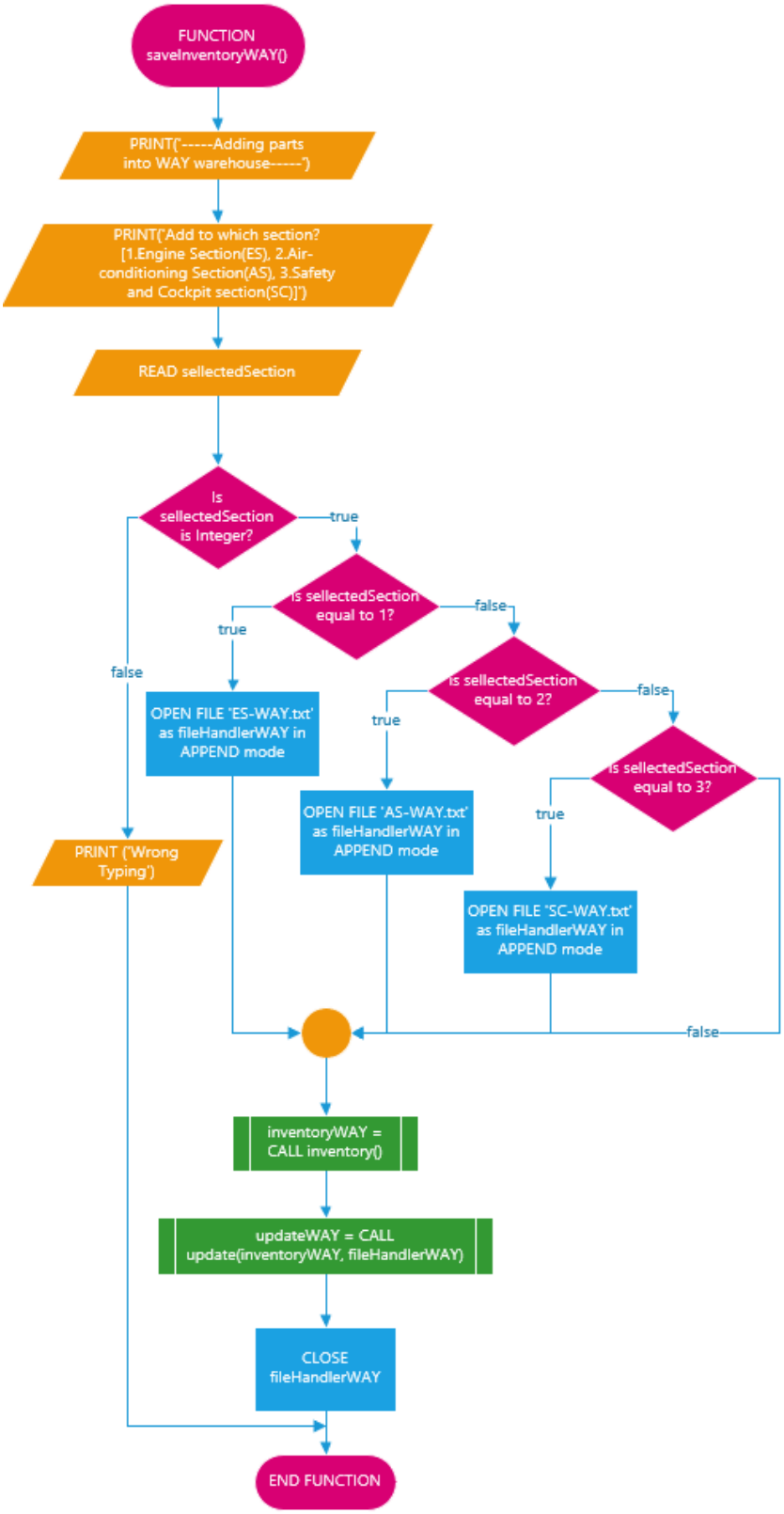
saveInventoryWBS():



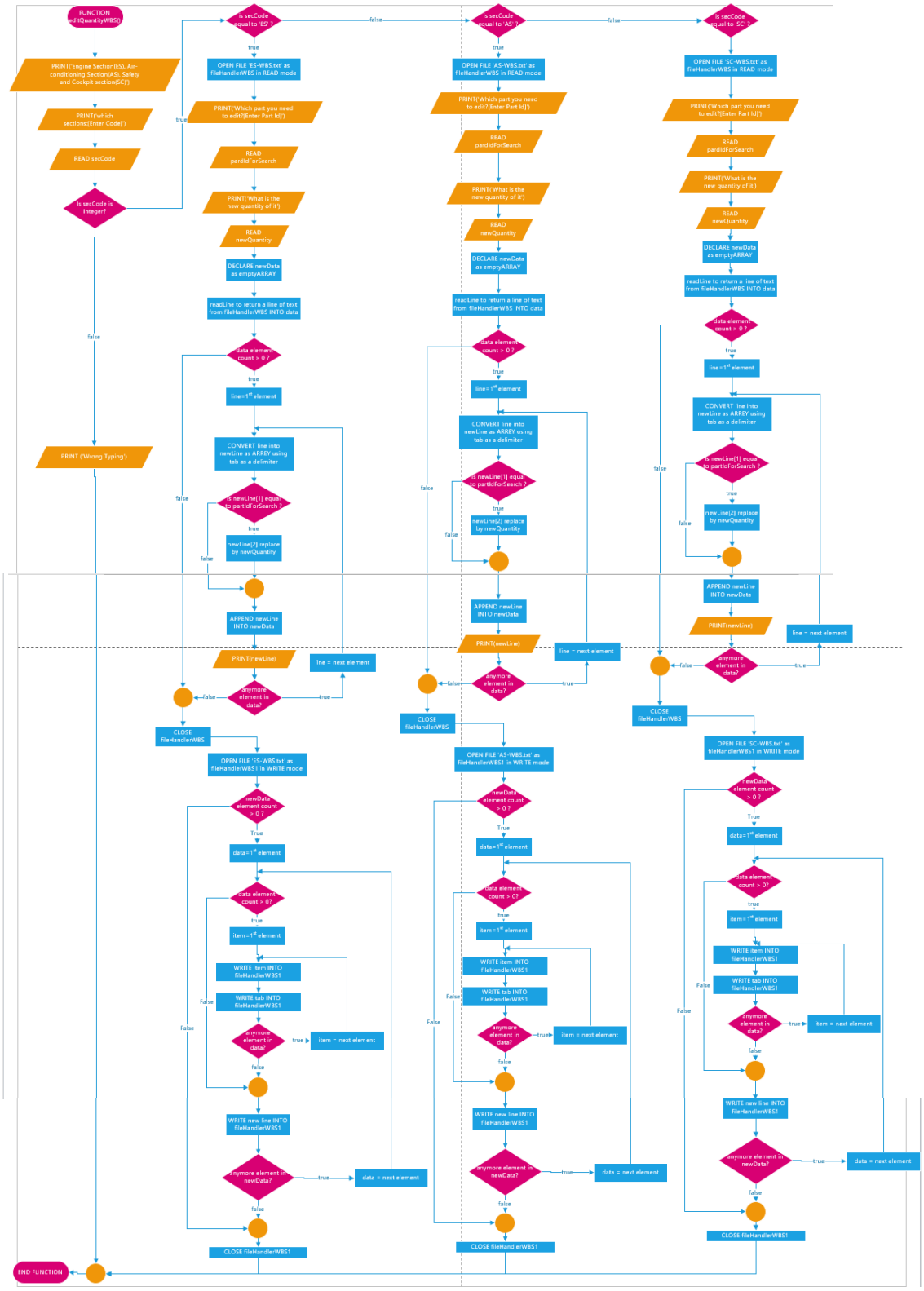
saveInventoryWBR():



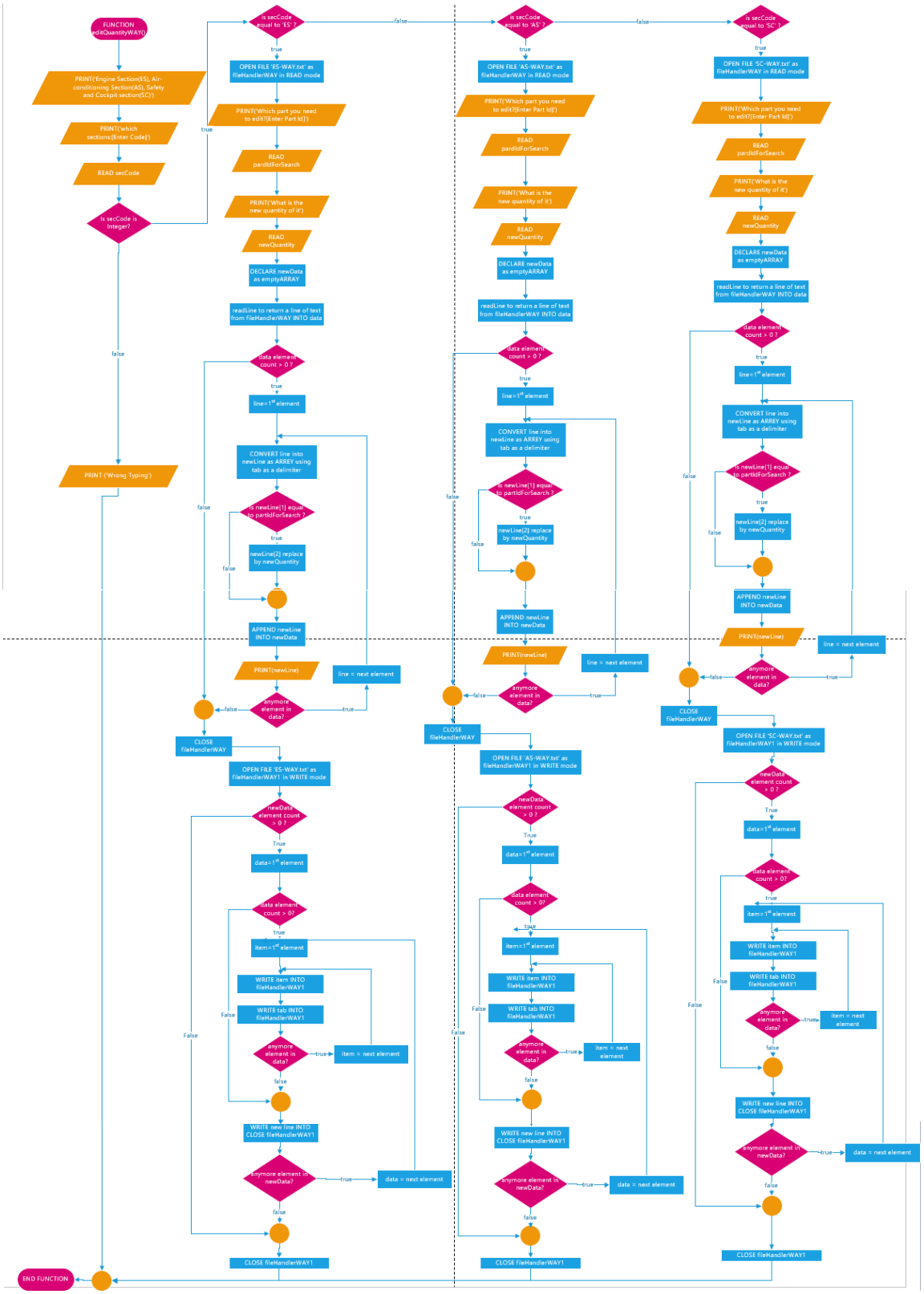
saveInventoryWAY():



