

```

'=== Pong ===
'=== by Doctor&Rose ==
'=== use the a, z, and arrow keys to move the paddles ===

_TITLE "Pong"
SCREEN 12
_FULLSCREEN ' Take out if you dont want fullscreen or press alt-enter

PRINT "Input the speed at which you want the ball to go. . ."
PRINT "    0.5 = SLOW"
PRINT "    1 = NORMAL"
PRINT "    2 = FAST"
PRINT "    3 = EXTREMELY FAST"
PRINT "    4 = INSANELY FAST"
INPUT ballspeed
PRINT
PRINT "Input the speed at which you want your padels to go. . ."
PRINT "    0.5 = VERY SLOW"
PRINT "    1 = SLOW"
PRINT "    2 = NORMAL"
PRINT "    3 = FAST"
PRINT "    4 = EXTREMELY FAST"
INPUT PD
PRINT
PRINT "          CONTROLS"
PRINT
PRINT "PLAYER 1          PLAYER 2"
PRINT " UP = ^          UP = A"
PRINT " DOWN = V        DOWN = Z"
PRINT
PRINT "          [SPACE] to start"
DO
    DO: K$ = UCASE$(INKEY$)
    LOOP UNTIL K$ = CHR$(32)
    IF K$ = CHR$(32) THEN GOTO STARTPONG
LOOP

STARTPONG:
'Load a sound to play
h& = _SNDOPEN("hit.wav")

' Draw a filled circle for our ball sprite
CIRCLE (5, 5), 5, 15
PAINT (6, 6), 15
LINE (0, 50)-(10, 110), 15, BF

' Set aside enough space to hold the sprites
DIM Ball%(200)
DIM RPaddle%(500)
DIM LPaddle%(500)

```

```

'CONST rightkey& = 19712
CONST downkey& = 20480
'CONST leftkey& = 19200
CONST upkey& = 18432
'CONST enterkey& = 13
CONST esckey& = 27
CONST spacekey& = 32
CONST akey& = 97
CONST zkey& = 122

' Get the sprites into the Ball%() & Paddle%() arrays

GET (0, 0)-(10, 10), Ball%()
GET (0, 50)-(10, 110), RPaddle%()
GET (0, 50)-(10, 110), LPaddle%()


10 CLS
'draw the border around the play area
LINE (0, 0)-(640, 480), 1, BF
LINE (5, 5)-(635, 475), 0, BF


'Print the scores
LOCATE 2, 10: PRINT LScore: LOCATE 2, 70: PRINT RScore


' Screen size
CONST XMax = 640
CONST YMax = 480


' Starting position of the sprites (ball & paddles)
X = 250
Y = 10
PDY1 = 200
PDY2 = 200


' Motion of ball
DX = ballspeed * -1
DY = ballspeed


DO

' Draw the sprites at their current position
PUT (X, Y), Ball%()
PUT (10, PDY1), LPaddle%()
PUT (620, PDY2), RPaddle%()


' Checks to see if the ball has hit the Left Paddle
IF X < 20 THEN
    IF Y > PDY1 THEN
        IF Y < PDY1 + 60 THEN
            DX = ballspeed
            IF h& THEN _SNDPLAY h& 'plays a sound

```

```

        END IF
    END IF
END IF

' Checks to see if the ball has hit the Right Paddle
IF X > 610 THEN
    IF Y > PDY2 THEN
        IF Y < PDY2 + 60 THEN
            DX = ballspeed * -1
            IF h& THEN _SNDPLAY h& 'plays a sound
        END IF
    END IF
END IF

' Save the old positions so that we can remove the sprites before
' we redraw them in their new location.
PX = X
PY = Y
PDY1OLD = PDY1
PDY2OLD = PDY2

' check to see if keys are pressed and move paddles
' check to see if paddle is at edge of screen
' if so dont let it go further

'Left paddle (player 1)
IF _KEYDOWN(akey&) THEN
    PDY1 = PDY1 - PD
    IF PDY1 < 6 THEN
        PDY1 = 6
    END IF
END IF

IF _KEYDOWN(zkey&) THEN
    PDY1 = PDY1 + PD
    IF PDY1 > YMax - 66 THEN
        PDY1 = YMax - 66
    END IF
END IF

'Right paddle (player 2)
IF _KEYDOWN(upkey&) THEN
    PDY2 = PDY2 - PD
    IF PDY2 < 6 THEN
        PDY2 = 6
    END IF
END IF

IF _KEYDOWN(downkey&) THEN
    PDY2 = PDY2 + PD
    IF PDY2 > YMax - 66 THEN
        PDY2 = YMax - 66
    END IF

```

END IF

' Change the ball position

X = X + DX

Y = Y + DY

' If we have hit the right-hand side of the screen

IF X > XMax - 15 THEN

LScore = LScore + 1

_AUTODISPLAY

LOCATE 2, 10: PRINT LScore: LOCATE 2, 70: PRINT RScore

LOCATE 15, 33: PRINT "PRESS SPACE BAR"

LOCATE 17, 35: PRINT "ESC to EXIT"

'DO: LOOP UNTIL INKEY\$ = ""

DO

IF _KEYDOWN(esckey&) THEN SYSTEM

LOOP UNTIL _KEYDOWN(spacekey&)

GOTO 10

END IF

' If we have hit the left-hand side of the screen

IF X < 1 THEN

RScore = RScore + 1

_AUTODISPLAY

LOCATE 2, 10: PRINT LScore: LOCATE 2, 70: PRINT RScore

LOCATE 15, 33: PRINT "PRESS SPACE BAR"

LOCATE 17, 35: PRINT "ESC to EXIT"

'DO: LOOP UNTIL INKEY\$ = ""

DO

IF _KEYDOWN(esckey&) THEN SYSTEM

LOOP UNTIL _KEYDOWN(spacekey&)

GOTO 10

END IF

' If we have hit the bottom of the screen

IF Y > YMax - 20 THEN

' Make the sprite go up

DY = ballspeed * -1

END IF

' If we have hit the top of the screen

IF Y < 6 THEN

' Make the sprite go down

DY = ballspeed

END IF

_DISPLAY

'_DELAY .005

' Remove the sprites from the screen before redrawing

```
' it in its new position at the top of the loop.  
PUT (PX, PY), Ball%()  
PUT (10, PDY1OLD), LPaddle%()  
PUT (620, PDY2OLD), RPaddle%()  
  
LOOP UNTIL _KEYDOWN(esckey&)  
SYSTEM
```