

ASSIGNMENT 3 REPORT - MUHAMMET FATİH ALBAYIN

CODE FOR THE ASSIGNMENT

```
C: MuhammetFatih_Albayin.c

1  #include <stdio.h>
2  #include <stdlib.h>
3
4  double get_balance();           /* Reads balance from file */
5  void update_balance(double new_balance); /* Updates balance in the file */
6  void check_balance();          /* Displays current balance */
7  void deposit_money();          /* Allows user to deposit money */
8  void withdraw_money();         /* Allows user to withdraw money */
9  void menu();                   /* Displays the ATM menu */
10
11 int main(){
12
13     menu();                      /* Starts the process by calling menu function */
14
15     return 0;
16 }
17
18 void menu(){
19
20     int opt;                      /* To store the option */
21
22     printf("-----Virtual ATM-----\n"); /* Takes the option input from the user */
23     printf("1. Check Balance\n");
24     printf("2. Deposit Money\n");
25     printf("3. Withdraw Money\n");
26     printf("4. Exit\n");
27     printf("Choose an option: ");
28     scanf("%d", &opt);
29
30     switch(opt){                  /* Calls the required function according to the choice */
31     case 1:
32         check_balance();
33         break;
34     case 2:
35         deposit_money();
36         break;
37     case 3:
38         withdraw_money();
39         break;
40     case 4:
41         printf("Terminating the virtual machine...\n");
42         return ;
43         break;
44     default:
45         printf("Please choose a valid option!\n"); /* Tells the user to input something valid in case it is not */
46         menu();
47     }
48 }
49
50
51 double get_balance(){
52     FILE *fptr;
53
54     fptr = fopen("account.txt", "r");
55
56     double balance;              /* To store the balance */
57     double initial = 100;        /* For the initial balance */
58
59     if(fptr == NULL){            /* Creates the file and places the initial value in it if the file does not exist */
60         fptr = fopen("account.txt", "w");
61         fprintf(fptr, "%lf", initial);
62         fclose(fptr);
63
64         fptr = fopen("account.txt", "r");
65         fscanf(fptr, "%lf", &balance);
66         fclose(fptr);
67     }
68
69     else{
70         fscanf(fptr, "%lf", &balance); /* Gets the balance from the file if it already exists */
71         fclose(fptr);
72     }
73     return balance;              /* Returns the balance value */
74 }
75
76 void update_balance(double new_balance){
77
78     FILE *fptr;
79 }
```

```

80  fptr = fopen("account.txt", "r+");
81
82  fprintf(fptr, "%lf", new_balance);          /* Replaces the new balance value with the old one */
83
84  fclose(fptr);
85
86  }
87
88  void check_balance(){                        /* Checks the balance value and prints it */
89
90  double balance = get_balance();
91
92  printf("Current balance: %.2lf TRY\n", balance);
93
94  }
95
96  void deposit_money(){
97
98  double depo;                               /* To store the deposit amount */
99  double current = get_balance();            /* To get the current balance */
100
101  printf("Enter the amount to deposit: ");   /* Gets the amount to deposit */
102  scanf("%lf", &depo);
103
104  if(depo < 0){                               /* Warns the user if they try to deposit a negative value */
105      printf("Deposit unsuccessful, you cannot deposit a negative value!\n");
106  }
107
108  else{
109      current += depo;                         /* Increments the current value */
110      update_balance(current);                 /* Updates the value in the file */
111      printf("Deposit successful! New balance: %.2lf TRY\n", current); /* Informs the user */
112  }
113
114  }
115
116  void withdraw_money(){
117
118  double wdr;                               /* To store the withdraw amount */
119  double current = get_balance();            /* To get the current balance */
120
121  printf("Enter the amount to withdraw: ");   /* Gets the amount to withdraw */
122  scanf("%lf", &wdr);
123
124  if(wdr < 0){                               /* Warns the user if they try to withdraw a negative value */
125      printf("Withdraw unsuccessful, you cannot withdraw a negative value!\n");
126  }
127
128  else if(wdr > current){                     /* Warns the user if they try to withdraw an amount more than the balance */
129      printf("Withdraw unsuccessful! Your balance is only %.2lf TRY\n", current);
130  }
131
132  else{
133      current -= wdr;                         /* Decrements the current balance */
134      update_balance(current);                 /* Updates the value in the file */
135      printf("Withdraw successful! New balance: %.2lf TRY\n", current); /* Informs the user */
136  }
137
138  }

```

THE GENERATED OUTPUT

```

albay@albay-VirtualBox:~/Desktop$ gcc -ansi MuhammetFatih_Albayın.c -o m
albay@albay-VirtualBox:~/Desktop$ ./m
-----Virtual ATM-----
1. Check Balance
2. Deposit Money
3. Withdraw Money
4. Exit
Choose an option: 1
Current balance: 100.00 TRY
albay@albay-VirtualBox:~/Desktop$ ./m
-----Virtual ATM-----

```

```
1. Check Balance
2. Deposit Money
3. Withdraw Money
4. Exit
Choose an option: 2
Enter the amount to deposit: 50
Deposit successful! New balance: 150.00 TRY
albay@albay-VirtualBox:~/Desktop$ ./m
-----Virtual ATM-----
1. Check Balance
2. Deposit Money
3. Withdraw Money
4. Exit
Choose an option: 3
Enter the amount to withdraw: 30
Withdraw successful! New balance: 120.00 TRY
albay@albay-VirtualBox:~/Desktop$ ./m
-----Virtual ATM-----
1. Check Balance
2. Deposit Money
3. Withdraw Money
4. Exit
Choose an option: 3
Enter the amount to withdraw: 200
Withdraw unsuccessful! Your balance is only 120.00 TRY
albay@albay-VirtualBox:~/Desktop$
```