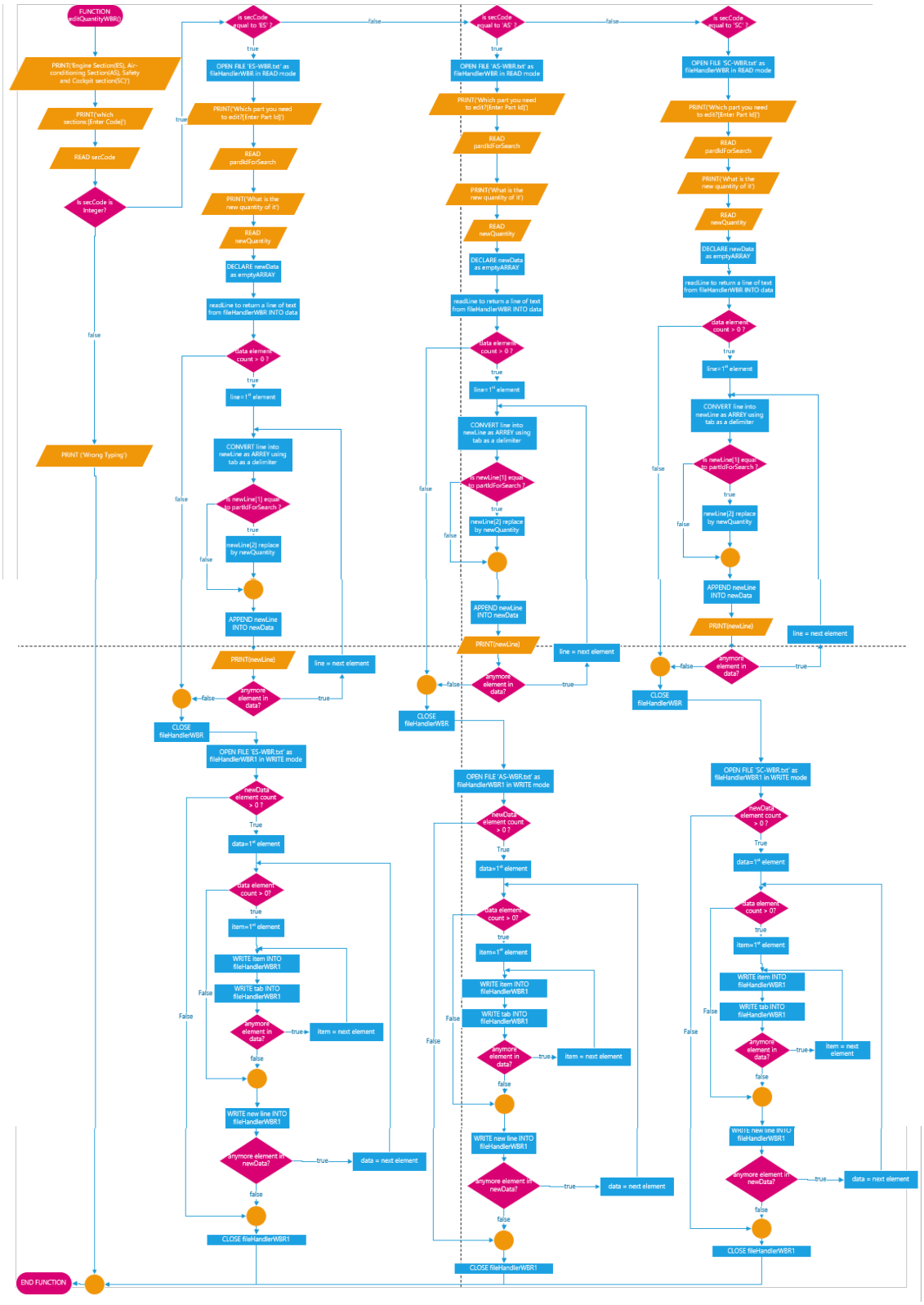
editQuantityWBS():



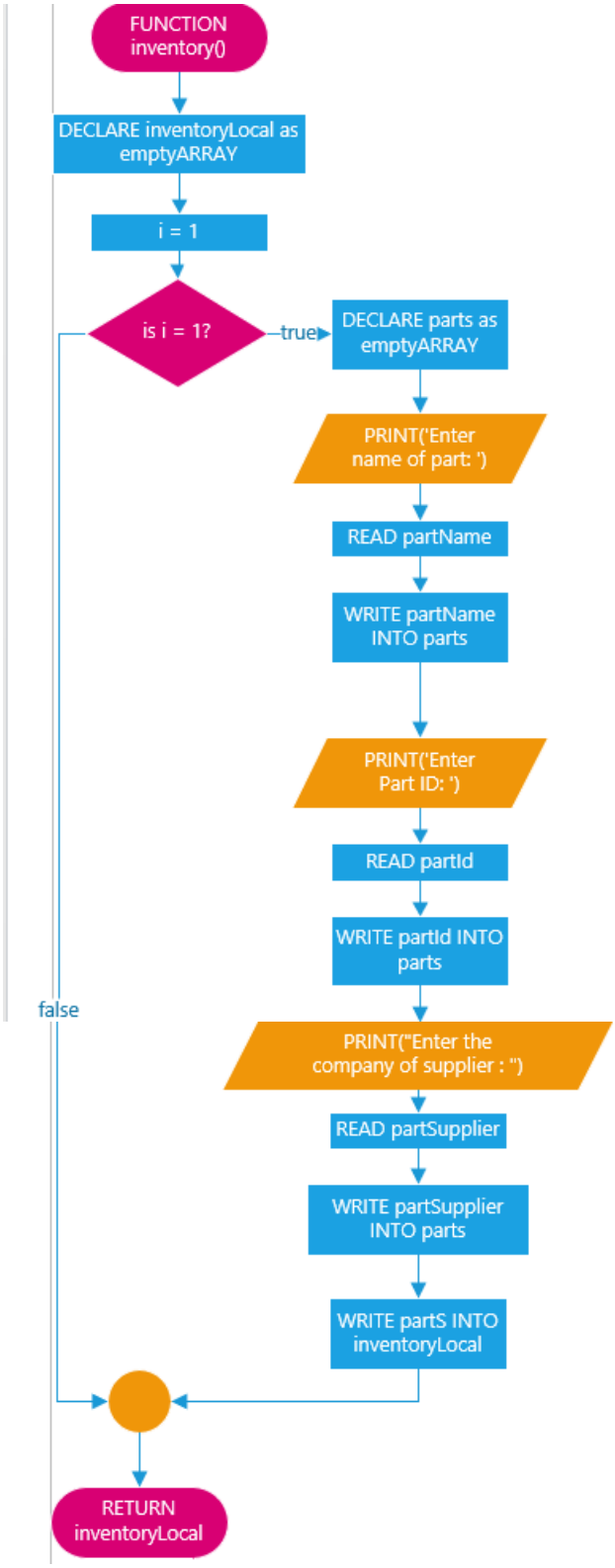
editQuantityWAY():



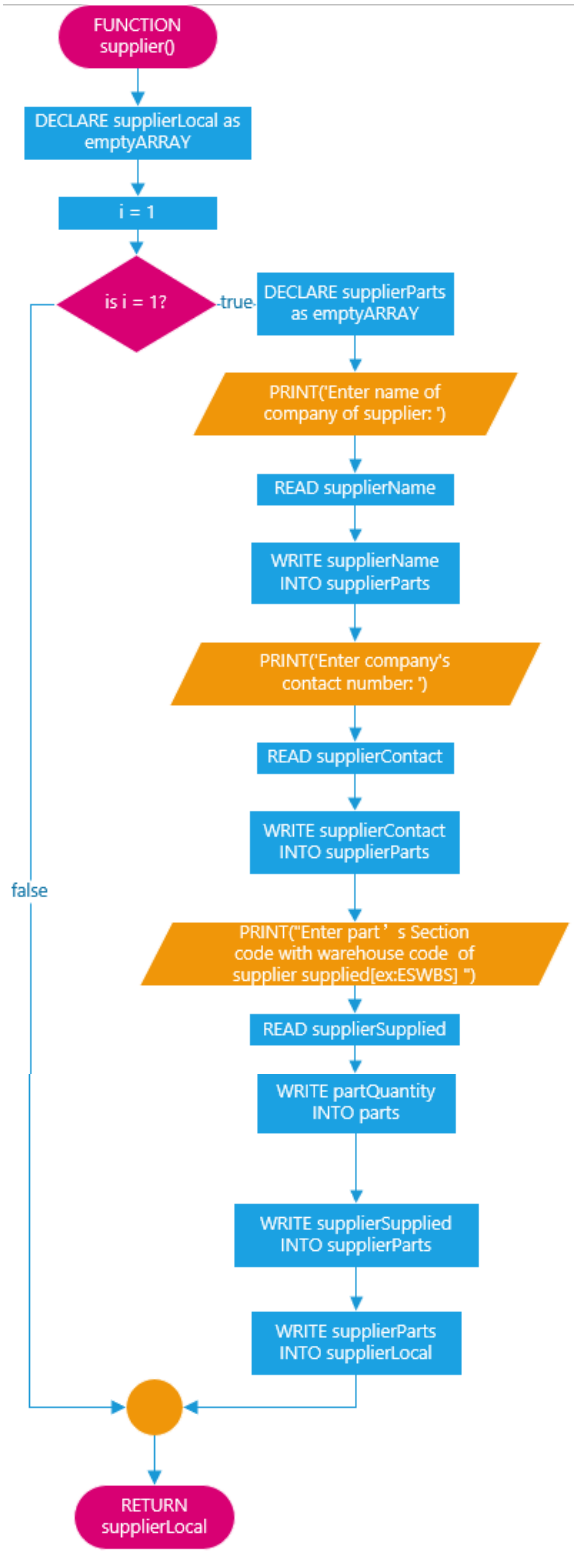
editQuantityWBR():



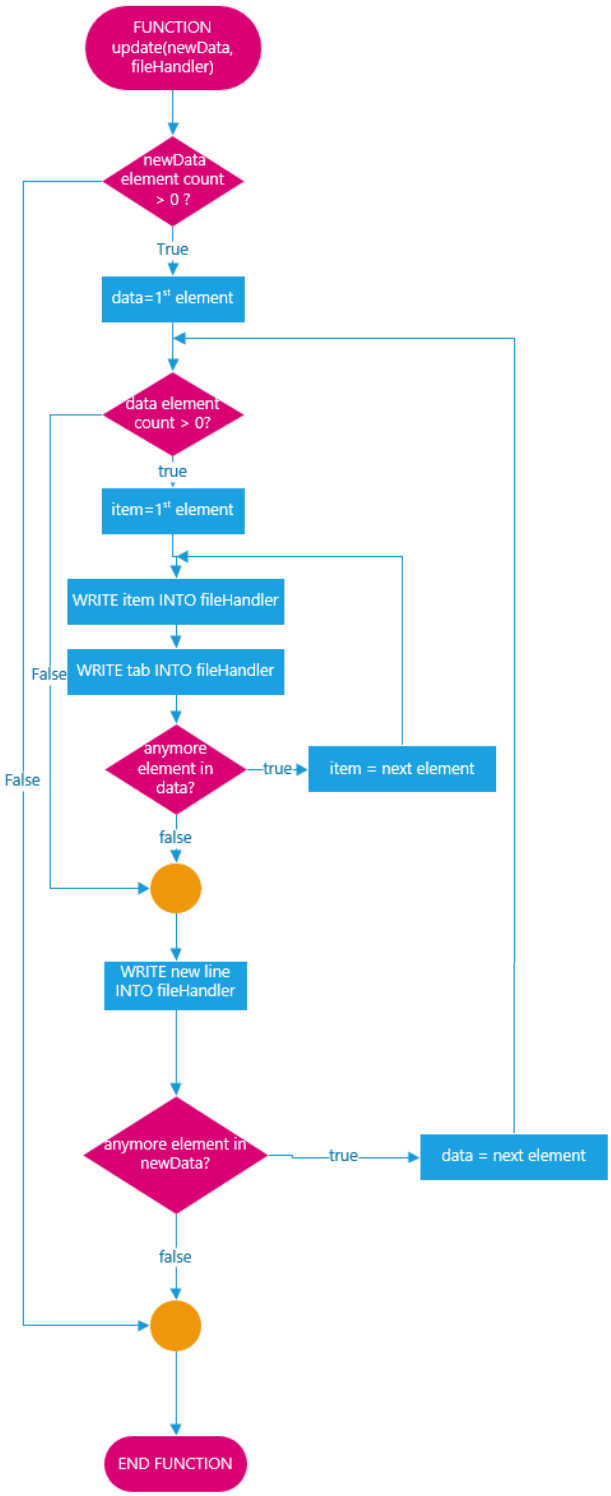
inventory():


FUNCTION inventory()
DECLARE inventoryLocal as emptyARRAY
i = 1
is i = 1?
true
DECLARE parts as emptyARRAY
PRINT('Enter name of part: ')
READ partName
WRITE partName INTO parts
PRINT('Enter Part ID: ')
READ partId
WRITE partId INTO parts
false
PRINT("Enter the company of supplier : ")
READ partSupplier
WRITE partSupplier INTO parts
WRITE partS INTO inventoryLocal
RETURN inventoryLocal

supplier():


FUNCTION supplier()
DECLARE supplierLocal as emptyARRAY
i = 1
is i = 1?
true
DECLARE supplierParts as emptyARRAY
PRINT('Enter name of company of supplier: ')
READ supplierName
WRITE supplierName INTO supplierParts
PRINT('Enter company's contact number: ')
READ supplierContact
WRITE supplierContact INTO supplierParts
PRINT("Enter part ' s Section code with warehouse code of supplier supplied[ex:ESWBS] ")
READ supplierSupplied
WRITE partQuantity INTO parts
WRITE supplierSupplied INTO supplierParts
WRITE supplierParts INTO supplierLocal
false
RETURN supplierLocal

update(newData, fileHandler):



Part inventory tracking

Pseudocode:

