

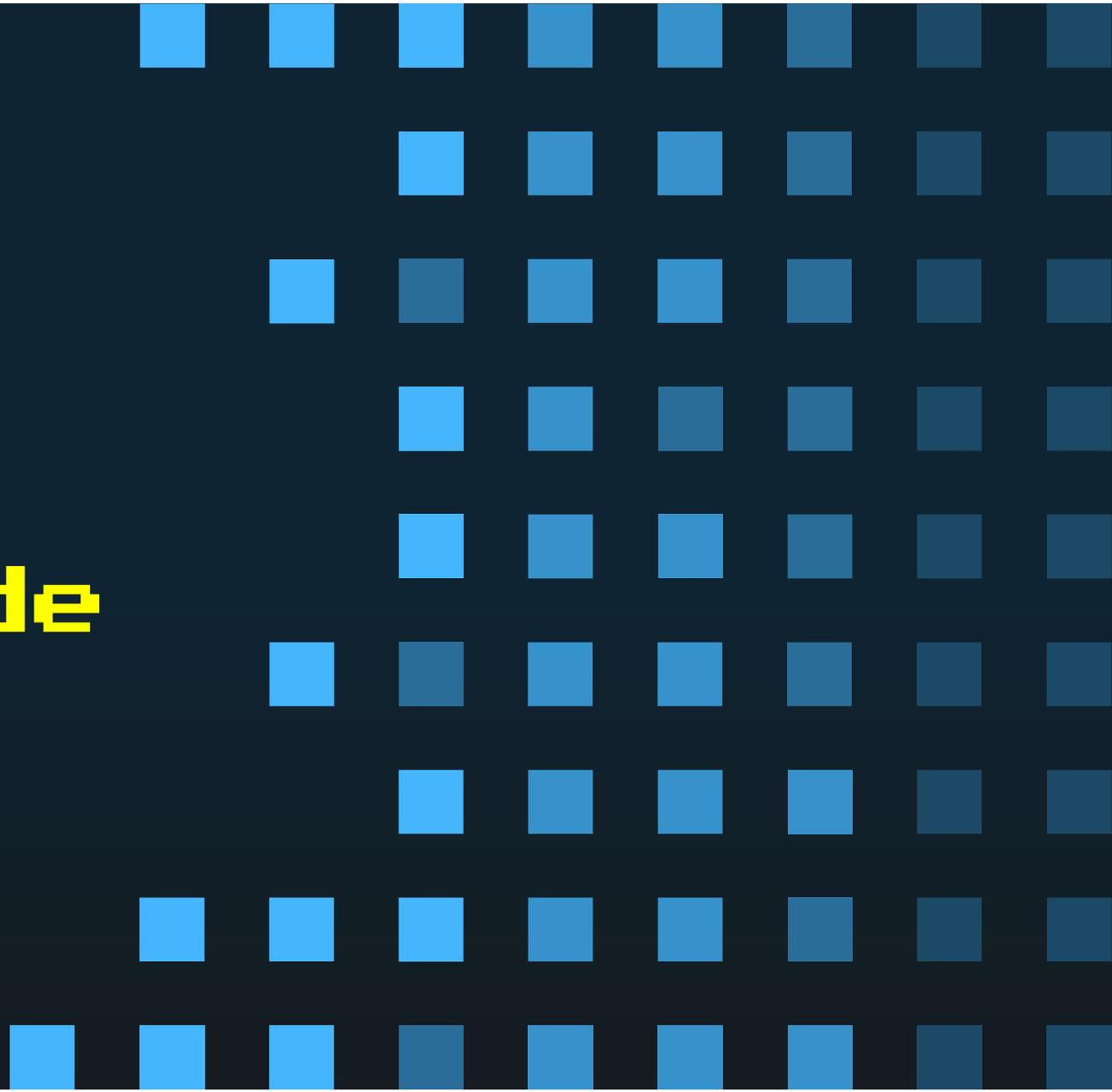
Programação RETR0

QB64

PSINF

12

Criação de
Jogos



SCREEN

- ❑ O QB64 permite a utilização de **alta resolução** (até 8k)
- ❑ Sintaxe:

```
Screen _NewImage(res_Hor, res_Vert, Num_Cores)
```

Exemplos

```
Screen _NewImage(1280, 1024, 256) → 256 cores
```

```
Screen _NewImage(1280, 1024, 32) → 32 bit – o que vamos utilizar
```

```
Screen _NewImage(1280, 1024, 13) → antigo screen 13 com maior  
resolução. Aceita ainda 0, 2, 7
```

