

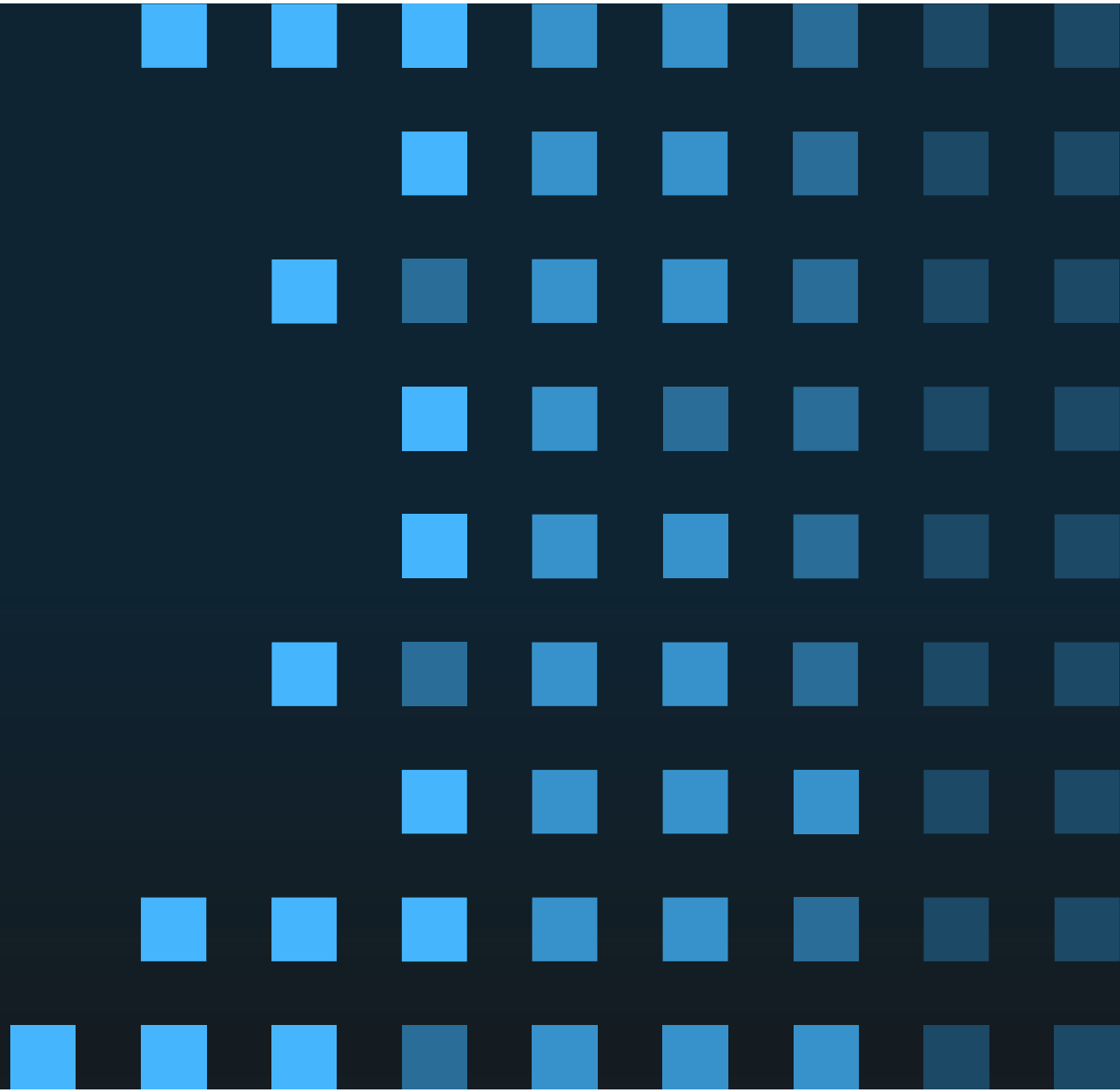
Programação RETR0

QB64

PSINF

09

Arrays



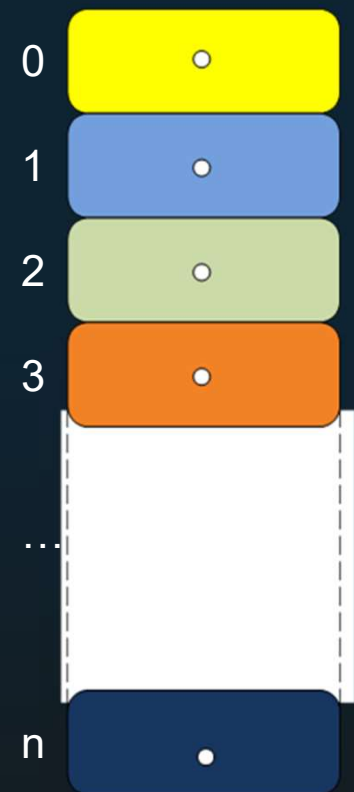
Arrays

- Um **array** é uma **variável** que pode **guardar muitos valores** acedidos através de um número
- Funciona como um **armário** de **gavetas**. Na primeira podemos ter meias, na segunda camisolas, na terceira calças, etc

Em **Qbasic** temos de **dimensionar** o array **previamente**

Em Qbasic a **primeira posição** do array é a **0** (total de posições +1 devido a **começar em zero**)

Em Qbasic o array **apenas** pode **conter dados** do **mesmo tipo** no seu Interior (ou seja no armário apenas podíamos ter meias!!).