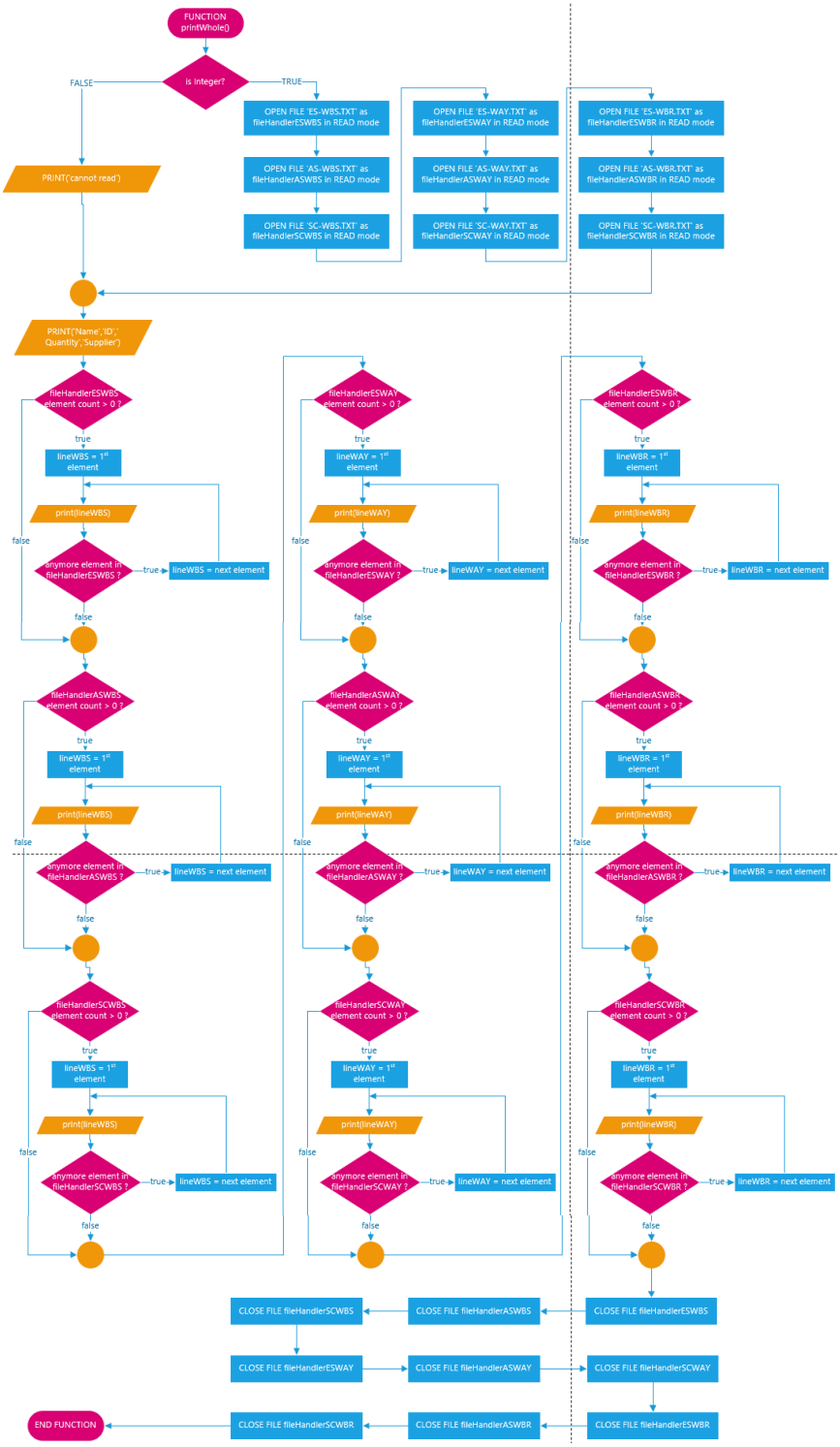
```
FUNCTION printWhole()
        IF integer THEN
                OPEN FILE 'ES-WBS.TXT' as fileHandlerESWBS in READ mode
                OPEN FILE 'AS-WBS.TXT' as fileHandlerASWBS in READ mode
                OPEN FILE 'SC-WBS.TXT' as fileHandlerSCWBS in READ mode
                OPEN FILE 'ES-WAY.TXT' as fileHandlerESWAY in READ mode
                OPEN FILE 'AS-WAY.TXT' as fileHandlerASWAY in READ mode
                OPEN FILE 'SC-WAY.TXT' as fileHandlerSCWAY in READ mode
                OPEN FILE 'ES-WBR.TXT' as fileHandlerESWBR in READ mode
                OPEN FILE 'AS-WBR.TXT' as fileHandlerASWBR in READ mode
                OPEN FILE 'SC-WBR.TXT' as fileHandlerSCWBR in READ mode
        ELSE
                PRINT('cannot read')
        PRINT('Name','ID','Quantity','Supplier')
        for lineWBS in fileHandlerESWBS
                print(lineWBS)
        for lineWBS in fileHandlerASWBS
                print(lineWBS)
        for lineWBS in fileHandlerSCWBS
                print(lineWBS)
        for lineWAY in fileHandlerESWAY
                print(lineWAY)
        for lineWAY in fileHandlerASWAY
                print(lineWAY)
        for lineWAY in fileHandlerSCWAY
                print(lineWAY)
        for lineWBR in fileHandlerESWBR
                print(lineWBR)
        for lineWBR in fileHandlerASWBR
                print(lineWBR)
        for lineWBR in fileHandlerSCWBR
                print(lineWBR)
        CLOSE FILE fileHandlerESWBS
        CLOSE FILE fileHandlerASWBS
        CLOSE FILE fileHandlerSCWBS
        CLOSE FILE fileHandlerESWAY
        CLOSE FILE fileHandlerASWAY
        CLOSE FILE fileHandlerSCWAY
        CLOSE FILE fileHandlerESWBR
        CLOSE FILE fileHandlerASWBR
        CLOSE FILE fileHandlerSCWBR
ENDFUNCTION
```

```
FUNCTION printLesser()
	IF integer THEN
		OPEN FILE 'ES-WBS.TXT' as fileHandlerESWBS in READ mode
		OPEN FILE 'AS-WBS.TXT' as fileHandlerASWBS in READ mode
		OPEN FILE 'SC-WBS.TXT' as fileHandlerSCWBS in READ mode
		OPEN FILE 'ES-WAY.TXT' as fileHandlerESWAY in READ mode
		OPEN FILE 'AS-WAY.TXT' as fileHandlerASWAY in READ mode
		OPEN FILE 'SC-WAY.TXT' as fileHandlerSCWAY in READ mode
		OPEN FILE 'ES-WBR.TXT' as fileHandlerESWBR in READ mode
		OPEN FILE 'AS-WBR.TXT' as fileHandlerASWBR in READ mode
		OPEN FILE 'SC-WBR.TXT' as fileHandlerSCWBR in READ mode
		DECLARE lesser AS an ARRAY ['0','1','2','3','4','5','6','7','8','9']
	ELSE
		PRINT('read file error')
	PRINT'Parts that quantity less than 10:'
	PRINT('Name','ID','Quantity','Supplier')
	IF integer THEN
		FOR lineWBS in fileHandlerESWBS
			CONVERT lineWBS INTO newLineWBS as ARRAY using tab as a delimiter
			IF newLineWBS[2] NOT IN lesser THEN
				CONTINUE LOOP
			print(lineWBS)
		FOR lineWBS in fileHandlerASWBS
			CONVERT lineWBS INTO newLineWBS as ARRAY using tab as a delimiter
			IF newLineWBS[2] NOT IN lesser THEN
				CONTINUE LOOP
			print(lineWBS)
		FOR lineWBS in fileHandlerSCWBS
			CONVERT lineWBS INTO newLineWBS as ARRAY using tab as a delimiter
			IF newLineWBS[2] NOT IN lesser THEN
				CONTINUE LOOP
			print(lineWBS)
		FOR lineWAY IN fileHandlerESWAY
			CONVERT lineWAY INTO newLineWAY as ARRAY using tab as a delimiter
			IF newLineWAY[2] NOT IN lesser THEN
				CONTINUE LOOP
			print(lineWAY)
		FOR lineWAY IN fileHandlerASWAY
			CONVERT lineWAY INTO newLineWAY as ARRAY using tab as a delimiter
			IF newLineWAY[2] NOT IN lesser THEN
				CONTINUE LOOP
			print(lineWAY)
		FOR lineWAY IN fileHandlerSCWAY
			CONVERT lineWAY INTO newLineWAY as ARRAY using tab as a delimiter
			IF newLineWAY[2] NOT IN lesser THEN
				CONTINUE LOOP
			print(lineWAY)
		FOR lineWBR IN fileHandlerESWBR
			CONVERT lineWBR INTO newLineWBR as ARRAY using tab as a delimiter
			IF newLineWBR[2] NOT IN lesser THEN
				CONTINUE LOOP
			print(lineWBR)
		FOR lineWBR IN fileHandlerASWBR
			CONVERT lineWBR INTO newLineWBR as ARRAY using tab as a delimiter
			IF newLineWBR[2] NOT IN lesser THEN
				CONTINUE LOOP
			print(lineWBR)
		FOR lineWBR IN fileHandlerSCWBR
			CONVERT lineWBR INTO newLineWBR as ARRAY using tab as a delimiter
			IF newLineWBR[2] NOT IN lesser THEN
				CONTINUE LOOP
			print(lineWBR)
	ELSE
		PRINT('print error')
	CLOSE FILE fileHandlerESWBS
	CLOSE FILE fileHandlerASWBS
	CLOSE FILE fileHandlerSCWBS
	CLOSE FILE fileHandlerESWAY
	CLOSE FILE fileHandlerASWAY
	CLOSE FILE fileHandlerSCWAY
	CLOSE FILE fileHandlerESWBR
	CLOSE FILE fileHandlerASWBR
	CLOSE FILE fileHandlerSCWBR
ENDFUNCTION
```

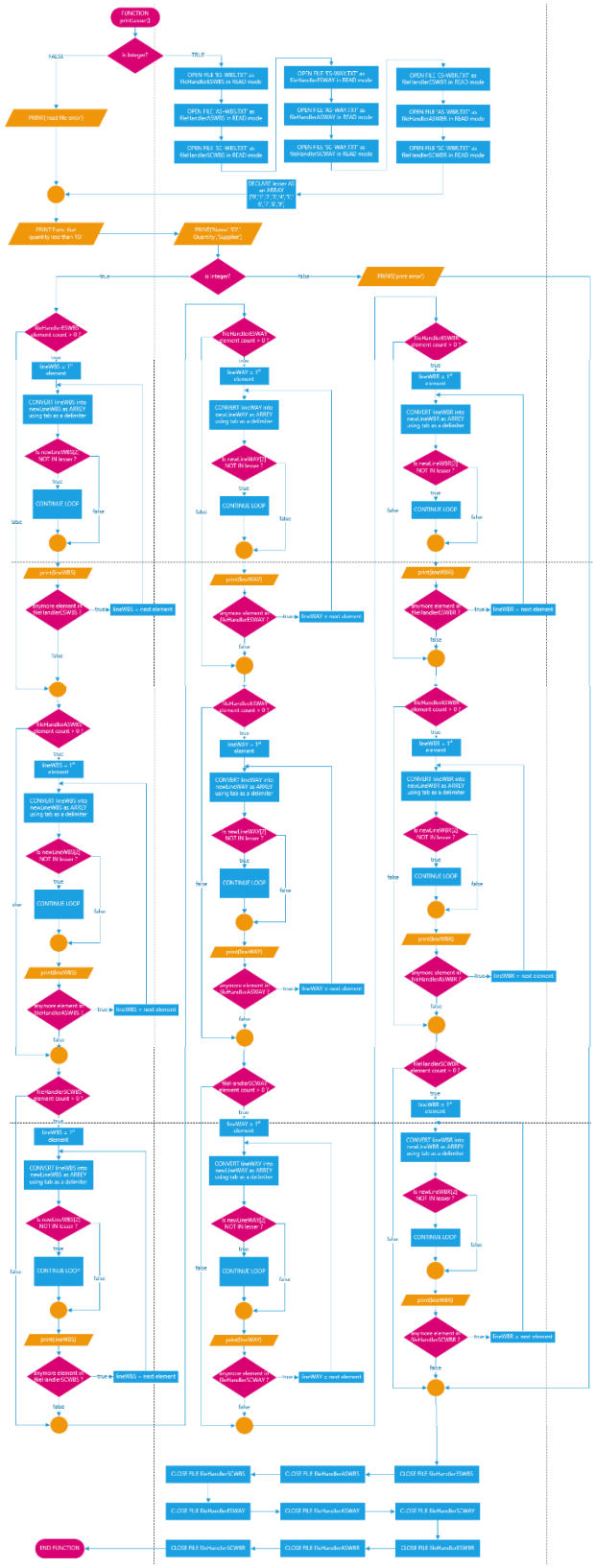
```
FUNCTION printWarehouseParts()
        PRINT('Enter number of warehouse you want to print(1.WBS,2.WAY,3.WBR)')
        READ warehouseChoosed
        IF warehouseChoosed is integer THEN
                PRINT('Name','ID','Quantity','Supplier')
                IF warehouseChoosed equal to 1 THEN
                        OPEN FILE 'ES-WBS.TXT' as fileHandlerESWBS in READ mode
                        OPEN FILE 'AS-WBS.TXT' as fileHandlerASWBS in READ mode
                        OPEN FILE 'SC-WBS.TXT' as fileHandlerSCWBS in READ mode
                        PRINT('the parts of engine section(ES)')
                        FOR lineWBS IN fileHandlerESWBS
                                print(lineWBS)
                        PRINT('the parts of air-conditioning section(AS)')
                        FOR lineWBS IN fileHandlerASWBS
                                print(lineWBS)
                        PRINT('the parts of safety and cockpit section(SC)')
                        FOR lineWBS IN fileHandlerSCWBS
                                print(lineWBS)
                        CLOSE FILE fileHandlerESWBS
                        CLOSE FILE fileHandlerASWBS
                        CLOSE FILE fileHandlerSCWBS
                ELIF warehouseChoosed equal to 2 THEN
                        OPEN FILE 'ES-WAY.TXT' as fileHandlerESWAY in READ mode
                        OPEN FILE 'AS-WAY.TXT' as fileHandlerASWAY in READ mode
                        OPEN FILE 'SC-WAY.TXT' as fileHandlerSCWAY in READ mode
                        PRINT('the parts of engine section(ES)')
                        FOR lineWAY IN fileHandlerESWAY
                                print(lineWAY)
                        PRINT('the parts of air-conditioning section(AS)')
                        FOR lineWAY IN fileHandlerASWAY
                                print(lineWAY)
                        PRINT('the parts of safety and cockpit section(SC)')
                        FOR lineWAY IN fileHandlerSCWAY
                                print(lineWAY)
                        CLOSE FILE fileHandlerESWAY
                        CLOSE FILE fileHandlerASWAY
                        CLOSE FILE fileHandlerSCWAY
                ELIF warehouseChoosed equal to 3 THEN
                        OPEN FILE 'ES-WBR.TXT' as fileHandlerESWBR in READ mode
                        OPEN FILE 'AS-WBR.TXT' as fileHandlerASWBR in READ mode
                        OPEN FILE 'SC-WBR.TXT' as fileHandlerSCWBR in READ mode
                        PRINT('the parts of engine section(ES)')
                        FOR lineWBR IN fileHandlerESWBR
                                print(lineWBR)
                        PRINT('the parts of air-conditioning section(AS)')
                        FOR lineWBR IN fileHandlerASWBR
                                print(lineWBR)
                        PRINT('the parts of safety and cockpit section(SC)')
                        FOR lineWBR IN fileHandlerSCWBR
                                print(lineWBR)
                        CLOSE FILE fileHandlerESWBR
                        CLOSE FILE fileHandlerASWBR
                        CLOSE FILE fileHandlerSCWBR
                ENDIF
        ELSE
                PRINT ('Wrong Typing')
        ENDIF
ENDFUNCTION
```

