```
// Group number : B
// Group member name : Bhanu Teja Solipeta
// Date : 10/10/2022
// Email : bhanu.solipeta@okstate.edu
/* Description : Displaying column names according to the file selected from options.txt
                 Saving records according to the user option from the data obtained from processes
*/

//In client.c

#define MAX 200
void read_options(int sockfd)
{
        char buff[MAX],no_of[50];
        char data[MAX],option[MAX],opt[100][100];
        int n,no_of_opts;
                bzero(buff, sizeof(buff));
                read(sockfd, buff, sizeof(buff));
                printf("The Options are: \n %s \n",buff);
         char * token = strtok(buff, ",");
         int j=0;
         while(token != NULL){
               printf("%d . %s \n",j+1, token);
                                j++;
               token = strtok(NULL, ",");
         }

                printf("\nplease enter one of the options");
                scanf("%s",data);

                write(sockfd, data, sizeof(data));
```

```
        printf("the available columns are , please select one of the column\n");
                bzero(buff, sizeof(buff));
                read(sockfd, buff, sizeof(buff));
                printf("%s",buff);
                scanf("%s",option);


                printf("Please enter exit to quit");

                if ((strncmp(buff, "exit", 4)) == 0) {
                        printf("Client Exit...\n");
                        exit(0);
                }
}

//Option2: Saving the records

 if(sel == 2){
                FILE *fptr = fopen("op.txt", "wb");
                fwrite(fie, sizeof(char), sizeof(fie), fptr);
                fclose(fptr);
        }

        }else if(atoi(input) == 2){
          FILE *fptr = fopen("out.txt","wb");
           for(int i=0;i<200;i++){
          if(strlen(arr[i]) !=0){
                char* token = strtok(arr[i], s);
                int j=1;
                //printf("===> %d \n",flag);
                while(token != 0){
```

```
                    if(j== flag){
                        token = strtok(0,s);
                    }
                        fprintf(fptr, " %s,", token);
                        token = strtok(0,s);
                        j++;

            }
            fprintf(fptr, "\n");
            fprintf(fptr, "\n");

        }


//In server.c

// read the message from client and copy it in buffer
                    read(connfd, buff, sizeof(buff));
                    printf("%s",buff);
                    // print buffer which contains the client contents
                    if(strcmp(buff,"bookInfo.txt")==0){
                            char column_names[50] = "Book category, Star rating, Stock";
                            char column_option[100];
                            write(connfd,column_names,sizeof(column_names));
                            printf("Waiting for column options\n");
                            read(connfd, column_option, sizeof(column_option));
                            printf("%s \n",column_option);
                            if((strcmp(column_option,"book
category")==0)||(strcmp(column_option,"book")==0)){
                                    printf("%s",column_option);
                            } else if((strcmp(column_option,"star
rating")==0)||(strcmp(column_option,"star")==0)){
```

```
                printf("%s",column_option);
            }else if(strcmp(column_option,"stock")){
                printf("%s",column_option);
            }else{
                printf("Hello world \n");
            }


        }else if(strcmp(buff,"amazonBestsellers.txt")==0){
            char column_names[50] = "User rating, Year, Genre";
            char column_option[100];
            write(connfd,column_names,sizeof(column_names));
            printf("Waiting for column options\n");
            read(connfd, column_option, sizeof(column_option));
            printf("%s \n",column_option);
            if((strcmp(column_option,"user
rating")==0)||(strcmp(column_option,"user")==0)){
                printf("%s",column_option);
            } else if(strcmp(column_option,"year")==0){
                printf("%s",column_option);
            }else if(strcmp(column_option,"genre")){
                printf("%s",column_option);
            }else{
                printf("Hello world \n");
            }
        }
        else{
            printf("option Not Available");

        }
```

```
                // if msg contains "Exit" then server exit and chat ended.
                if (strncmp("exit", buff, 4) == 0) {
                        printf("Server Exit...\n");
                //      break;
                }
}

//In process.c

 //copy process results into arr
        char arr[200][200];
        for(int i = 0; i < 200; i++){
           strcpy(arr[i], process_array[choice][i]);
        }

        //send array to server
        if(write(fout[1], arr, sizeof(sizeof(char) * 200) * 200) < 0){
           return 1;
        }

      }
```

```
else if(option == 2){
    for(int i = 0; i < processes; i++){
        char process_str[100];
        sprintf(process_str, "%s\n", values[i]);
        strcat(str, process_str);
    }

        size_t length = strlen(str);
    write(fout[1], str, length);

        read(fin[0], buffer, sizeof(buffer));


         int choice;
    for(int i = 0; i< processes; i++){
        if(strcmp(values[i], buffer) == 0){
            choice = i;
            break;
        }
    }


 char arr[200][200];
    for(int i = 0; i < 200; i++){
        strcpy(arr[i], process_array[choice][i]);
    }


 if(write(fout[1], arr, sizeof(sizeof(char) * 200) * 200) < 0){
        return 1;
    }
```