|  |
| --- |
| Homework : \_\_7A\_\_\_\_  Source Code  //Christopher Badolato 3064088  //Assignment 7  //ENC 3211 0003  /\*  Will take user entered data and store it on a data file. Will then write the data  on to a hardware.txt file.  \*/  #include <stdio.h>  #include <string.h>  struct hardwareInventory{  int recordNumber;  char toolName[20];  char toolQuantity[6];  double toolCost;  };  //function prototypes.  int inputChoice(void);  int modifyChoice(void);  void createRecord(FILE \*createdfilePointer);  void addRecord(FILE \*filePointer);  void listRecord(FILE \*readPointer);  void deleteRecord(FILE \*filePointer);  void updateRecord(FILE \*filePointer);  void displayData(FILE \*filePointer);  int main(){  FILE \*createdFilePointer;  int userChoice;  //create a new empty data file pointer  //we are going to send that file pointer around to the  //corresponding function  while((userChoice = inputChoice() ) != 9){  if((createdFilePointer = fopen("hardware.dat","rb+")) == NULL){  if((createdFilePointer = fopen("hardware.dat","wb")) == NULL){  printf("File unable to be opened\n");  }  }  switch (userChoice){  //creates our hardware.txt file from our dat file.  case 1:  listRecord(createdFilePointer);  printf("The hardware.txt file was successfully written.\n");  break;  //update an old record  case 2:  updateRecord(createdFilePointer);  break;  //add a new record  case 3:  addRecord(createdFilePointer);  break;  //delete a record  case 4:  deleteRecord(createdFilePointer);  break;  //display a record  case 5:  displayData(createdFilePointer);  break;  default:  printf("Nothing here!\n");  break;  }  fclose(createdFilePointer);  }  return 0;  }  //just our main menus input screen. returns the choice of our users  int inputChoice(){  int userChoice;  printf( "\nYour choices are:\n" );  printf("1 - STORE a formatted text file of inventoried tools called hardware.txt for printing\n");  printf("2 - UPDATE a tool Record\n");  printf("3 - ADD a new tool record\n");  printf("4 - DELETE a tool record\n");  printf("5 - DISPLAY a tool record\n");  printf("9 - End program\n");  printf("\nSelection : ");  scanf("%d", &userChoice);  return(userChoice);  }  //adds a new record to the file.  void addRecord(FILE \*filePointer){  struct hardwareInventory inputValue ={0, "", "", 0.0};  printf("Enter new tool inventory number <1 - 100>: ");  scanf("%d", &inputValue.recordNumber);  printf("Enter the tool name, quantity, and cost\n");  scanf("%s %s %Lf", inputValue.toolName, inputValue.toolQuantity, &inputValue.toolCost);  //we will seek the location on our data for the info and write it at that record.  fseek(filePointer, (inputValue.recordNumber - 1) \* sizeof(struct hardwareInventory), SEEK\_SET);  fwrite(&inputValue, sizeof(struct hardwareInventory), 1, filePointer);  }  //display the record at the specified location  void displayData(FILE \*filePointer){  int recordValue;  //we need these dumby structs to store the record we are searching for  struct hardwareInventory updateData = {0, "", "", 0.0};  printf("Enter the record number to display <1 - 100>: ");  scanf( "%d", &recordValue );  fseek(filePointer, (recordValue - 1) \* sizeof(struct hardwareInventory), SEEK\_SET);  fread(&updateData, sizeof(struct hardwareInventory), 1, filePointer);  if (recordValue < 0 || recordValue > 100){  printf( "Record #%d does not exist\n\n", recordValue);  }  else{  printf("Current Tool Info\n");  printf( "%-6d%-21s %-6s %-10.2f\n" ,updateData.recordNumber, updateData.toolName, updateData.toolQuantity, updateData.toolCost);  }  }  //creates the new hardware.txt file, rewinds our dat file, reads it and prints the info as it goes.  void listRecord(FILE \*readPointer){  if(readPointer != NULL){  FILE \*writeFilePointer = fopen("hardware.txt", "w");  struct hardwareInventory list = {0, "", "", 0.0 };  rewind(readPointer);  fprintf( writeFilePointer, "%s\t\t %s\t\t %s\t %s\n", "Record", "Tool Name", "Quantity", "tool Cost");  while (!feof(readPointer)){  if(!fread(&list, sizeof(struct hardwareInventory), 1, readPointer)){  break;  }  if (list.recordNumber != 0){  fprintf(writeFilePointer, "%-8d %20s %15s %20.2f\n", list.recordNumber, list.toolName, list.toolQuantity, list.toolCost);  }  }  fclose(writeFilePointer);  }  }  //deletes a record. uses an empty struct and just stores the new em  void deleteRecord(FILE \*filePointer){  int recordValue;  struct hardwareInventory deleteData;  struct hardwareInventory emptyData = {0, "", "", 0.0 };  printf( "Enter the record number to delete <1 - 100>: ");  scanf("%d", &recordValue );  printf("\nCurrent Tool info\n");  fseek(filePointer, (recordValue - 1) \* sizeof(struct hardwareInventory), SEEK\_SET);  fread(&deleteData, sizeof(struct hardwareInventory), 1, filePointer);  if (recordValue <= 0 || recordValue > 100 || deleteData.recordNumber == 0){  printf("Record #%d does not exist\n\n", recordValue);  }  else{  fseek(filePointer, (recordValue - 1) \* sizeof(struct hardwareInventory), SEEK\_SET);  fwrite(&emptyData, sizeof(struct hardwareInventory), 1, filePointer);  printf( "%-6d%-21s%-6s%-10.2f\n\n", deleteData.recordNumber, deleteData.toolName, deleteData.toolQuantity, deleteData.toolCost );  }  }  //This function updates and existing record.  void updateRecord(FILE \*filePointer){  int recordValue;  struct hardwareInventory updateData = {0, "", "", 0.0};  printf("Enter the record number to update <1 - 100>: ");  scanf( "%d", &recordValue );  fseek(filePointer, (recordValue - 1) \* sizeof(struct hardwareInventory), SEEK\_SET);  fread(&updateData, sizeof(struct hardwareInventory), 1, filePointer);  if (updateData.recordNumber == 0 || recordValue <= 0 || recordValue > 100 ){  printf("Record #%d does not exist\n\n", recordValue);  }  else{  printf("Current Tool Info\n");  printf( "%-6d%-21s %-6s%-10.2f\n",updateData.recordNumber, updateData.toolName, updateData.toolQuantity, updateData.toolCost);  printf("Enter tool name, quantity, cost\n");  scanf("%s %s %Lf", updateData.toolName, updateData.toolQuantity, &updateData.toolCost);  fseek(filePointer, (recordValue - 1) \* sizeof(struct hardwareInventory), SEEK\_SET);  fwrite(&updateData, sizeof(struct hardwareInventory), 1, filePointer);  }  } |
|  |
|  |
| Output  Your choices are:  1 - STORE a formatted text file of inventoried tools called hardware.txt for printing  2 - UPDATE a tool Record  3 - ADD a new tool record  4 - DELETE a tool record  5 - DISPLAY a tool record  9 - End program  Selection : 3  Enter new tool inventory number <1 - 100>: 3  Enter the tool name, quantity, and cost  Electric\_sander 7 57.98  Your choices are:  1 - STORE a formatted text file of inventoried tools called hardware.txt for printing  2 - UPDATE a tool Record  3 - ADD a new tool record  4 - DELETE a tool record  5 - DISPLAY a tool record  9 - End program  Selection : 3  Enter new tool inventory number <1 - 100>: 17  Enter the tool name, quantity, and cost  Hammer2 79 11.99  Your choices are:  1 - STORE a formatted text file of inventoried tools called hardware.txt for printing  2 - UPDATE a tool Record  3 - ADD a new tool record  4 - DELETE a tool record  5 - DISPLAY a tool record  9 - End program  Selection : 2  Enter the record number to update <1 - 100>: 17  Current Tool Info  17 Hammer2 79 11.99  Enter tool name, quantity, cost  Hammer 76 11.99  Your choices are:  1 - STORE a formatted text file of inventoried tools called hardware.txt for printing  2 - UPDATE a tool Record  3 - ADD a new tool record  4 - DELETE a tool record  5 - DISPLAY a tool record  9 - End program  Selection : 5  Enter the record number to display <1 - 100>: 17  Current Tool Info  17 Hammer 76 11.99  Your3 choices are:  1 - STORE a formatted text file of inventoried tools called hardware.txt for printing  2 - UPDATE a tool Record  3 - ADD a new tool record  4 - DELETE a tool record  5 - DISPLAY a tool record  9 - End program  Selection : 3  Enter new tool inventory number <1 - 100>: 99  Enter the tool name, quantity, and cost  To.be.deleted 0 1.23  Your choices are:  1 - STORE a formatted text file of inventoried tools called hardware.txt for printing  2 - UPDATE a tool Record  3 - ADD a new tool record  4 - DELETE a tool record  5 - DISPLAY a tool record  9 - End program  Selection : 4  Enter the record number to delete <1 - 100>: 99  Current Tool info  99 To.be.deleted 0 1.23  Your choices are:  1 - STORE a formatted text file of inventoried tools called hardware.txt for printing  2 - UPDATE a tool Record  3 - ADD a new tool record  4 - DELETE a tool record  5 - DISPLAY a tool record  9 - End program  Selection : 4  Enter the record number to delete <1 - 100>: 99  Current Tool info  Record #99 does not exist  Your choices are:  1 - STORE a formatted text file of inventoried tools called hardware.txt for printing  2 - UPDATE a tool Record  3 - ADD a new tool record  4 - DELETE a tool record  5 - DISPLAY a tool record  9 - End program  Selection : 1  The hardware.txt file was successfully written.  Your choices are:  1 - STORE a formatted text file of inventoried tools called hardware.txt for printing  2 - UPDATE a tool Record  3 - ADD a new tool record  4 - DELETE a tool record  5 - DISPLAY a tool record  9 - End program  Selection : 9  Process returned 0 (0x0) execution time : 120.117 s  Press any key to continue. |
| Second (my own inputs)  Your choices are:  1 - STORE a formatted text file of inventoried tools called hardware.txt for printing  2 - UPDATE a tool Record  3 - ADD a new tool record  4 - DELETE a tool record  5 - DISPLAY a tool record  9 - End program  Selection : 3  Enter new tool inventory number <1 - 100>: 55  Enter the tool name, quantity, and cost  ChainSaw 55 4.99  Your choices are:  1 - STORE a formatted text file of inventoried tools called hardware.txt for printing  2 - UPDATE a tool Record  3 - ADD a new tool record  4 - DELETE a tool record  5 - DISPLAY a tool record  9 - End program  Selection : 3  Enter new tool inventory number <1 - 100>: 65  Enter the tool name, quantity, and cost  Drill 101 14.99  Your choices are:  1 - STORE a formatted text file of inventoried tools called hardware.txt for printing  2 - UPDATE a tool Record  3 - ADD a new tool record  4 - DELETE a tool record  5 - DISPLAY a tool record  9 - End program  Selection : 2  Enter the record number to update <1 - 100>: 55  Current Tool Info  55 ChainSaw 55 4.99  Enter tool name, quantity, cost  ChainSaw 55 44.99  Your choices are:  1 - STORE a formatted text file of inventoried tools called hardware.txt for printing  2 - UPDATE a tool Record  3 - ADD a new tool record  4 - DELETE a tool record  5 - DISPLAY a tool record  9 - End program  Selection : 4  Enter the record number to delete <1 - 100>: 65  Current Tool info  65 Drill 101 14.99  Your choices are:  1 - STORE a formatted text file of inventoried tools called hardware.txt for printing  2 - UPDATE a tool Record  3 - ADD a new tool record  4 - DELETE a tool record  5 - DISPLAY a tool record  9 - End program  Selection : 5  Enter the record number to display <1 - 100>: 65  Current Tool Info  0 0.00  Your choices are:  1 - STORE a formatted text file of inventoried tools called hardware.txt for printing  2 - UPDATE a tool Record  3 - ADD a new tool record  4 - DELETE a tool record  5 - DISPLAY a tool record  9 - End program  Selection : 1  The hardware.txt file was successfully written.  Your choices are:  1 - STORE a formatted text file of inventoried tools called hardware.txt for printing  2 - UPDATE a tool Record  3 - ADD a new tool record  4 - DELETE a tool record  5 - DISPLAY a tool record  9 - End program  Selection : 9  Process returned 0 (0x0) execution time : 105.025 s  Press any key to continue. |