Flowchart:

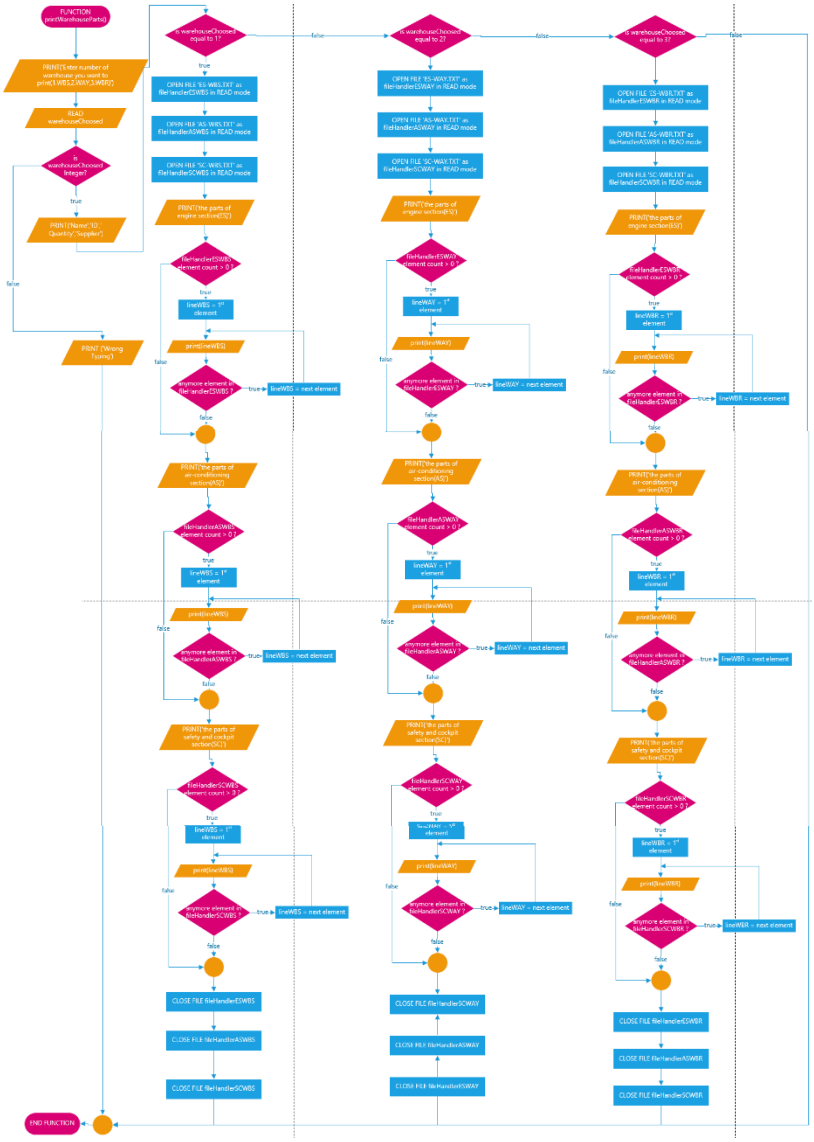
printWhole():



printLesser():



printWarehouseParts():



Searching Functionalities

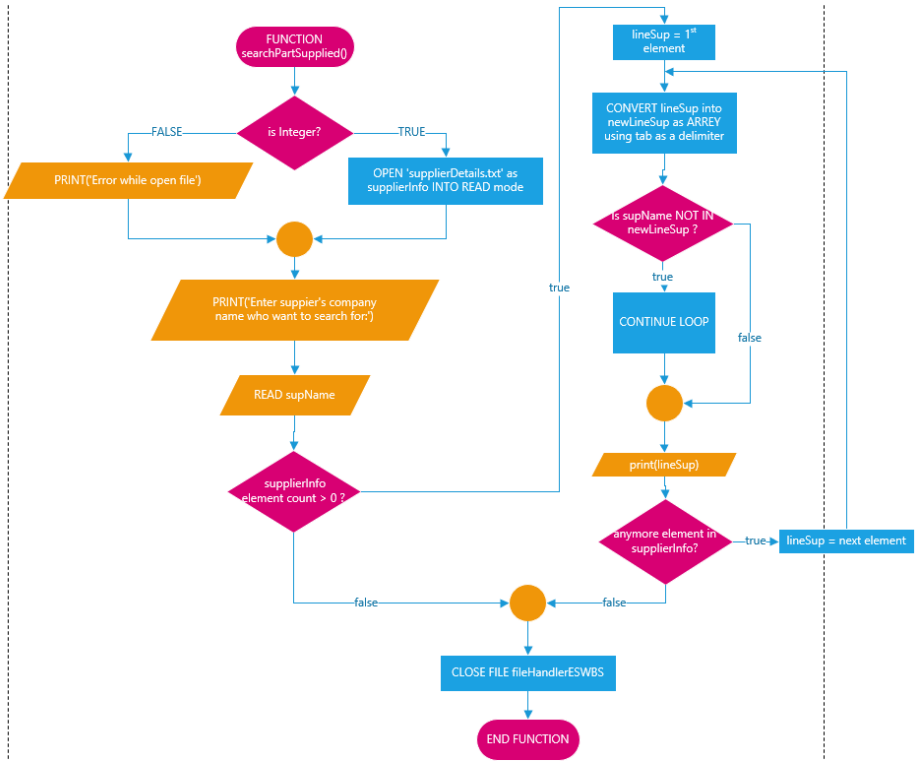
Pseudocode:

```
FUNCTION searchPart()
        OPEN FILE 'ES-WBS.TXT' as fileHandlerESWBS in READ mode
        OPEN FILE 'AS-WBS.TXT' as fileHandlerASWBS in READ mode
        OPEN FILE 'SC-WBS.TXT' as fileHandlerSCWBS in READ mode
        OPEN FILE 'ES-WAY.TXT' as fileHandlerESWAY in READ mode
        OPEN FILE 'AS-WAY.TXT' as fileHandlerASWAY in READ mode
        OPEN FILE 'SC-WAY.TXT' as fileHandlerSCWAY in READ mode
        OPEN FILE 'ES-WBR.TXT' as fileHandlerESWBR in READ mode
        OPEN FILE 'AS-WBR.TXT' as fileHandlerASWBR in READ mode
        OPEN FILE 'SC-WBR.TXT' as fileHandlerSCWBR in READ mode
        PRINT ('Enter Part Id you want to search')
        READ keyword
        PRINT('Name','ID','Quantity','Supplier')
        IF keyword is integer THEN
                FOR lineWBS IN fileHandlerESWBS
                        CONVERT lineWBS INTO newLineWBS as ARRAY using tab as a delimiter
                        IF newLineWBS[1] equal to keyword
                                print(newLineWBS)
                FOR lineWBS IN fileHandlerASWBS
                        CONVERT lineWBS INTO newLineWBS as ARRAY using tab as a delimiter
                        IF newLineWBS[1] equal to keyword
                                print(newLineWBS)
                FOR lineWBS IN fileHandlerSCWBS
                        CONVERT lineWBS INTO newLineWBS as ARRAY using tab as a delimiter
                        IF newLineWBS[1] equal to keyword
                                print(newLineWBS)
                FOR lineWAY IN fileHandlerESWAY
                        CONVERT lineWAY INTO newLineWAY as ARRAY using tab as a delimiter
                        IF newLineWAY[1] equal to keyword
                                print(newLineWAY)
                FOR lineWAY IN fileHandlerASWAY
                        CONVERT lineWAY INTO newLineWAY as ARRAY using tab as a delimiter
                        IF newLineWAY[1] equal to keyword
                                print(newLineWAY)
                FOR lineWAY IN fileHandlerSCWAY
                        CONVERT lineWAY INTO newLineWAY as ARRAY using tab as a delimiter
                        IF newLineWAY[1] equal to keyword
                                print(newLineWAY)
                FOR lineWBR IN fileHandlerESWBR
                        CONVERT lineWBR INTO newLineWBR as ARRAY using tab as a delimiter
                        IF newLineWBR[1] equal to keyword
                                print(newlineWBR)
                FOR lineWBR IN fileHandlerASWBR
                        CONVERT lineWBR INTO newLineWBR as ARRAY using tab as a delimiter
                        IF newLineWBR[1] equal to keyword
                                print(newlineWBR)
                FOR lineWBR IN fileHandlerSCWBR
                        CONVERT lineWBR INTO newLineWBR as ARRAY using tab as a delimiter
                        IF newLineWBR[1] equal to keyword
                                print(newlineWBR)
        ELSE
                PRINT('searching error')
        ENDIF
        CLOSE FILE fileHandlerESWBS
        CLOSE FILE fileHandlerASWBS
        CLOSE FILE fileHandlerSCWBS
        CLOSE FILE fileHandlerESWAY
        CLOSE FILE fileHandlerASWAY
        CLOSE FILE fileHandlerSCWAY
        CLOSE FILE fileHandlerESWBR
        CLOSE FILE fileHandlerASWBR
        CLOSE FILE fileHandlerSCWBR
ENDFUNCTION
```

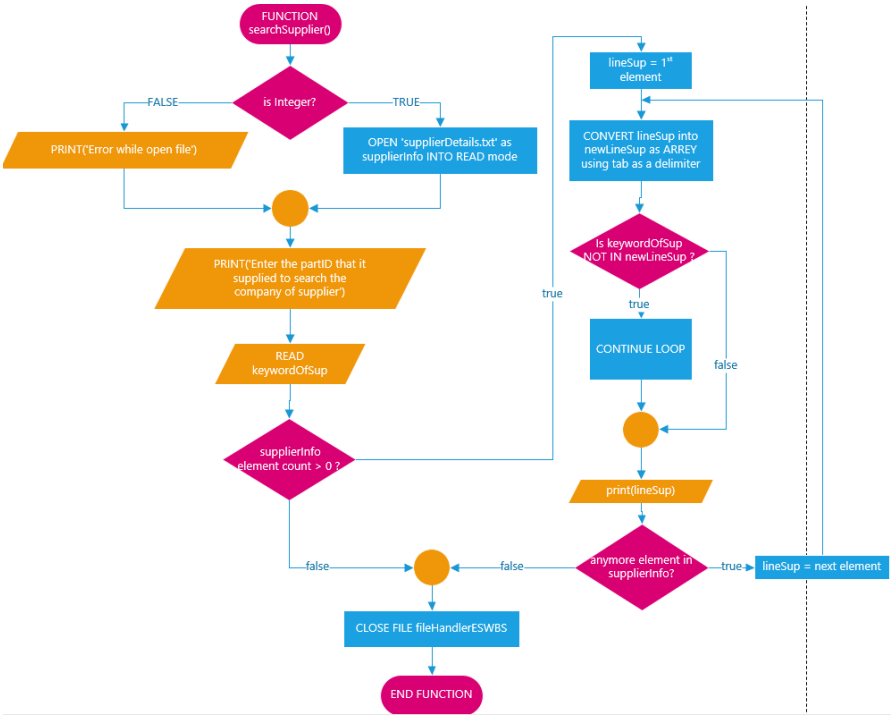
```
FUNCTION searchSupplier()
        IF integer THEN
                OPEN 'supplierDetails.txt' as supplierInfo INTO READ mode
        ELSE
                PRINT('Error while open file')
        PRINT('Enter the partID that it supplied to search the company of supplier')
        READ keywordOfSup
        FOR lineSup IN supplierInfo
                CONVERT lineSup INTO newLineSup as ARRAY using tab as a delimiter
                IF keywordOfSup NOT IN newLineSup
                        CONTINUE LOOP
                ENDIF
                print(lineSup )
        CLOSE FILE supplierInfo
ENDFUNCTION

FUNCTION searchPartSupplied()
        IF integer THEN
                OPEN 'supplierDetails.txt' as supplierInfo INTO READ mode
        ELSE
                PRINT('Error while open file')
        PRINT('Enter suppier's company name who want to search for:')
        READ supName
        FOR lineSup IN supplierInfo
                CONVERT lineSup INTO newLineSup as ARRAY using tab as a delimiter
                IF supName NOT IN newLineSup
                        CONTINUE LOOP
                ENDIF
                print(lineSup )
        CLOSE FILE supplierInfo
ENDFUNCTION
```

