#include <stdio.h>

#include <stdlib.h>

//code all necessary include's

//structures needed

struct customer

{

};

struct tool

{

};

struct transaction

{

};

//add alll required prototype

//organize them by section: customer, tool, transaction

//define any needed var

int main()

{ //declarae the necessary arrays of objects

//==>array of 100 customers

//==>array of 100 tools

//==>array of 100 transactions

//define any working variables

//start a main loop

//call main menu

//switch on Moption

//for every case, call related menu and switch on that option:

//embed another loop for each of the menus so you can stay with a menu until user select to go back

while(Moption != 4)

{

Moption = mainMenu();

switch(mainOption)

{

case 1: while (Coption != 3)

{

switch(coption)

{

//3 cases

}//end switch coptiuon

}//end while coption

case 2: while(Toption != 3)

{

Toption = ToolMenu()

switch(Toption)

{

//3 cases

}//end switch toption

}//end while toption

case 3: while(TransacOPtion != 4)

{

TransacOption = TransactionMenu();

switch(TranscOption)

{

//4 cases

}//end TransacOption

}//end while TransacOPtion

}//end mainoption

}//while mainoption

//program is done

return 0;

}//end main()

//========================necessary menus==============

???? mainMenu()

{

//display the required options

//get option

//return option assume it is called Moption

}//end mainMenu

//======================================================

???? CustMenu()

{

//display the required options

//get option

//return option assume it is called Coption

}//end custmenu

//======================================================

???? ToolMenu()

{

//display the required options

//get option

//return option assume it is called Toption

}//end ToolMenu

//=======================================================

???? TransactionMenu()

{

//display the required options

//get option

//return option assume it is called Transoption

}//end TransactionMenu

//========================================================

//========================function that process a customer=========

????getCustomer(????)

{

//input attributes

//validate each input using a validation function coded later

}//end getCustomer

//=======================================================

????dispCustomer(????)

{

//display a customer attributes in a readable friendly format

}//end dispCustomer

//=======================================================

????searchCustomer(????)

{

//search for a customer ID

//retunr -1 if not found or position where it was found

}//end search Customer

//========================================================

????validateCustomerID(????)

{

//validate

//return true or false

}//end validateID

//=========================================================

????validateName(????)

{

//validate for all letters either first or last

//return true/false

}//end validateName

//==========================================================

????validatePhone(????)

{

//validate

//return true or false

}//end validatePhone

//===========================================================

????validateEmail(????)

{

//make sure of the one @ presence, its position

//assume the only valid extension is the .com

//return TRue/False

}//end validateEmail

//add any other function needed for4 customers here

//===================================================functions to [process a tool

????getTool(????)

{

//get the attributes

//validate each using funttion to be coded next

}//end getTool

//===================================================

????dispTool(????)

{

//display all attributes of a tool

//in a good friendly format

}//end dispTool

//===================================================

????searchTool(????)

{

//search and if found return position otherwise return -1

}//end searchTool

//====================================================

????validateToolID(????)

{

//validate for all allowed char and length

//return true or false

}//end valdiateToolID

//====================================================

????validateToolType(?????)

{

//validate for spaces and letters

}//end validateToolType

//====================================================

????validateRentalFees(????)

{

//validate a real number period and two decimals

//return true/false

}//end validateRentalFees

//=====================================================

//add any other function to may need for tools

//==================================================functions that process transactions

????InitiateTransaction(?????)

{

//in main , prompt user for Cust ID, search for it and if found:

//prompt user for Tool ID and if found and NOT rented then proceed to ask for rental term

//if full day apply correct fees and store 'F' under rental term if half day apply fees and store 'H' for half day

//then store all values / record in the transaction using current pointer location ...passed to this function

}//end InitiateTransaction

//=====================================================

????dispTransaction(?????????????)

{

//if a transaction is found via the related search

//then display everything about the customer and the tool and the transaction

}//end dispTransaction

//======================================================

????SearchTransaction(?????)

{

//search by Cust ID or Tool ID ...thsi information should be provided to this search

//or you may want to code two different fucntions ...one to search by Cust ID and another one to search by Tool ID

//function return true/false

}//end SearchTransaction

//======================================================

????reportAllTransactions(??????????????)

{

//display all: all transaction

//may not be necessary as you can recycle the dispTransaction() and embed the call within a loop from main

//it is up to you

}//end reportAllTransaction

//add any other function that deals with the transaction

//================================================================================================================================