Arrays

- Sintaxe

DIM nome_array(tamanho do array) AS Tipo_de_dados

- Atribuir valores às posições do array

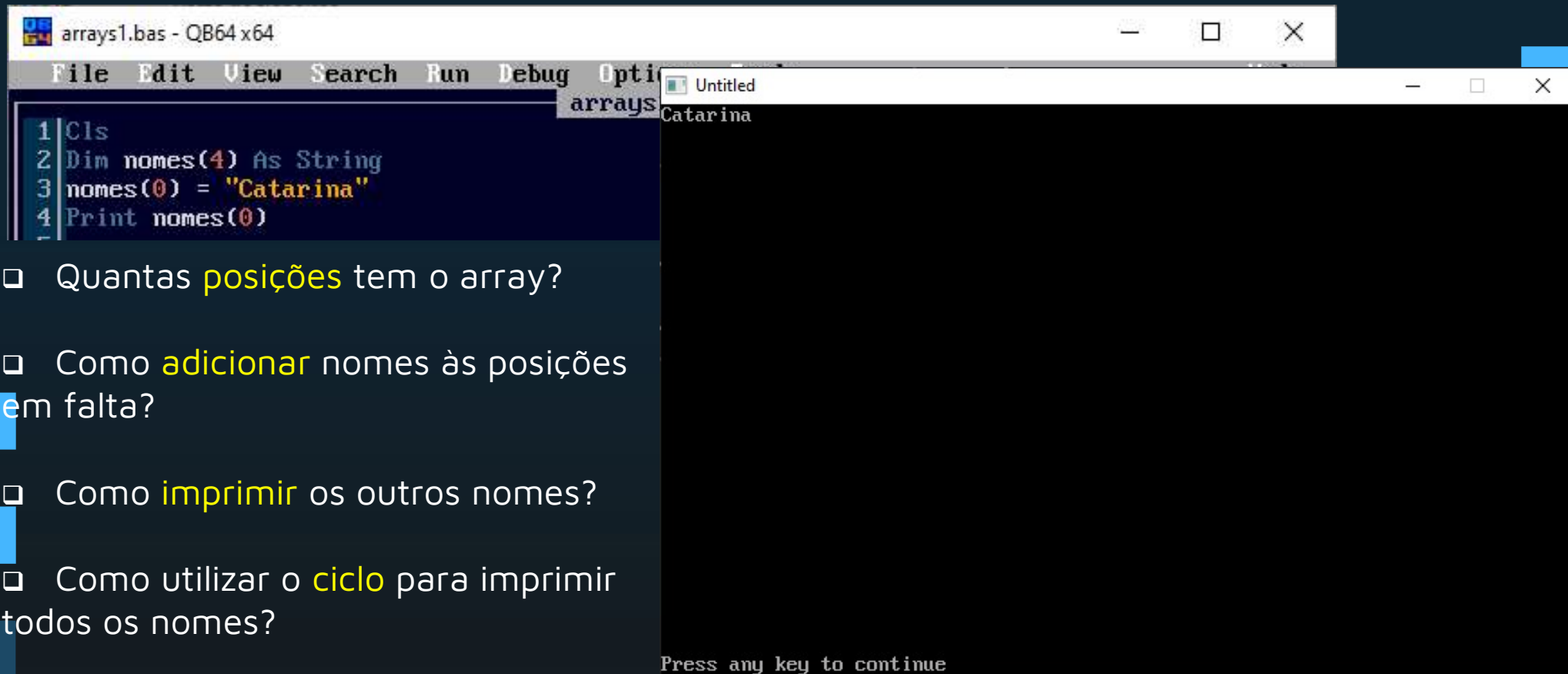
Nome_array(pos) = valor

- Imprimir conteúdo

Print nome_array(pos)

Arrays