Flowchart:

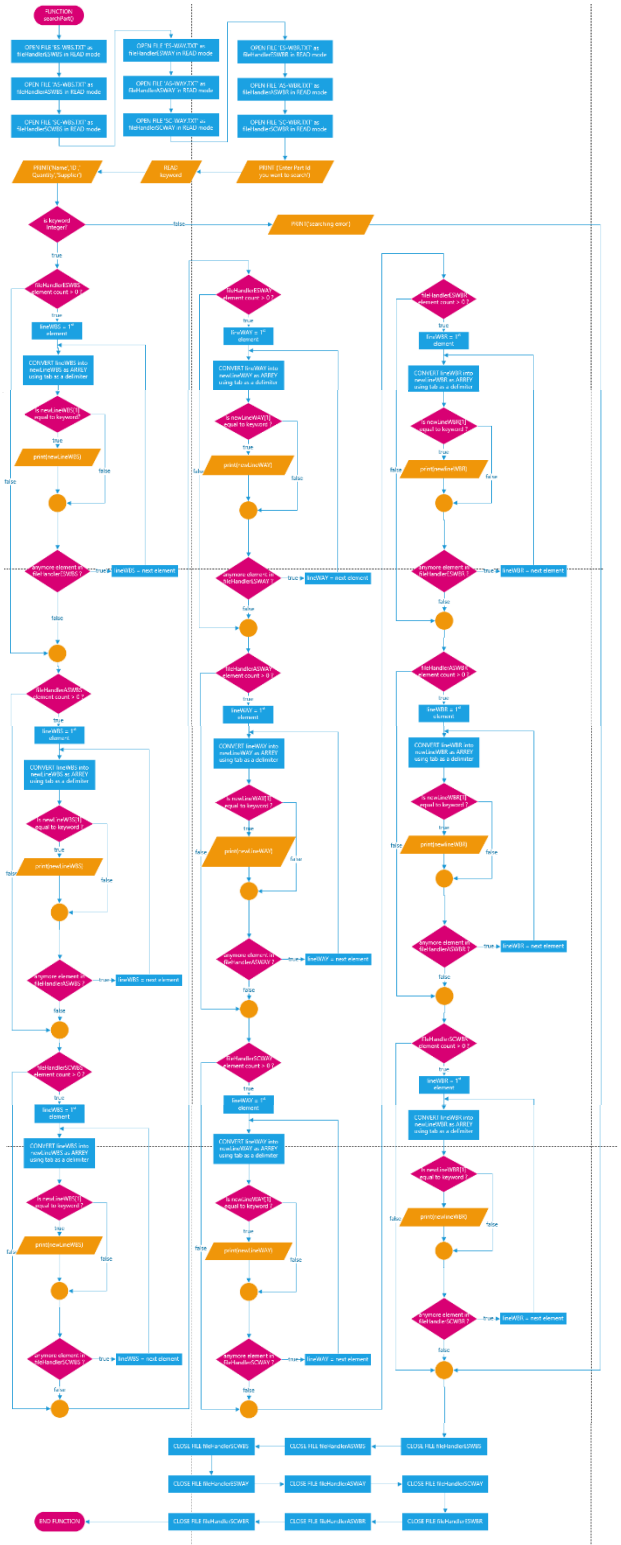
searchPartSupplied():



searchSupplier():



searchPart():

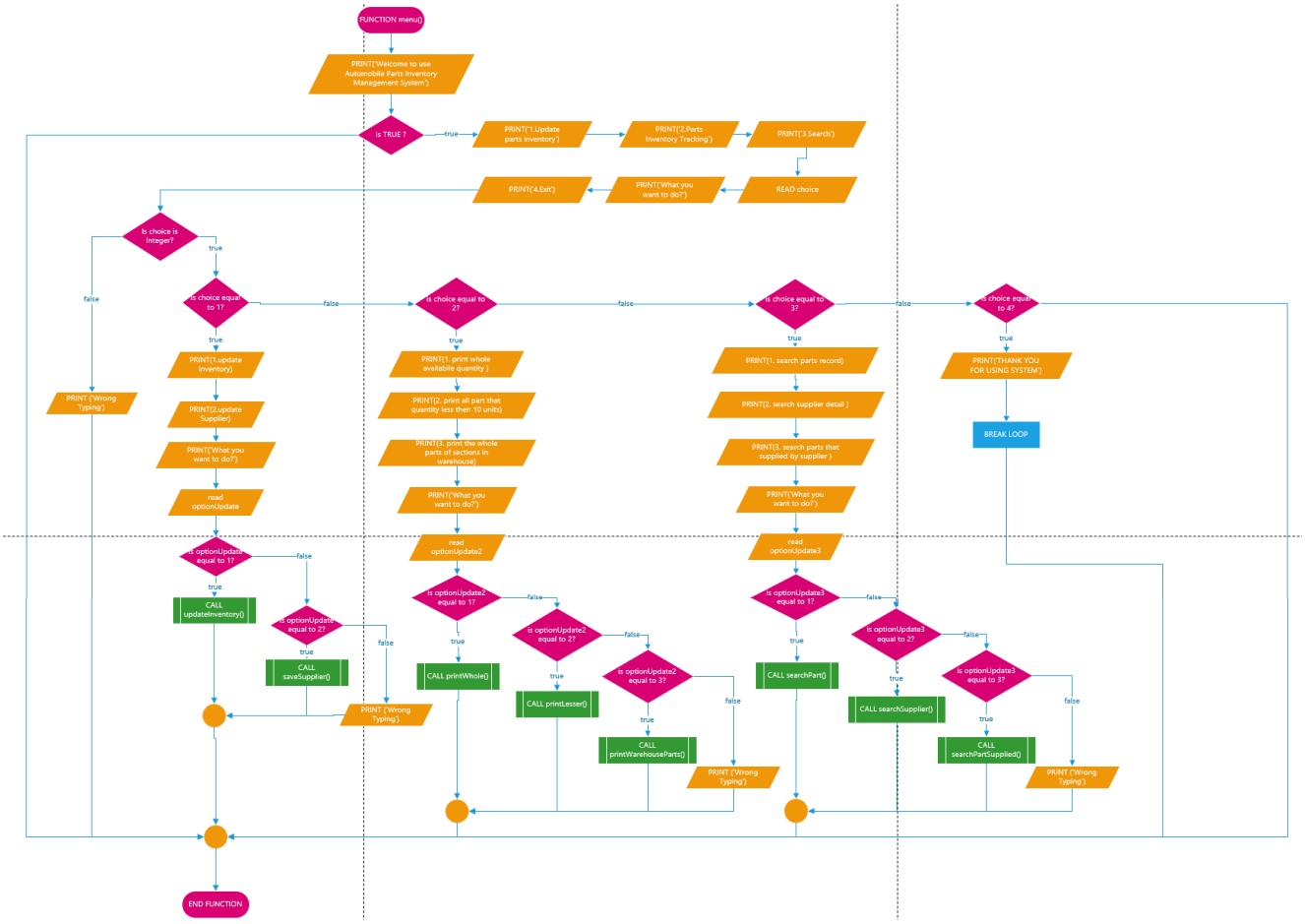


Menu Function

Pseudocode:

```
FUNCTION menu()
        PRINT('Welcome to use Automobile Parts Inventory Management System')
        DOWHILE TRUE
                PRINT('1.Update parts inventory')
                PRINT('2.Parts Inventory Tracking')
                PRINT('3.Search')
                PRINT('4.Exit')
                PRINT('What you want to do?')
                READ choice
                IF choice is Integer
                        IF choice equal to 1 THEN
                                PRINT(1.update inventory)
                                PRINT(2.update Supplier)
                                PRINT('What you want to do?')
                                read optionUpdate
                                IF optionUpdate equal to 1 THEN
                                        CALL updateInventory()
                                ELIF optionUpdate equal to 2 THEN
                                        CALL saveSupplier()
                                ELSE
                                        PRINT ('Wrong Typing')
                        ELIF choice equal to 2 THEN
                                PRINT(1. print whole availabile quantity )
                                PRINT(2. print all part that quantity less then 10 units)
                                PRINT(3. print the whole parts of sections in warehouse)
                                PRINT('What you want to do?')
                                read optionUpdate2
                                IF optionUpdate2 equal to 1 THEN
                                        CALL printWhole()
                                ELIF optionUpdate2 equal to 2 THEN
                                        CALL printLesser()
                                ELIF optionUpdate2 equal to 3 THEN
                                        CALL printWarehouseParts()
                                ELSE
                                        PRINT ('Wrong Typing')
                        ELIF choice equal to 3 THEN
                                PRINT(1. search parts record)
                                PRINT(2. search supplier detail )
                                PRINT(3. search parts that supplied by supplier )
                                PRINT('What you want to do?')
                                read optionUpdate3
                                IF optionUpdate3 equal to 1 THEN
                                        CALL searchPart()
                                ELIF optionUpdate3 equal to 2 THEN
                                        CALL searchSupplier()
                                ELIF optionUpdate3 equal to 3 THEN
                                        CALL searchPartSupplied()
                                ELSE
                                        PRINT ('Wrong Typing')
                        ELIF choice equal to 4 THEN
                                PRINT('THANK YOU FOR USING SYSTEM')
                                BREAK LOOP
                ELSE
                        PRINT ('Wrong Typing')
        ENDWHILE
ENDFUNCTION
```

menu():


FUNCTION menu()
PRINT('Welcome to use Automobile Parts Inventory Management System')
is TRUE ?
true
PRINT('1.Update parts inventory')
PRINT('2.Parts Inventory Tracking')
PRINT('3.Search')
READ choice
PRINT('What you want to do?')
PRINT('4.Exit')
is choice is Integer?
false
PRINT ('Wrong Typing')
true
is choice equal to 1?
false
true
PRINT('1.update inventory')
PRINT('2.update Supplier')
PRINT('What you want to do?')
read optionUpdate
is optionUpdate equal to 1?
true
CALL updateInventory()
false
is optionUpdate equal to 2?
true
CALL saveSupplier()
false
PRINT ('Wrong Typing')
is choice equal to 2?
true
PRINT(1. print whole availiable quantity)
PRINT(2. print all part that quantity less then 10 units)
PRINT(3. print the whole parts of sections in warehouse)
PRINT('What you want to do?')
read optionUpdate2
is optionUpdate2 equal to 1?
true
CALL printWhole()
false
is optionUpdate2 equal to 2?
true
CALL printLesser()
false
is optionUpdate2 equal to 3?
true
CALL printWarehouseParts()
false
PRINT ('Wrong Typing')
is choice equal to 3?
true
PRINT(1. search parts record)
PRINT(2. search supplier detail)
PRINT(3. search parts that supplied by supplier)
PRINT('What you want to do?')
read optionUpdate3
is optionUpdate3 equal to 1?
true
CALL searchPart()
false
is optionUpdate3 equal to 2?
true
CALL searchSupplier()
false
is optionUpdate3 equal to 3?
true
CALL searchPartSupplied()
false
PRINT ('Wrong Typing')
is choice equal to 4?
true
PRINT('THANK YOU FOR USING SYSTEM')
BREAK LOOP
END FUNCTION

Program source code and explanation

Function inventory()

```
def inventory():
    inventoryLocal =[]
    for i in range(1):
        parts=[]
        partName=input('Enter name of part: ')
        parts.append(partName)
        partId=input('Enter Part ID:')
        parts.append(partId)
        partQuantity=input("Enter Quantity of part: ")
        parts.append(partQuantity)
        partSupplier=input("Enter the company of supplier : ")
        parts.append(partSupplier)
        inventoryLocal.append(parts)
    return inventoryLocal
```

Function inventory() is used to allow users to input the details of parts. All of the data will append into parts and after all data have store into parts then the parts will be appending into inventoryLocal.

Function supplier()

```
def supplier():
    supplierLocal = []
    for i in range(1):
        supplierParts=[]
        supplierName= input('Enter name of company of supplier: ')
        supplierParts.append(supplierName)
        supplierContect=input('Enter company\'s contact number: ')
        supplierParts.append(supplierContect)
        supplierSupplied=input("Enter part's Section code with warehouse code of supplier supplied:[ex:ESWBS]")
        supplierParts.append(supplierSupplied)
        supplierLocal.append(supplierParts)
    return supplierLocal
```

Function supplier() is used to allow users to input the data , append into supplierParts and append supplierParts into supplierLocal.

Function update(newData,fileHandler)

```
def update(newData, fileHandler):
    for data in newData:
        for item in data:
            fileHandler.write(item)
            fileHandler.write('\t')
        fileHandler.write('\n')
```

Function update(newData,fileHandler) is used to update the list. If there have element inside the data, the item with a tab will step by step write into the fileHandler. Then the system will write a new line before reading the next element of newData.