IMAGENS

- ❑ Podemos carregar **imagens** para o nosso **programa**
- ❑ Indicar a **Path** a partir do local onde está gravado o FX .bas
- ❑ Utilizar o Screen de **32 Bit** (menos bugs)
- ❑ Sintaxe:

```
_LoadImage("Path/nomeimagem.extensão")
```

Esta imagem **não** será **mostrada** até utilizarmos a função seguinte:

```
_PutImage(coordenadaX, coordenadaY), imagem
```

Devem-se remover da **memória** sempre que não estão a ser utilizadas_
_FreeImage nome_da_imagem

IMAGENS

- Exemplo:

```
imagens.bas - QB64 x64
File Edit View Search Run Debug Options Tools Help
imagens.bas*
1 Dim imagem As Long
2 Screen _NewImage(1600, 1200, 32)
3 imagem = _LoadImage("imagens/car1.png")
4 _PutImage (0, 0), imagem
```

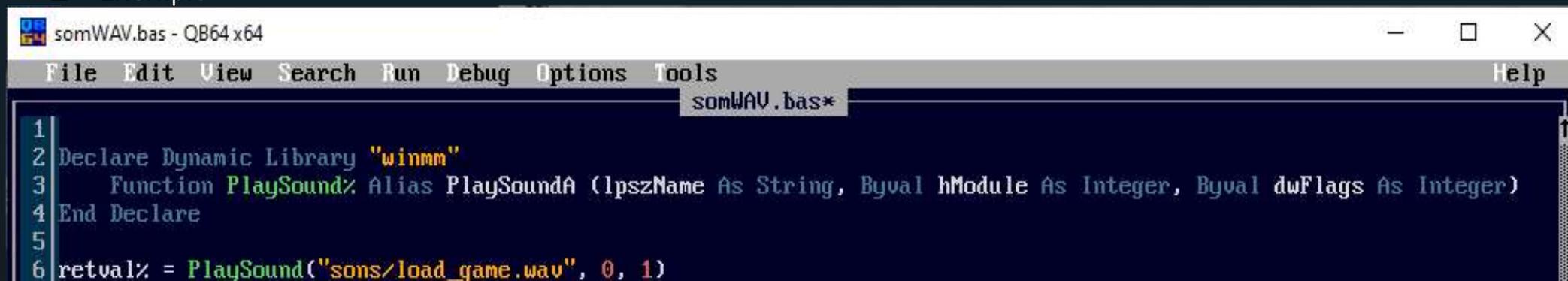


Som

- ❑ A capacidade de gerar som já existia no Qbasic4.5
- ❑ O comando **BEEP** só por si já permite emitir um som
- ❑ O comando **PLAY** permite vários parâmetros (para músicos)
 - ❑ O - Octave (0 - 6): O6
< or > - Moves Up or Down One Octave
P - Pauses (1 - 64): P1 Pause Whole Note
T - Tempo (32 - 255) in Quarter Notes Per Minute: T120
A, B, C, D, E, F, G - Plays The Note in the Current Octave
Playing Length(1 - 64): C2 Plays an C Half Note
+ or # Makes The Preceding Note Sharp
- (Minus) Makes the Preceding Note Flat
 - ❑ Exemplo:
 - ❑ `PLAY "T120<<e8e8f8g8g8f8e8d8c8c8d8e8d8c12c4"`

Som

- ❑ O comando **SOUND** permite emitir a frequência de som por um determinado tempo (frequência de 37 a 32767 e duração de 0 a 65535)
 - ❑ Exemplo: SOUND 500, 4 → emite a freq 500 durante 4 ciclos de relógio
- ❑ Para tocar ficheiros **WAV** temos de utilizar uma biblioteca do **Windows Dynamic Library "winmm"**
- ❑ Utiliza-se o `PlaySounA("path Fx", 0, tipo de sincronismo)`
 - ❑ **0** síncrono – **toca o som e só depois** passa à frente
 - ❑ **1** assíncrono – **toca o som enquanto** faz outras coisas
- ❑ Exemplo:



```
somWAV.bas - QB64 x64
File Edit View Search Run Debug Options Tools Help
somWAV.bas*
1
2 Declare Dynamic Library "winmm"
3   Function PlaySound% Alias PlaySoundA (lpszName As String, ByVal hModule As Integer, ByVal dwFlags As Integer)
4 End Declare
5
6 retVal% = PlaySound("sons/load_game.wav", 0, 1)
```