Exemplo:



The screenshot shows a QB64 IDE window titled "arrays1.bas - QB64 x64". The code in the editor is:

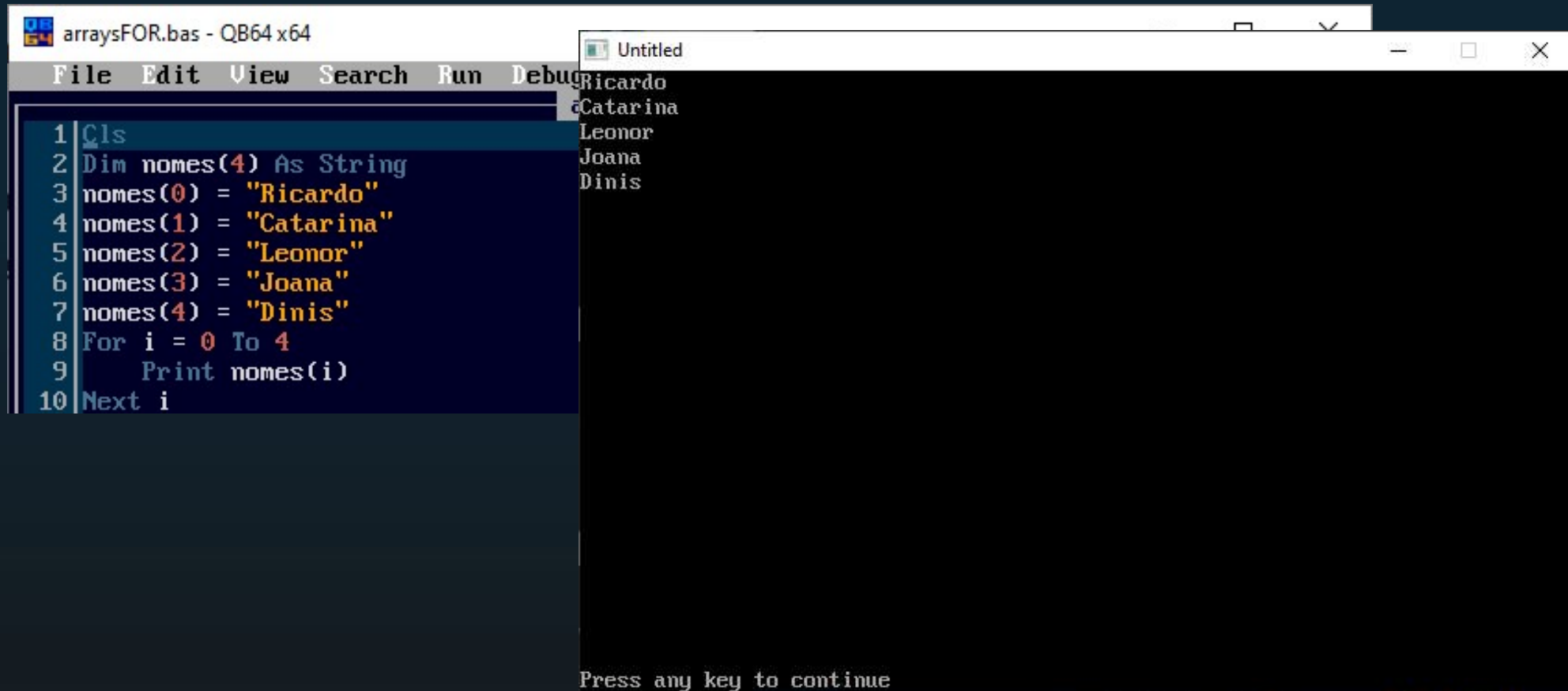
```
1 Cls
2 Dim nomes(4) As String
3 nomes(0) = "Catarina"
4 Print nomes(0)
```