Function saveInventoryWBS(), saveInventoryWAY(), and saveInventoryWBR()

```
def saveInventoryWBS():
    print('-----Adding parts into WBS warehouse-----')
    sellectedSection=int(input('Add to which section?[1.Engine Section(ES), 2.Air-conditioning Section(AS), 3.Safety and Cockpit section(SC)]'))
    try:
        if sellectedSection == 1:
            fileHandlerWBS = open('ES-WBS.txt','a')
        elif sellectedSection == 2:
            fileHandlerWBS = open('AS-WBS.txt','a')
        elif sellectedSection == 3:
            fileHandlerWBS = open('SC-WBS.txt','a')
        inventoryWBS = inventory()
        updateWBS = update(inventoryWBS, fileHandlerWBS)
        fileHandlerWBS.close()
    except:
        print('Wrong Typing')


def saveInventoryWAY():
    print('-----Adding parts into WAY warehouse-----')
    sellectedSection=int(input('Add to which section?[1.Engine Section(ES), 2.Air-conditioning Section(AS), 3.Safety and Cockpit section(SC)]'))
    try:
        if sellectedSection == 1:
            fileHandlerWAY = open('ES-WAY.txt','a')
        elif sellectedSection == 2:
            fileHandlerWAY = open('AS-WAY.txt','a')
        elif sellectedSection == 3:
            fileHandlerWAY = open('SC-WAY.txt','a')
        wayInventory = inventory()
        updateWAY = update(wayInventory, fileHandlerWAY)
        fileHandlerWAY.close()
    except:
        print('Wrong Typing')

def saveInventoryWBR():
    print('-----Adding parts into WBR warehouse-----')
    sellectedSection=int(input('Add to which section?[1.Engine Section(ES), 2.Air-conditioning Section(AS), 3.Safety and Cockpit section(SC)]'))
    try:
        if sellectedSection == 1:
            fileHandlerWBR = open('ES-WBR.txt','a')
        elif sellectedSection == 2:
            fileHandlerWBR = open('AS-WBR.txt','a')
        elif sellectedSection == 3:
            fileHandlerWBR = open('SC-WBR.txt','a')
        wbrInventory = inventory()
        updateWBR = update(wbrInventory, fileHandlerWBR)
        fileHandlerWBR.close()
    except:
        print('Wrong Typing')
```

Three of the functions are almost same function, the system ask user to choose which section they need to upload in different warehouse. There are used for calling out the current files and call inventory () function with a variable to get the input from users. After that, the system stores the information through call the update (newData, fileHandler) function with the current variable of different txt file.

Function editQuantityWBS(), editQuantityWAY(), and editQuantityWBR()

```
def editQuantityWBS():
    print('Engine Section(ES), Air-conditioning Section(AS), Safety and Cockpit section(SC)')
    secCode= input('which sections:[Enter Code]')
    try:
        if secCode == 'ES':
            fileHandlerWBS = open('ES-WBS.txt','r')
            pardIdForSearch=input('Which part you need to edit?[Enter Part Id]')
            newQuantity=input('What is the new quantity of it')
            newData=[]
            data = fileHandlerWBS.readlines()
            for line in data:
                line = line.rstrip()
                newLine = line.split("\t")
                if newLine[1] == pardIdForSearch:
                    newLine[2] = newQuantity
                newData.append(newLine)
                print(newLine)
            fileHandlerWBS.close()

            fileHandlerWBS1 = open('ES-WBS.txt','w')
            for data in newData:
                for item in data:
                    fileHandlerWBS1.write(item)
                    fileHandlerWBS1.write('\t')
                fileHandlerWBS1.write('\n')
            fileHandlerWBS1.close()
        elif secCode == 'AS':
            fileHandlerWBS = open('AS-WBS.txt','r')
            pardIdForSearch=input('Which part you need to edit?[Enter Part Id]')
            newQuantity=input('What is the new quantity of it')
            newData=[]
            data = fileHandlerWBS.readlines()
            for line in data:
                line = line.rstrip()
                newLine = line.split("\t")
                if newLine[1] == pardIdForSearch:
                    newLine[2] = newQuantity
                newData.append(newLine)
                print(newLine)
            fileHandlerWBS.close()
            fileHandlerWBS1 = open('AS-WBS.txt','w')
            for data in newData:
                for item in data:
                    fileHandlerWBS1.write(item)
                    fileHandlerWBS1.write('\t')
                fileHandlerWBS1.write('\n')
            fileHandlerWBS1.close()
        elif secCode == 'SC':
            fileHandlerWBS = open('SC-WBS.txt','r')
            pardIdForSearch=input('Which part you need to edit?[Enter Part Id]')
            newQuantity=input('What is the new quantity of it')
            newData=[]
            data = fileHandlerWBS.readlines()
            for line in data:
                line = line.rstrip()
                newLine = line.split("\t")
                if newLine[1] == pardIdForSearch:
                    newLine[2] = newQuantity
                newData.append(newLine)
                print(newLine)
            fileHandlerWBS.close()

            fileHandlerWBS1 = open('SC-WBS.txt','w')
            for data in newData:
                for item in data:
                    fileHandlerWBS1.write(item)
                    fileHandlerWBS1.write('\t')
                fileHandlerWBS1.write('\n')
            fileHandlerWBS1.close()
    except:
        print('Wrong Typing')
```

```
def editQuantityWAY():
    print('Engine Section(ES), Air-conditioning Section(AS), Safety and Cockpit section(SC)')
    secCode= input('which sections:[Enter Code]')
    try:
        if secCode == 'ES':
            fileHandlerWAY = open('ES-WAY.txt','r')
            pardIdForSearch=input('Which part you need to edit?[Enter Part Id]')
            newQuantity=input('What is the new quantity of it')
            newData=[]
            data = fileHandlerWAY.readlines()
            for line in data:
                line = line.rstrip()
                newLine = line.split("\t")
                if newLine[1] == pardIdForSearch:
                    newLine[2] = newQuantity
                newData.append(newLine)
                print(newLine)
            fileHandlerWAY.close()

            fileHandlerWAY1 = open('ES-WAY.txt','w')
            for data in newData:
                for item in data:
                    fileHandlerWAY1.write(item)
                    fileHandlerWAY1.write('\t')
                fileHandlerWAY1.write('\n')
            fileHandlerWAY1.close()
        elif secCode == 'AS':
            fileHandlerWAY = open('AS-WAY.txt','r')
            pardIdForSearch=input('Which part you need to edit?[Enter Part Id]')
            newQuantity=input('What is the new quantity of it')
            newData=[]
            data = fileHandlerWAY.readlines()
            for line in data:
                line = line.rstrip()
                newLine = line.split("\t")
                if newLine[1] == pardIdForSearch:
                    newLine[2] = newQuantity
                newData.append(newLine)
                print(newLine)
            fileHandlerWAY.close()
            fileHandlerWAY1 = open('AS-WAY.txt','w')
            for data in newData:
                for item in data:
                    fileHandlerWAY1.write(item)
                    fileHandlerWAY1.write('\t')
                fileHandlerWAY1.write('\n')
            fileHandlerWAY1.close()
        elif secCode == 'SC':
            fileHandlerWAY = open('SC-WAY.txt','r')
            pardIdForSearch=input('Which part you need to edit?[Enter Part Id]')
            newQuantity=input('What is the new quantity of it')
            newData=[]
            data = fileHandlerWAY.readlines()
            for line in data:
                line = line.rstrip()
                newLine = line.split("\t")
                if newLine[1] == pardIdForSearch:
                    newLine[2] = newQuantity
                newData.append(newLine)
                print(newLine)
            fileHandlerWAY.close()

            fileHandlerWAY1 = open('SC-WAY.txt','w')
            for data in newData:
                for item in data:
                    fileHandlerWAY1.write(item)
                    fileHandlerWAY1.write('\t')
                fileHandlerWAY1.write('\n')
            fileHandlerWAY1.close()
    except:
        print('Wrong Typing')
```

```
def editQuantityWBR():
    print('Engine Section(ES), Air-conditioning Section(AS), Safety and Cockpit section(SC)')
    secCode= input('which sections:[Enter Code]')
    try:
        if secCode == 'ES':
            fileHandlerWBR = open('ES-WBR.txt','r')
            pardIdForSearch=input('Which part you need to edit?[Enter Part Id]')
            newQuantity=input('What is the new quantity of it')
            newData=[]
            data = fileHandlerWBR.readlines()
            for line in data:
                line = line.rstrip()
                newLine = line.split("\t")
                if newLine[1] == pardIdForSearch:
                    newLine[2] = newQuantity
                newData.append(newLine)
                print(newLine)
            fileHandlerWBR.close()

            fileHandlerWBR1 = open('ES-WBR.txt','w')
            for data in newData:
                for item in data:
                    fileHandlerWBR1.write(item)
                    fileHandlerWBR1.write('\t')
                fileHandlerWBR1.write('\n')
            fileHandlerWBR1.close()
        elif secCode == 'AS':
            fileHandlerWBR = open('AS-WBR.txt','r')
            pardIdForSearch=input('Which part you need to edit?[Enter Part Id]')
            newQuantity=input('What is the new quantity of it')
            newData=[]
            data = fileHandlerWBR.readlines()
            for line in data:
                line = line.rstrip()
                newLine = line.split("\t")
                if newLine[1] == pardIdForSearch:
                    newLine[2] = newQuantity
                newData.append(newLine)
                print(newLine)
            fileHandlerWBR.close()
            fileHandlerWBR1 = open('AS-WBR.txt','w')
            for data in newData:
                for item in data:
                    fileHandlerWBR1.write(item)
                    fileHandlerWBR1.write('\t')
                fileHandlerWBR1.write('\n')
            fileHandlerWBR1.close()
        elif secCode == 'SC':
            fileHandlerWBR = open('SC-WBR.txt','r')
            pardIdForSearch=input('Which part you need to edit?[Enter Part Id]')
            newQuantity=input('What is the new quantity of it')
            newData=[]
            data = fileHandlerWBR.readlines()
            for line in data:
                line = line.rstrip()
                newLine = line.split("\t")
                if newLine[1] == pardIdForSearch:
                    newLine[2] = newQuantity
                newData.append(newLine)
                print(newLine)
            fileHandlerWBR.close()

            fileHandlerWBR1 = open('SC-WBR.txt','w')
            for data in newData:
                for item in data:
                    fileHandlerWBR1.write(item)
                    fileHandlerWBR1.write('\t')
                fileHandlerWBR1.write('\n')
            fileHandlerWBR1.close()
    except:
        print('Wrong Typing')
```

Three of the functions are almost same function, there are used for calling out the current files and ask the user to input the part's id as pardIdForSearch and the new quantity as newQuantity that they want to change. If the system found the partIdForSearch is inside the file, the quantity of the part will be replaced to newQuantity. Then system will call the current file again and update the newQuantity into the file

Function updateInventory()

```
def updateInventory():
    print('WBS warehouse,WAY warehouse, WBR warehouse')
    warCode = input('Which warehouse : [Enter code in Capital Letter]')
    try:
        if warCode == 'WBS':
            service=int(input('Which service you want:\n1.Add part\n2.Edit stock of parts'))
            try:
                if service == 1:
                    content = saveInventoryWBS()
                elif service == 2:
                    content = editQuantityWBS()
            except:
                print('Wrong Typing')
        if warCode == 'WAY':
            service=int(input('Which service you want:\n1.Add part\n2.Edit stock of parts'))
            try:
                if service == 1:
                    content = saveInventoryWAY()
                elif service == 2:
                    content = editQuantityWAY()
            except:
                print('Wrong Typing')
        if warCode == 'WBR':
            service=int(input('Which service you want:\n1.Add part\n2.Edit stock of parts'))
            try:
                if service == 1:
                    content = saveInventoryWBR()
                elif service == 2:
                    content = editQuantityWBR()
            except:
                print('Wrong Typing')
    except:
        print('Wrong Typing')
```

This function is used to create parts inventory, update Inventory's quantity. The user can choose the part's warehouse that they want to edit before running through the function of save inventory or edit quantity.

Function saveSupplier()

```
def saveSupplier():
    fileHandlerSup = open('supplierDetails.txt','a')
    print('-----Adding supplier details-----')
    supplierDetails = supplier()
    updateSup = update(supplierDetails , fileHandlerSup)
    fileHandlerSup.close()
```

This function is used to append the details of supplier that been store at the supplierDetails into the txt file named supplierDetails

Function printWhole()

```
def printWhole():
    try:
        fileHandlerESWBS = open('ES-WBS.txt','r')
        fileHandlerASWBS = open('AS-WBS.txt','r')
        fileHandlerSCWBS = open('SC-WBS.txt','r')
        fileHandlerESWAY = open('ES-WAY.txt','r')
        fileHandlerASWAY = open('AS-WAY.txt','r')
        fileHandlerSCWAY = open('SC-WAY.txt','r')
        fileHandlerESWBR = open('ES-WBR.txt','r')
        fileHandlerASWBR = open('AS-WBR.txt','r')
        fileHandlerSCWBR = open('SC-WBR.txt','r')
    except:
        print('cannot read')
    print('Name"\t"ID"\t"Quantity"\t"Supplier')
    for lineWBS in fileHandlerESWBS:
        print(lineWBS)
    for lineWBS in fileHandlerASWBS:
        print(lineWBS)
    for lineWBS in fileHandlerSCWBS:
        print(lineWBS)
    for lineWAY in fileHandlerESWAY:
        print(lineWAY)
    for lineWAY in fileHandlerASWAY:
        print(lineWAY)
    for lineWAY in fileHandlerSCWAY:
        print(lineWAY)
    for lineWBR in fileHandlerESWBR:
        print(lineWBR)
    for lineWBR in fileHandlerASWBR:
        print(lineWBR)
    for lineWBR in fileHandlerSCWBR:
        print(lineWBR)
    fileHandlerESWBS.close()
    fileHandlerASWBS.close()
    fileHandlerSCWBS.close()
    fileHandlerESWAY.close()
    fileHandlerASWAY.close()
    fileHandlerSCWAY.close()
    fileHandlerESWBR.close()
    fileHandlerASWBR.close()
    fileHandlerSCWBR.close()
```

This function is use for printing all the parts that stored inside the company. The system needs to open all txt files, read and print line by line of them.