Som

- Ainda para tocar outros tipos de Fx sem recorrer a uma biblioteca:

`_SNDOPEN`("path", "sync,vol"[, opções])

`_SNDPLAY`

- Ou ainda

```
Dim introsound As Long
introsound = _SndOpen("sons/load_game.wav")
_SndPlay introsound
_SndPlayFile "sons/load_game.wav", , 0.5
```

`_SndPlayFile` "Path", [nada], [volume 0 a 1]

- Fx suportados:

WAV, FLAC, OGG, MP3, MID, IT, XM, S3M, MOD, RAD (v2 only), AHX & HVL

Rato

- O QB64 permite utilizar o **rato** como INPUT.

_MOUSEINPUT

_MOUSEX – coordenada X do rato

_MOUSEY – coordenada Y do rato

_MOUSEBUTTON(1) – Botão esquerdo -1 premido 0 não premido

_MOUSEBUTTON(2) – Botão direito -1 premido 0 não premido

_MOUSEBUTTON(3) – Botão centro -1 premido 0 não premido

_MOUSEWHEEL – Roda do rato -1 rodou para cima; 0 não rodou; 1 rodou para baixo

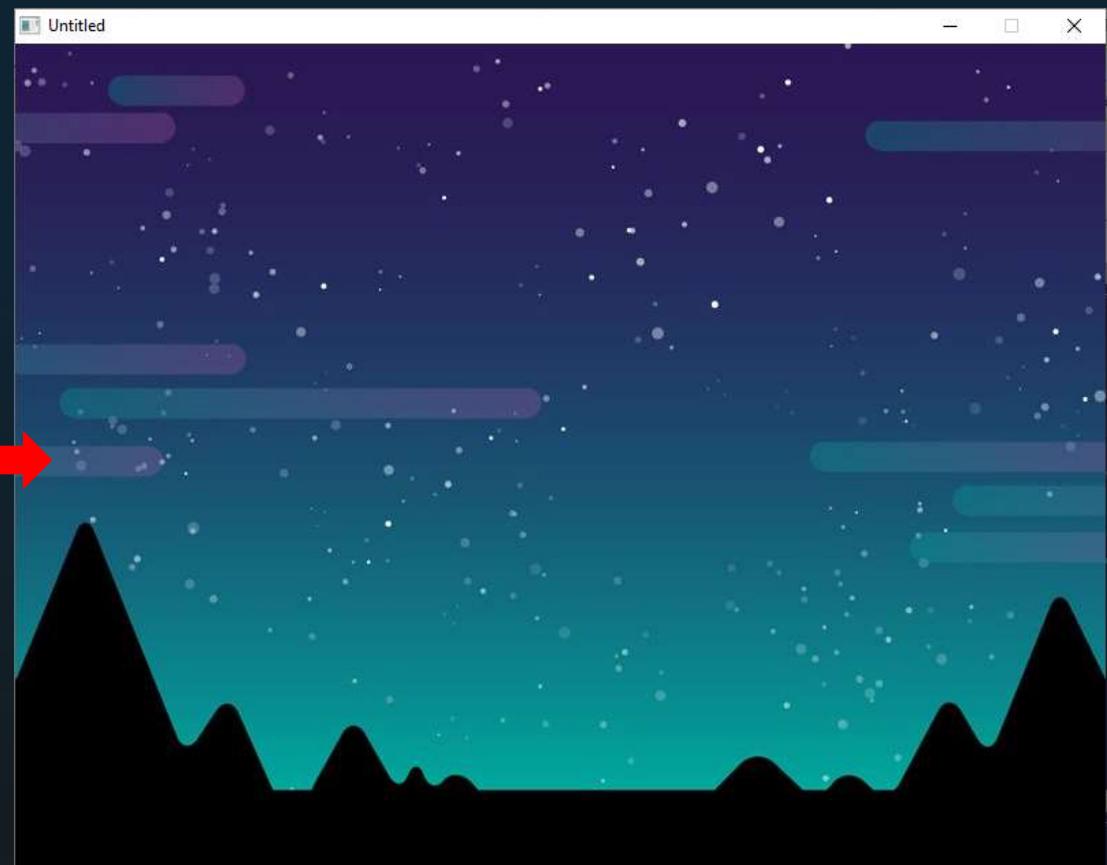
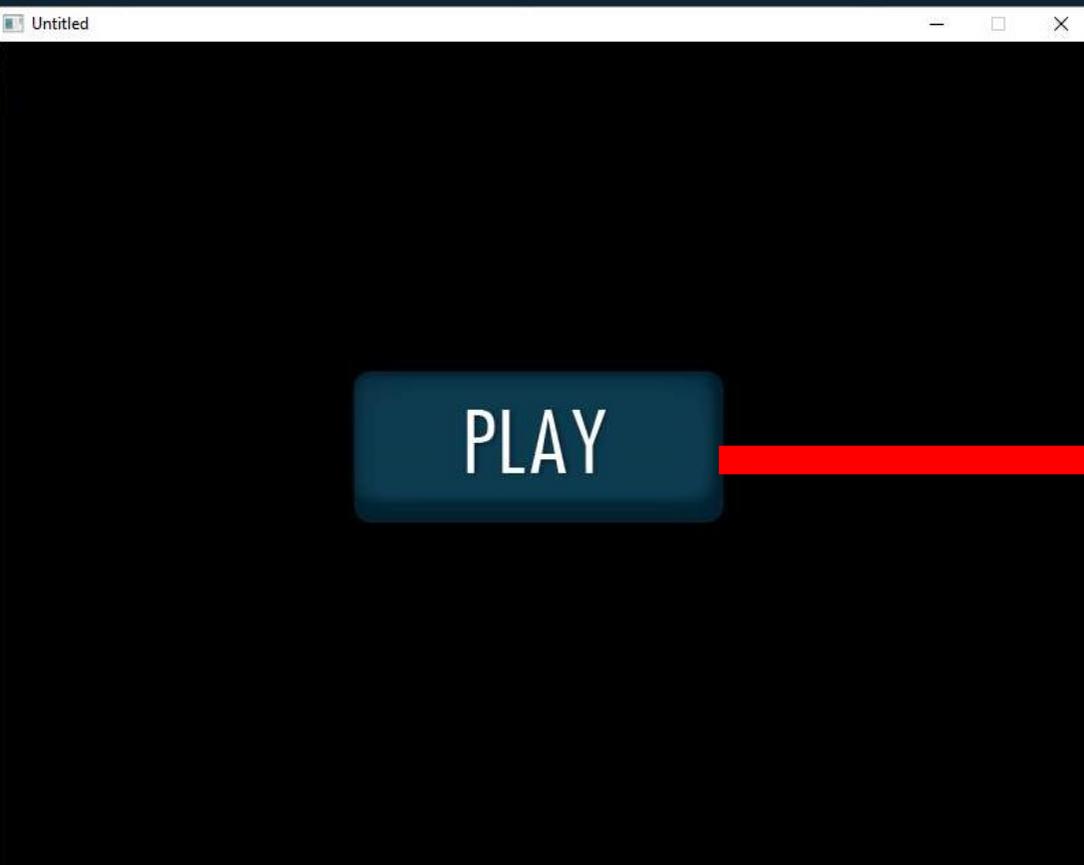
Rato

Exemplo:

```
mousebutton.bas - QB64 x64
File Edit View Search Run Debug Options Tools Help
mousebutton.bas
1 Dim largura As Integer
2 Dim altura As Integer
3 Dim botao As Long
4 largura = 800
5 altura = 600
6 Screen _NewImage(largura, altura, 32)
7 botao = _LoadImage("imagens/playButton.png")
8 _PutImage ((largura - _Width(botao)) / 2, (altura - _Height(botao)) / 2), botao 'botao centrado dinamicamente
9 'detecao de click sobre botao
10 Do
11     While _MouseInput
12         If _MouseX >= ((largura - _Width(botao)) / 2) And _MouseX <= ((largura - _Width(botao)) / 2 + _Width(botao))
13             Cls
14             fundo = _LoadImage("imagens/fundo2.jpg")
15             _PutImage (0, 0), fundo
16         ElseIf _MouseButton(1) = -1 Then
17             Locate 1, 1
18             Print "fora botao"
19         End If
20     Wend
21 Loop While INKEYS$ = "" 'para ao tocar numa tecla
```

Rato

□ Exemplo:



Para hoje

Realizar os guiões no Website.

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, and infographics & images by **Freepik** and illustrations by **Stories**