Below the code editor, there is a console window titled "Untitled" showing the output "Catarina". At the bottom of the console, it says "Press any key to continue".

- Quantas **posições** tem o array?
- Como **adicionar** nomes às posições em falta?
- Como **imprimir** os outros nomes?
- Como utilizar o **ciclo** para imprimir todos os nomes?

Arrays

- Exemplo:



The screenshot shows a QB64 IDE window titled "arraysFOR.bas - QB64 x64". The code in the editor is as follows:

```
1 Cls
2 Dim nomes(4) As String
3 nomes(0) = "Ricardo"
4 nomes(1) = "Catarina"
5 nomes(2) = "Leonor"
6 nomes(3) = "Joana"
7 nomes(4) = "Dinis"
8 For i = 0 To 4
9     Print nomes(i)
10 Next i
```

The output window, titled "Untitled", displays the following text:

```
Ricardo
Catarina
Leonor
Joana
Dinis
```

At the bottom of the IDE, the text "Press any key to continue" is visible.

Arrays 2D

- Os arrays podem ter **várias dimensões**
- Um array **2D** é o que conheces por **tabela de dupla entrada**
- Quais as posições preenchidas?**
 - (0,3)
 - (1,4)
 - (3,2)

	0	1	2	3	4
0				A	
1					C
2					
3			B		
4					

Arrays 2D

- Sintaxe:
DIM nome_array(**linhas**,**colunas**) as tipo_de_dados
- Qual o **tamanho** do array seguinte?
 - (3,11)

2D Array - Matrix

1,1	1,2	1,3	1,4	1,5	1,6	1,6	1,8	1,9	1, 10	1, 11
2,1	2,2	2,3	2,4	2,5	2,6	2,7	2,8	2,9	2, 10	2, 11
3,1	3,2	3,3	3,4	3,5	3,6	3,7	3,8	3,9	3, 10	3, 11

Arrays 2D

- ❑ Limitações Qbasic
 - ❑ O array **só** pode ter **um tipo** de **dados** para todas as dimensões
 - ❑ **Não** é **possível misturar** tipos de dados

Arrays 2D

- Exemplo:

```
arrays2D.bas - QB64 x64
File Edit View Search Run Debug Options Tools Help
arrays2D.bas*
1 Cls
2 Dim nome_idade(4, 4) As String
3 'primeiro os nomes
4 nome_idade(0, 0) = "Ricardo"
5 nome_idade(1, 0) = "Catarina"
6 nome_idade(2, 0) = "Leonor"
7 nome_idade(3, 0) = "Joana"
8 nome_idade(4, 0) = "Dinis"
9 'depois as idades
10 nome_idade(0, 1) = "17"
11 nome_idade(1, 1) = "4"
12 nome_idade(2, 1) = "11"
13 nome_idade(3, 1) = "18"
14 nome_idade(4, 1) = "13"
```

	0	1
0	Ricardo	17
1	Catarina	4
2	Leonor	11
3	Joana	18
4	Dinis	13

Arrays Multidimensionais

- ❑ Podemos ter ainda **mais dimensões**
- ❑ Podemos ter até 60 dimensões no Qbasic
- ❑ Problema: aceder à informação torna-se **muito complexo**
- ❑ Há **alternativas** para guardar informação **mais intuitivas** e fáceis de aceder

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**