Function printLesser()

```
def printLesser():
    try:
        fileHandlerESWBS = open('ES-WBS.txt','r')
        fileHandlerASWBS = open('AS-WBS.txt','r')
        fileHandlerSCWBS = open('SC-WBS.txt','r')
        fileHandlerESWAY = open('ES-WAY.txt','r')
        fileHandlerASWAY = open('AS-WAY.txt','r')
        fileHandlerSCWAY = open('SC-WAY.txt','r')
        fileHandlerESWBR = open('ES-WBR.txt','r')
        fileHandlerASWBR = open('AS-WBR.txt','r')
        fileHandlerSCWBR = open('SC-WBR.txt','r')
        lesser = ['0','1','2','3','4','5','6','7','8','9']
    except:
        print('read file error')

    print('Parts that quantity less then 10')
    print('Name"\t"ID"\t"Quantity"\t"Supplier')
    try:
        for lineWBS in fileHandlerESWBS:
            lineWBS = lineWBS.rstrip()
            newLineWBS = lineWBS.split("\t")
            if not newLineWBS[2] in lesser:
                continue
            print(lineWBS)
        for lineWBS in fileHandlerASWBS:
            lineWBS = lineWBS.rstrip()
            newLineWBS = lineWBS.split("\t")
            if not newLineWBS[2] in lesser:
                continue
            print(lineWBS)
        for lineWBS in fileHandlerSCWBS:
            lineWBS = lineWBS.rstrip()
            newLineWBS = lineWBS.split("\t")
            if not newLineWBS[2] in lesser:
                continue
            print(lineWBS)
        for lineWAY in fileHandlerESWAY:
            lineWAY = lineWAY.rstrip()
            newLineWAY = lineWAY.split("\t")
            if not newLineWAY[2] in lesser:
                continue
            print(lineWAY)
        for lineWAY in fileHandlerASWAY:
            lineWAY = lineWAY.rstrip()
            newLineWAY = lineWAY.split("\t")
            if not newLineWAY[2] in lesser:
                continue
            print(lineWAY)
        for lineWAY in fileHandlerSCWAY:
            lineWAY = lineWAY.rstrip()
            newLineWAY = lineWAY.split("\t")
            if not newLineWAY[2] in lesser:
                continue
            print(lineWAY)
        for lineWBR in fileHandlerESWBR:
            lineWBR = lineWBR.rstrip()
            newLineWBR = lineWBR.split("\t")
            if not newLineWBR[2] in lesser:
                continue
            print(lineWBR)
        for lineWBR in fileHandlerASWBR:
            lineWBR = lineWBR.rstrip()
            newLineWBR = lineWBR.split("\t")
            if not newLineWBR[2] in lesser:
                continue
            print(lineWBR)
        for lineWBR in fileHandlerSCWBR:
            lineWBR = lineWBR.rstrip()
            newLineWBR = lineWBR.split("\t")
            if not newLineWBR[2] in lesser:
                continue
            print(lineWBR)
    except:
        print('print error')
    fileHandlerESWBS.close()
    fileHandlerASWBS.close()
    fileHandlerSCWBS.close()
    fileHandlerESWAY.close()
    fileHandlerASWAY.close()
    fileHandlerSCWAY.close()
    fileHandlerESWBR.close()
    fileHandlerASWBR.close()
    fileHandlerSCWBR.close()
```

This function is use for printing all the parts which is almost out of stock. If the third element of the lines are not equal to the element of less then it will been skip, the other lines will been print.

Function printWarehouseParts ()

```
def printWarehouseParts():
    warehouseChoosed=int(input('Enter number of warehouse you want to print(1.WBS,2.WAY,3.WBR)'))
    try:
        print('Name''\t''ID''\t''Quantity''\t''Supplier')
        if warehouseChoosed == 1:
            fileHandlerESWBS = open('ES-WBS.txt','r')
            fileHandlerASWBS = open('AS-WBS.txt','r')
            fileHandlerSCWBS = open('SC-WBS.txt','r')
            print('the parts of engine section(ES)')
            for lineWBS in fileHandlerESWBS:
                print(lineWBS)
            print('the parts of air-conditioning section(AS)')
            for lineWBS in fileHandlerASWBS:
                print(lineWBS)
            print('the parts of safety and cockpit section(SC)')
            for lineWBS in fileHandlerSCWBS:
                print(lineWBS)
            fileHandlerESWBS.close()
            fileHandlerASWBS.close()
            fileHandlerSCWBS.close()
        elif warehouseChoosed == 2:
            fileHandlerESWAY = open('ES-WAY.txt','r')
            fileHandlerASWAY = open('AS-WAY.txt','r')
            fileHandlerSCWAY = open('SC-WAY.txt','r')
            print('the parts of engine section(ES)')
            for lineWAY in fileHandlerESWAY:
                print(lineWAY)
            print('the parts of air-conditioning section(AS)')
            for lineWAY in fileHandlerASWAY:
                print(lineWAY)
            print('the parts of safety and cockpit section(SC)')
            for lineWAY in fileHandlerSCWAY:
                print(lineWAY)
            fileHandlerESWAY.close()
            fileHandlerASWAY.close()
            fileHandlerSCWAY.close()
        elif warehouseChoosed == 3:
            fileHandlerESWBR = open('ES-WBR.txt','r')
            fileHandlerASWBR = open('AS-WBR.txt','r')
            fileHandlerSCWBR = open('SC-WBR.txt','r')
            print('the parts of engine section(ES)')
            for lineWBR in fileHandlerESWBR:
                print(lineWBR)
            print('the parts of air-conditioning section(AS)')
            for lineWBR in fileHandlerASWBR:
                print(lineWBR)
            for lineWBR in fileHandlerSCWBR:
                print(lineWBR)
            print('the parts of safety and cockpit section(SC)')
            fileHandlerESWBR.close()
            fileHandlerASWBR.close()
            fileHandlerSCWBR.close()
    except:
        print('Wrong Typing')
```

This function is used for printing all the details of parts inside the warehouse which selected by user according to the different sections. It will clearly be shown the parts according their sections.

Function printWarehouseParts ()

```
def searchParts():
    fileHandlerESWBS = open('ES-WBS.txt','r')
    fileHandlerASWBS = open('AS-WBS.txt','r')
    fileHandlerSCWBS = open('SC-WBS.txt','r')
    fileHandlerESWAY = open('ES-WAY.txt','r')
    fileHandlerASWAY = open('AS-WAY.txt','r')
    fileHandlerSCWAY = open('SC-WAY.txt','r')
    fileHandlerESWBR = open('ES-WBR.txt','r')
    fileHandlerASWBR = open('AS-WBR.txt','r')
    fileHandlerSCWBR = open('SC-WBR.txt','r')
    keyword=input('Enter Part Id you want to search')
    print('Name"\t"ID"\t"Quantity"\t"Supplier')
    try:
        for lineWBS in fileHandlerESWBS:
            lineWBS = lineWBS.rstrip()
            newLineWBS = lineWBS.split("\t")
            if newLineWBS[1] == keyword:
                print(newLineWBS)
        for lineWBS in fileHandlerASWBS:
            lineWBS = lineWBS.rstrip()
            newLineWBS = lineWBS.split("\t")
            if newLineWBS[1] == keyword:
                print(newLineWBS)
        for lineWBS in fileHandlerSCWBS:
            lineWBS = lineWBS.rstrip()
            newLineWBS = lineWBS.split("\t")
            if newLineWBS[1] == keyword:
                print(newLineWBS)
        for lineWAY in fileHandlerESWAY:
            lineWAY = lineWAY.rstrip()
            newLineWAY = lineWAY.split("\t")
            if newLineWAY[1] == keyword:
                print(newLineWAY)
        for lineWAY in fileHandlerASWAY:
            lineWAY = lineWAY.rstrip()
            newLineWAY = lineWAY.split("\t")
            if newLineWAY[1] == keyword:
                print(newLineWAY)
        for lineWAY in fileHandlerSCWAY:
            lineWAY = lineWAY.rstrip()
            newLineWAY = lineWAY.split("\t")
            if newLineWAY[1] == keyword:
                print(newLineWAY)
        for lineWBR in fileHandlerESWBR:
            lineWBR = lineWBR.rstrip()
            newLineWBR = lineWBR.split("\t")
            if newLineWBR[1] == keyword:
                print(newLineWBR)
        for lineWBR in fileHandlerASWBR:
            lineWBR = lineWBR.rstrip()
            newLineWBR = lineWBR.split("\t")
            if newLineWBR[1] == keyword:
                print(newLineWBR)
        for lineWBR in fileHandlerSCWBR:
            lineWBR = lineWBR.rstrip()
            newLineWBR = lineWBR.split("\t")
            if newLineWBR[1] == keyword:
                print(newLineWBR)
    except:
        print('Searching error')
    fileHandlerESWBS.close()
    fileHandlerASWBS.close()
    fileHandlerSCWBS.close()
    fileHandlerESWAY.close()
    fileHandlerASWAY.close()
    fileHandlerSCWAY.close()
    fileHandlerESWBR.close()
    fileHandlerASWBR.close()
    fileHandlerSCWBR.close()
```

This function will require users to input the part id as keyword. It will be looking towards all the lines of files. If the second element of the lines is equal to the keyword then it will been print, the other lines will been skip.

Function searchSupplier ()

```
def searchSupplier():
    try:
        supplierInfo = open('supplierDetails.txt','r')
    except:
        print('Error while open file')
    keywordOfSup = input('Enter the part that it supplied to search the company of supplier:[ex:ESWBS]')
    for lineSup in supplierInfo:
        lineSup = lineSup.rstrip()
        newLineSup = lineSup.split('\t')
        if not keywordOfSup in newLineSup:
            continue
        print(lineSup)
    supplierInfo.close()
```

This function will open the supplierDetails.txt file and ask users to input keywordOfSup for search the suplier details. If there are not the keywordOfSup inside the line then it will be skip.

Function searchPartSupplied ()

```
def searchPartSupplied():
    try:
        supplierInfo = open('supplierDetails.txt','r')
    except:
        print('Error while open file')
    supName= input('Enter suppier\'s company name who want to search for:')
    for lineSup in supplierInfo:
        lineSup = lineSup.rstrip()
        newLineSup = lineSup.split('\t')
        if not supName in newLineSup:
            continue
        print(lineSup)
    supplierInfo.close()
```

This function will open the supplierDetails.txt file and ask users to input supName for search the parts that suplier supplied. If there are not the supName inside the line then it will be skip.

Function menu ()

```
def menu():
    print('Welcome to use Automobile Parts Inventory Management System')
    while True:
        choice=int(input('What you want to do?\n1.Update parts inventory\n2.Parts Inventory Tracking\n3.Search\n4.Exit'))
        try:
            if choice == 1:
                optionUpdate=int(input('What you want to do?\n1.update inventory\n2.update Supplier'))
                if optionUpdate == 1:
                    updateInventory()
                elif optionUpdate == 2:
                    saveSupplier()
                else:
                    print('wrong typing')
            elif choice == 2:
                optionUpdate2=int(input('What you want to do?\n1.print whole availabile quantity\n2.print all part that quantity less then 10 units\n3.print the whole parts of sections in warehouse'))
                if optionUpdate2 == 1:
                    printWhole()
                elif optionUpdate2 == 2:
                    printLesser()
                elif optionUpdate2 == 3:
                    printWarehouseParts()
                else:
                    print('wrong typing')
            elif choice == 3:
                optionUpdate3=int(input('What you want to do?\n1.search parts record\n2.search supplier detail\n3.search parts that supplied by supplier'))
                if optionUpdate3 == 1:
                    searchParts()
                elif optionUpdate3 == 2:
                    searchSupplier()
                elif optionUpdate3 == 3:
                    searchPartSupplied()
                else:
                    print('wrong typing')
            elif choice == 4:
                print('THANK YOU FOR USING SYSTEM!')
                break
        except:
            print('wrong typing')
```

It is the main function of this system to run all the other function. Since it has the while loop on the top, then the function will not stop repeating until break the loop.

