

Patience

by JohnnyD

Hi all, Here's my solitaire game. QCARD32.DLL is required in the LB folder

The object of this game is to place cards in pairs over existing cards such that the value of each pair of covered cards adds up to eleven.

(e.g. 6 & 5, 7 & 4, Jack & Ace, etc)

Suits are irrelevant. Ace counts as one, picture cards count as ten.

A Jack-Queen-King sequence is also permitted.

```
nomainwin
dim saved(3,4):dim deck(52,2):dim pos(9,4)
for a=0 to 3
  for b= 1 to 13
    deck(a*13+b,1)=a*13+b:deck(a*13+b,2)=b
  next b
next a
restore[posn]
for a= 1 to 9
  read b,c:pos(a,1)=b:pos(a,2)=c
next a
```

```
[posn]
data 10,2,110,2,210,2,10,102,110,102,210,102,10,202,110,202,210,
202
```

```
WindowWidth=306:WindowHeight=382
UpperLeftX=(DisplayWidth-WindowWidth)/2
UpperLeftY=(DisplayHeight-WindowHeight)/2
menu #1,"&Options",&Deal",[deal],&Help",[help],"E&xit",[quit]
graphicbox #1.g,0,0,300,330
open "Patience" for window_nf as #1
#1 "trapclose [quit]"
#1.g "setfocus; when leftButtonUp [checkIndex]"
hBox=hwnd(#1.g)
open "qcard32.dll" for dll as #qc
calldll #qc,"InitializeDeck",hBox as long,r as long
```

```
[deal]
dealt=0:clicks=0:card=0:score=0:pic=0
```

```
reclaim=0:dim jk(3):nomore=0
callDll #qc,"SetDefaultValues",r as void
gosub[shuffle]
#1.g "down; fill 10 190 225"
#1.g "backcolor 10 190 225"
for i=1 to 9
    Call DealCard hBox,deck(i,1),pos(i,1),pos(i,2)
    pos(i,3)=deck(i,1):pos(i,4)=deck(i,2)
    dealt=dealt+1
next i
gosub[moretest]
```

```
[wait]
#1.g "place 10 320"
#1.g "\Cards Left:";52-dealt;" "
if nomore=1 then
    confirm "Play Again?";q$
    if q$="yes" then
        goto[deal]
    else
        goto[quit]
    end if
end if
wait
```

```
[checkIndex]
x=MouseX:y=MouseY
for a= 1 to 9
    if (x>pos(a,1)) and (x<pos(a,1)+70) and (y>pos(a,2)) and (y<
pos(a,2)+95) then exit for
next a
card=a
if a=10 then
    card=0:goto[wait]
end if
dealt=dealt+1
#1.g "place 80 320":#1.g "\";52-dealt;" "
clicks=clicks+1
if pos(card,4)>10 then
    score=score+10
else
    score=score+pos(card,4)
end if
saved(clicks,1)=card:saved(clicks,2)=pos(card,3):saved(
clicks,3)=pos(card,4)
Call DealCard hBox,deck(dealt,1),pos(card,1),pos(card,2)
```

```
if pos(card,4)<11 then[2pictest]
pic=pic+1
if jk(pos(card,4)-10)=pos(card,4) then[gotpic]
jk(pos(card,4)-10)=pos(card,4)
pos(card,3)=deck(dealt,1):pos(card,4)=deck(dealt,2)
if pic=3 then[zeroscore]
goto[2clicktest]
```

```
[gotpic]
select case pos(card,4)
case 11
name$="J"
case 12
name$="Q"
case 13
name$="K"
end select
notice "Already Have a ";name$;"!":dim jk(3)
```

```
[redeal]
reclaim=clicks:gosub[redealold]
```

```
[zeroscore]
score=0:pic=0:clicks=0:reclaim=0:dim jk(3)
gosub[moretest]
goto[dealtall]
```

```
[2pictest]
if pic=2 then
notice "Can't Have ";pos(card,4);" in J-Q-K Run!"
dim jk(3)
goto[redeal]
end if
```

```
[2clicktest]
pos(card,3)=deck(dealt,1):pos(card,4)=deck(dealt,2)
if pic>1 then[dealtall]
if clicks=2 then
reclaim=clicks:clicks=0
if score <> 11 then
dim name$(2)
for a= 1 to 2
name$(a)=str$(saved(a,3))
select case name$(a)
case "1"
name$(a)="A"
```

```
        case "11"
            name$(a)="J"
        case "12"
            name$(a)="Q"
        case "13"
            name$(a)="K"
        end select
    next a
    notice name$(1)+" and "+name$(2)+" is not 11!"
    gosub[redealold]
else
    score=0:reclaim=0:pic=0:dim jqk(3)
    gosub[moretest]
end if
end if

[dealtall]
    if dealt=52 then
        confirm "You Did it! Play Again?";q$
        if q$="yes" then [deal] else [quit]
    end if
    goto[wait]

[shuffle]
    for a=1 to 500
        b=int(rnd(0)*52)+1:c=int(rnd(0)*52)+1
        d=deck(b,1):dd=deck(b,2)
        deck(b,1)=deck(c,1):deck(b,2)=deck(c,2)
        deck(c,1)=d:deck(c,2)=dd
    next a
    return

[redealold]
    for a=1 to reclaim
        Call DealCard hBox,saved(a,2),pos(saved(a,1),1),pos(saved(a,1),2)
        pos(saved(a,1),3)=saved(a,2):pos(saved(a,1),4)=saved(a,3)
    next a
    dealt=dealt-reclaim:score=0:reclaim=0:pic=0:dim jqk(3)
    return

[quit]
    close #qc:close #1:end

[moretest]
    more=0:nomore=0
```

```
for a= 1 to 8
  for b=a+1 to 9
    if pos(a,4)+pos(b,4)=11 then more=1
  next b
next a
jqk=0
for a=1 to 9
  if pos(a,4)>10 then jqk=1
next a
for a= 1 to 9
  if pos(a,4)=1 and jqk=1 then more=1
next a
dim jqktest(3)
for a=1 to 9
  if pos(a,4)>10 then jqktest(pos(a,4)-10)=1
next a
if jqktest(1)=1 and jqktest(2)=1 and jqktest(3)=1 then more=1
if more=0 then nomore=1: notice "No More Moves"
return
```

[help]

```
  help$="Help"+chr$(13)
  help$=help$+
"The object of the game is place pairs of cards on"+chr$(13)
  help$=help$+
"existing cards such that each pair of covered cards"+chr$(13)
  help$=help$+
"adds up to eleven, until the pack is used up."+chr$(13)+chr$(13)
  help$=help$+"Suit is irrelevant, Ace counts as one,"+chr$(13)
  help$=help$+"picture cards count as ten."+chr$(13)
  help$=help$+
"Example moves are 6 & 5, 8 & 3, Jack & Ace, etc."+chr$(13)+chr$(13)
  help$=help$+
"A Jack - Queen - King run is also permitted."+chr$(13)
  help$=help$+
"Incorrect moves will be indicated and the deck will back up."
  notice help$
  wait
```

```
Sub DealCard hndle,nC,x,y
  calldll #qc,"DealCard",hndle as long,nC as long,x as long,y as
long,r as void
End Sub
```