ExitProcess proto

WriteString proto

WriteInt64 proto

Crlf proto

ReadInt64 proto

.data

String1 db "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*", 0

String2 db "\*\* WelCome \*\*", 0

String3 db "\*\* To \*\*", 0

String4 db "\*\* OnlineBook \*\*", 0

String5 db "\*\* Shop \*\*", 0

String6 db "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*!", 0

String7 db "Enter 1 to Display Books List: ", 0

String8 db "Books List", 0

String9 db "1.English", 0

String10 db "2.Urdu", 0

String11 db "3.Islamic", 0

String45 db "4.Persian", 0

String12 db "Enter Your Choice: ", 0

String13 db "Invalid Input !", 0

String14 db "\*\*\*\*\*\*\* English Books List \*\*\*\*\*\*\*", 0

String15 db "1.Love & Wine Rs-500/-", 0

String16 db "2.Romeo & Juliet Rs-500/-", 0

String17 db "3.World War II Rs-500/-", 0

String18 db "4.Hitler Rs-500/-", 0

String19 db "5.Shinning Stars Rs-500/-", 0

String20 db "\*\*\*\*\*\*\*\* Urdu Books List \*\*\*\*\*\*\*\*\*", 0

String21 db "1.Pir-e-Kamil Rs-250/-", 0

String22 db "2.Halam Rs-250/-", 0

String23 db "3.Jannat kay Patay Rs-250/-", 0

String24 db "4.Bang-e-Dara Rs-250/-", 0

String25 db "5.Baal-e-Jabril Rs-250/-", 0

String26 db "\*\*\*\*\*\*\* Islamic Books List \*\*\*\*\*\*\*", 0

String27 db "1.Saheh Bukhari Rs-750/-", 0

String28 db "2.Saheh Muslim Rs-750/-", 0

String29 db "3.Jamia Tirmazi Rs-750/-", 0

String30 db "4.Amne Alam Rs-750/-", 0

String31 db "5.Aqwal-e-Hikmat Rs-750/-", 0

String32 db "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*", 0

String33 db "Pick Your Book: ", 0

String34 db "Enter Quantity: ", 0

String35 db "Total Price: ", 0

String36 db "1.Books List:", 0

String37 db "2.Exit:", 0

String38 db "\*\*\*\*\*\*\* Persian Books List \*\*\*\*\*\*\*", 0

String39 db "1.Equal of the Sun Rs-350/-", 0

String40 db "2.The Blood of the Sun Rs-350/-", 0

String41 db "3.Rubaiyat Rs-350/-", 0

String42 db "4.Aria Rs-350/-", 0

String43 db "5.Shadow Spinner Rs-350/-", 0

String44 db "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*", 0

quan dq ?

num dq ?

res dq ?

.code

main proc

mov rdx, offset String1

call WriteString

call Crlf

mov rdx, offset String2

call WriteString

call Crlf

mov rdx, offset String3

call WriteString

call Crlf

mov rdx, offset String4

call WriteString

call Crlf

mov rdx, offset String5

call WriteString

call Crlf

mov rdx, offset String6

call WriteString

call Crlf

; Show string 7 and wait for user input 1

mov rdx, offset String7

call WriteString

; Read user input

call ReadInt64

cmp rax, 1

je DisplayBooksList

; Invalid input

mov rdx, offset String13

call WriteString

call Crlf

call ExitProcess

DisplayBooksList:

; Show string 8, 9, 10, 11, 45

mov rdx, offset String8

call WriteString

call Crlf

mov rdx, offset String9

call WriteString

call Crlf

mov rdx, offset String10

call WriteString

call Crlf

mov rdx, offset String11

call WriteString

call Crlf

mov rdx, offset String45

call WriteString

call Crlf

; Show string 12 and wait for user input

mov rdx, offset String12

call WriteString

call ReadInt64

cmp rax, 1

je EnglishBooksList

cmp rax, 2

je UrduBooksList

cmp rax, 3

je IslamicBooksList

cmp rax, 4

je PersianBooksList

jmp EndProgram

EnglishBooksList:

; Show string 14 to 19

mov rdx, offset String14

call WriteString

call Crlf

mov rdx, offset String15

call WriteString

call Crlf

mov rdx, offset String16

call WriteString

call Crlf

mov rdx, offset String17

call WriteString

call Crlf

mov rdx, offset String18

call WriteString

call Crlf

mov rdx, offset String19

call WriteString

call Crlf

; Show string 33 and wait for user input

mov rdx, offset String33

call WriteString

call ReadInt64

mov num,rax

; Assume user input is stored in rax

; Show string 34 and wait for user input

mov rdx, offset String34

call WriteString

call ReadInt64

mov quan, rax

; Multiply 500 by the user input

mov rax, 500

mov rbx, quan

imul rax,rbx

mov res, rax

; Show string 35 and display the result

mov rdx, offset String35

call WriteString

mov rax,res

call WriteInt64

call Crlf

call Crlf

; Show string 36 and 37 and wait for user input

mov rdx, offset String36

call WriteString

call Crlf

mov rdx, offset String37

call WriteString

call Crlf

mov rdx, offset String12

call WriteString

; Read user input

call ReadInt64

; Check user input

cmp rax, 1

je DisplayBooksList

UrduBooksList:

; Show string 20 to 25

mov rdx, offset String20

call WriteString

call Crlf

mov rdx, offset String21

call WriteString

call Crlf

mov rdx, offset String22

call WriteString

call Crlf

mov rdx, offset String23

call WriteString

call Crlf

mov rdx, offset String24

call WriteString

call Crlf

mov rdx, offset String25

call WriteString

call Crlf

; Show string 33 and wait for user input

mov rdx, offset String33

call WriteString

call ReadInt64

mov num, rax

; Assume user input is stored in rax

; Show string 34 and wait for user input

mov rdx, offset String34

call WriteString

call ReadInt64

mov quan, rax

; Assume user input is stored in rax

; Multiply 250 by the user input

mov rax, 250

mov rbx, quan

imul rax,rbx

mov res, rax

; Show string 35 and display the result

mov rdx, offset String35

call WriteString

mov rax, res

call WriteInt64

call Crlf

call Crlf

; Show string 36 and 37 and wait for user input

mov rdx, offset String36

call WriteString

call Crlf

mov rdx, offset String37

call WriteString

call Crlf

mov rdx, offset String12

call WriteString

; Read user input

call ReadInt64

; Check user input

cmp rax, 1

je DisplayBooksList

IslamicBooksList:

; Show string 26 to 32

mov rdx, offset String26

call WriteString

call Crlf

mov rdx, offset String27

call WriteString

call Crlf

mov rdx, offset String28

call WriteString

call Crlf

mov rdx, offset String29

call WriteString

call Crlf

mov rdx, offset String30

call WriteString

call Crlf

mov rdx, offset String31

call WriteString

call Crlf

mov rdx, offset String32

call WriteString

call Crlf

; Show string 33 and wait for user input

mov rdx, offset String33

call WriteString

call ReadInt64

; Assume user input is stored in rax

; Show string 34 and wait for user input

mov rdx, offset String34

call WriteString

call ReadInt64

mov quan, rax

; Assume user input is stored in rax

; Multiply 750 by the user input

mov rax, 750

mov rbx, quan

imul rax,rbx

mov res, rax

; Show string 35 and display the result

mov rdx, offset String35

call WriteString

; Display the Result

mov rax, res

call WriteInt64

call Crlf

call Crlf

; Show string 36 and 37 and wait for user input

mov rdx, offset String36

call WriteString

call Crlf

mov rdx, offset String37

call WriteString

call Crlf

mov rdx, offset String12

call WriteString

; Read user input

call ReadInt64

; Check user input

cmp rax, 1

je DisplayBooksList

PersianBooksList:

; Show string 38 to 44

mov rdx, offset String38

call WriteString

call Crlf

mov rdx, offset String39

call WriteString

call Crlf

mov rdx, offset String40

call WriteString

call Crlf

mov rdx, offset String41

call WriteString

call Crlf

mov rdx, offset String42

call WriteString

call Crlf

mov rdx, offset String43

call WriteString

call Crlf

mov rdx, offset String44

call WriteString

call Crlf

; Show string 33 and wait for user input

mov rdx, offset String33

call WriteString

call ReadInt64

; Assume user input is stored in rax

; Show string 34 and wait for user input

mov rdx, offset String34

call WriteString

call ReadInt64

mov quan, rax

; Assume user input is stored in rax

; Multiply 350 by the user input

mov rax, 350

mov rbx, quan

imul rax,rbx

mov res, rax

; Show string 35 and display the result

mov rdx, offset String35

call WriteString

; Display the Result

mov rax, res

call WriteInt64

call Crlf

call Crlf

; Show string 36 and 37 and wait for user input

mov rdx, offset String36

call WriteString

call Crlf

mov rdx, offset String37

call WriteString

call Crlf

mov rdx, offset String12

call WriteString

; Read user input

call ReadInt64

; Check user input

cmp rax, 1

je DisplayBooksList

EndProgram:

call ExitProcess

main endp

end