Screenshots of sample input/ output and explanation

The input/ output of the program:

```
Python 3.7.7 (tags/v3.7.7:d7c567b08f, Mar 10 2020, 10:41:24) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\Owner\Documents\assignment\FSD assignment\SIM_YOKE_SHIN_TP059851.py
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit
```

This screenshot is shown the menu() function that show all three update, print,and searching function as options to choose by the users.Also, users can also enter ‘4’ to exit the system.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit1
What you want to do?
1.update inventory
2.update Supplier
```

This screenshot is the resalt of type ‘1’. It means the user can enter either ‘1’ or ‘2’ for update the record of inventories or update the record of suppliers.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit1
What you want to do?
1.update inventory
2.update Supplier1
WBS warehouse,WAY warehouse, WBR warehouse
Which warehouse : [Enter code in Capital Letter]
```

It is the screenshot of the updateInventory() function while the user has choose option ‘1’ to update the record of inventory. Users must enter the code of warehouse to update the record of the warehouse.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit1
What you want to do?
1.update inventory
2.update Supplier1
WBS warehouse,WAY warehouse, WBR warehouse
Which warehouse : [Enter code in Capital Letter]WBS
Which service you want:
1.Add part
2.Edit stock of parts
```

It is the screenshot of the output of updateInventory() function while the user has enter 'WBS' that can allow users type '1' or '2' to add new part's detail or edit the quantity of parts.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit1
What you want to do?
1.update inventory
2.update Supplier1
WBS warehouse,WAY warehouse, WBR warehouse
Which warehouse : [Enter code in Capital Letter]WBS
Which service you want:
1.Add part
2.Edit stock of parts1
-----Adding parts into WBS warehouse-----
Add to which section?[1.Engine Section(ES), 2.Air-conditioning Section(AS), 3.Safety and Cockpit section(SC)]
```

It is the screenshot of the output of updateInventory() function while the user has enter '1' to add new part and shown 3 names of section to allow user to choose.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit1
What you want to do?
1.update inventory
2.update Supplier1
WBS warehouse,WAY warehouse, WBR warehouse
Which warehouse : [Enter code in Capital Letter]WBS
Which service you want:
1.Add part
2.Edit stock of parts2
Engine Section(ES), Air-conditioning Section(AS), Safety and Cockpit section(SC)
which sections:[Enter Code]
```

It is the screenshot of the output of updateInventory() function while the user has entered '2' to edit the quantity of part and shown 3 names of section to allow user to choose the section's.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit1
What you want to do?
1.update inventory
2.update Supplier1
WBS warehouse,WAY warehouse, WBR warehouse
Which warehouse：[Enter code in Capital Letter]WBS
Which service you want:
1.Add part
2.Edit stock of parts1
-----Adding parts into WBS warehouse-----
Add to which section?[1.Engine Section(ES), 2.Air-conditioning Section(AS), 3.Safety and Cockpit section(SC)]1
Enter name of part:
```

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit1
What you want to do?
1.update inventory
2.update Supplier1
WBS warehouse,WAY warehouse, WBR warehouse
Which warehouse：[Enter code in Capital Letter]WBS
Which service you want:
1.Add part
2.Edit stock of parts1
-----Adding parts into WBS warehouse-----
Add to which section?[1.Engine Section(ES), 2.Air-conditioning Section(AS), 3.Safety and Cockpit section(SC)]1
Enter name of part: engineWBS01
Enter Part ID:
```

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit1
What you want to do?
1.update inventory
2.update Supplier1
WBS warehouse,WAY warehouse, WBR warehouse
Which warehouse：[Enter code in Capital Letter]WBS
Which service you want:
1.Add part
2.Edit stock of parts1
-----Adding parts into WBS warehouse-----
Add to which section?[1.Engine Section(ES), 2.Air-conditioning Section(AS), 3.Safety and Cockpit section(SC)]1
Enter name of part: engineWBS01
Enter Part ID:ESWBS01
Enter Quantity of part:
```

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit1
What you want to do?
1.update inventory
2.update Supplier1
WBS warehouse,WAY warehouse, WBR warehouse
Which warehouse：[Enter code in Capital Letter]WBS
Which service you want:
1.Add part
2.Edit stock of parts1
-----Adding parts into WBS warehouse-----
Add to which section?[1.Engine Section(ES), 2.Air-conditioning Section(AS), 3.Safety and Cockpit section(SC)]1
Enter name of part: engineWBS01
Enter Part ID:ESWBS01
Enter Quantity of part: 7
Enter the company of supplier :
```

There are the screenshot of inventory() function that let user input the detail of new part's name , id , quantity, and supplier's companny name.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit1
What you want to do?
1.update inventory
2.update Supplier1
WBS warehouse,WAY warehouse, WBR warehouse
Which warehouse : [Enter code in Capital Letter]WBS
Which service you want:
1.Add part
2.Edit stock of parts1
-----Adding parts into WBS warehouse-----
Add to which section?[1.Engine Section(ES), 2.Air-conditioning Section(AS), 3.Safety and Cockpit section(SC)]1
Enter name of part: engineWBS01
Enter Part ID:ESWBS01
Enter Quantity of part: 7
Enter the company of supplier : Aisin Seiki Corporation
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit
```

After finish input all the details of part will return into the main menu.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit1
What you want to do?
1.update inventory
2.update Supplier2
-----Adding supplier details-----
Enter name of company of supplier:
```

It is the screenshot of the output of saveSupplier() function while the user has choose option '2' to update the record of supplier.It require user to input the name of the supplier's company.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit1
What you want to do?
1.update inventory
2.update Supplier2
-----Adding supplier details-----
Enter name of company of supplier: Robert Bosch GmbH
Enter company's contact number:
```

It is the screenshot of the saveSupplier() function that let user input the contact number of new suppliers.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit1
What you want to do?
1.update inventory
2.update Supplier2
-----Adding supplier details-----
Enter name of company of supplier: Robert Bosch GmbH
Enter company's contact number: 07-2336060
Enter part's Section code with warehouse code of supplier supplied:[ex:ESWBS]
```

It is the screenshot of the saveSupplier() function that let user input the code of section code with warehouse code likes ESWBS that the supplier supplied.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit1
What you want to do?
1.update inventory
2.update Supplier2
-----Adding supplier details-----
Enter name of company of supplier: Robert Bosch GmbH
Enter company's contact number: 07-2336060
Enter part's Section code with warehouse code of supplier supplied:[ex:ESWBS]SCWAY
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit
```

After finish input all the details of supplier will return into the main menu.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit2
What you want to do?
1.print whole availabile quantity
2.print all part that quantity less then 10 units
3.print the whole parts in each warehouse
```

This screenshot is the result of type ‘2’. It can allow user to choose print whole available quantity, print all parts that is almost out of stock, or print the parts according with the option of warehouse.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit2
What you want to do?
1.print whole availabile quantity
2.print all part that quantity less then 10 units
3.print the whole parts of sections in warehouse1
Name     ID          Quantity  Supplier
engineWBS01      ESWBS01          7          Aisin Seiki Corporation

engineWBS02      ESWBS02          13         Aisin Seiki Corporation

Air-conditioningWBS01        ASWBS01          9          Denso Corporation

SafetyWBS01      SCWBS01          12         ZF Friedrichshafen AG

engineWAY01      ESWAY01          6          Aisin Seiki Corporation

air-conditioningWAY01        ASWAY01          4          Denso Corporation

SafetyWAY01      SCWAY01          17         Robert Bosch GmbH

engineWBR01      ESWBR01          11         Aisin Seiki Corporation

air-conditioningWBR01        ASWBR01          4          Denso Corporation

safetyWBR01      SCWBR01          3          Robert Bosch GmbH

What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit
```

This screenshot is the result of type '1' and active the printWhole() function. It has print all the parts detail that been store in all txt file.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit2
What you want to do?
1.print whole availabile quantity
2.print all part that quantity less then 10 units
3.print the whole parts of sections in warehouse2
Parts that quantity less then 10
Name     ID          Quantity  Supplier
engineWBS01      ESWBS01          7          Aisin Seiki Corporation
Air-conditioningWBS01        ASWBS01          9          Denso Corporation
engineWAY01      ESWAY01          6          Aisin Seiki Corporation
air-conditioningWAY01        ASWAY01          4          Denso Corporation
air-conditioningWBR01        ASWBR01          4          Denso Corporation
safetyWBR01      SCWBR01          3          Robert Bosch GmbH
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit
```

This screenshot is the result of type '2' and active the printLesser() function. It has print all the part's details which its quantity is less then 10 units.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit2
What you want to do?
1.print whole availabile quantity
2.print all part that quantity less then 10 units
3.print the whole parts of sections in warehouse3
Enter number of warehouse you want to print(1.WBS,2.WAY,3.WBR)
```

This screenshot is the result of type '3' and allow user to choose which warehouse's parts they want to print.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit2
What you want to do?
1.print whole availabile quantity
2.print all part that quantity less then 10 units
3.print the whole parts of sections in warehouse3
Enter number of warehouse you want to print(1.WBS,2.WAY,3.WBR)1
Name     ID          Quantity  Supplier
the parts of engine section(ES)
engineWBS01      ESWBS01          7         Aisin Seiki Corporation

engineWBS02      ESWBS02          13        Aisin Seiki Corporation

the parts of air-conditioning section(AS)
Air-conditioningWBS01       ASWBS01          9         Denso Corporation

the parts of safety and cockpit section(SC)
SafetyWBS01      SCWBS01          12        ZF Friedrichshafen AG

What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit
```

This screenshot is the result of type '1'. It has print all the part's details which belong to WBS warehouse.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit2
What you want to do?
1.print whole availabile quantity
2.print all part that quantity less then 10 units
3.print the whole parts of sections in warehouse3
Enter number of warehouse you want to print(1.WBS,2.WAY,3.WBR)2
Name     ID          Quantity  Supplier
the parts of engine section(ES)
engineWAY01      ESWAY01          6         Aisin Seiki Corporation

the parts of air-conditioning section(AS)
air-conditioningWAY01       ASWAY01          4         Denso Corporation

the parts of safety and cockpit section(SC)
SafetyWAY01      SCWAY01          17        Robert Bosch GmbH

What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit
```

This screenshot is the result of type '2'. It has print all the part's details which belong to WAY warehouse.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit2
What you want to do?
1.print whole availabile quantity
2.print all part that quantity less then 10 units
3.print the whole parts of sections in warehouse3
Enter number of warehouse you want to print(1.WBS,2.WAY,3.WBR)3
Name      ID          Quantity  Supplier
the parts of engine section(ES)
engineWBR01       ESWBR01          11          Aisin Seiki Corporation

the parts of air-conditioning section(AS)
air-conditioningWBR01        ASWBR01          4          Denso Corporation

safetyWBR01       SCWBR01          3          Robert Bosch GmbH

the parts of safety and cockpit section(SC)
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit
```

This screenshot is the result of type '3'. It has print all the part's details which belong to WBR warehouse and return to the main menu.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit3
What you want to do?
1.search parts record
2.search supplier detail
3.search parts that supplied by supplier
```

This screenshot is the result of type '3'. It allow user to search the details of part, search the details of supplier, or search the parts that the supplier supplied.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit3
What you want to do?
1.search parts record
2.search supplier detail
3.search parts that supplied by supplier1
Enter Part Id you want to search
```

This screenshot is the result of type '1'. It require user to enter part id for search the detail of part.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit3
What you want to do?
1.search parts record
2.search supplier detail
3.search parts that supplied by supplier1
Enter Part Id you want to searchESWBS01
Name      ID          Quantity  Supplier
['engineWBS01', 'ESWBS01', '7', 'Aisin Seiki Corporation']
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit
```

The screenshot is shown the result after enter the part id, it successfully print the details of part.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit3
What you want to do?
1.search parts record
2.search supplier detail
3.search parts that supplied by supplier2
Enter the part that it supplied to search the company of supplier:[ex:ESWBS]
```

This screenshot is the result of type ‘2’. It require user to enter the part id with the warehouse code for search the detail of supplier.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit3
What you want to do?
1.search parts record
2.search supplier detail
3.search parts that supplied by supplier2
Enter the part that it supplied to search the company of supplier:[ex:ESWBS]ESWAY
Aisin Seiki Corporation         07-1234567          ESWAY
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit
```

This screenshot is shown the result after the user type the special code of part’s section and warehouse. It has print the details of the supplier which the user want to search.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit3
What you want to do?
1.search parts record
2.search supplier detail
3.search parts that supplied by supplier3
Enter suppier's company name who want to search for:
```

This screenshot is the result of type '3'. It require user to input the suppplier's company name to search the detail of the supplier.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit3
What you want to do?
1.search parts record
2.search supplier detail
3.search parts that supplied by supplier3
Enter suppier's company name who want to search for:Aisin Seiki Corporation
Aisin Seiki Corporation         07-1234567          ESWBR
Aisin Seiki Corporation         07-1234567          ESWAY
Aisin Seiki Corporation         07-1234567          ESWBS
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit
```

This screenshot is the result of enter the name of the supplier's company. It has print all the part's section that the supplier supplied.

```
Welcome to use Automobile Parts Inventory Management System
What you want to do?
1.Update parts inventory
2.Parts Inventory Tracking
3.Search
4.Exit4
THANK YOU FOR USING SYSTEM!
>>>
```

This screenshot is the resalt of type '4'. It means the user want to exit the system and it will break the loop of repeat running the system.

Conclution

In conclution, the proposed flowchart , pseudocode and code met all the requirement and fulfilled the made assumptions. The purpose of the research were reached at the end. The researcher had added a large amount of knowledge regarding coding. This report had well documented the designs of the proram.

Reference