

Aprendendo ainda mais sobre Strings

Validação por tipo de conteúdo



Aprendendo ainda mais sobre Strings

Validação por tipo de conteúdo

Nesta aula veremos como analisar e verificar uma string para identificar letras e números em seu conteúdo e também faremos outras validações.



Aprendendo ainda mais sobre Strings

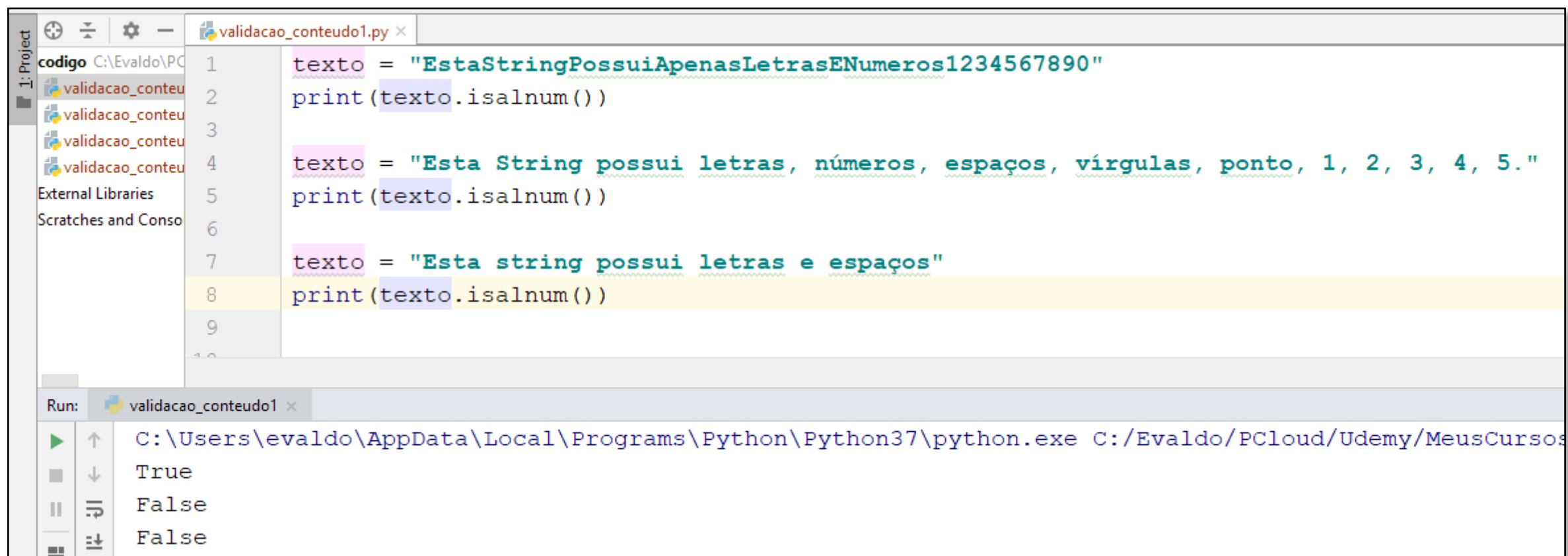
Validação por tipo de conteúdo

Para verificar se uma string tem conteúdo e se todos os seus caracteres são letras ou números podemos utilizar o método *isalnum* (é alfanumérico).

Este método retorna verdadeiro caso a string não esteja vazia e se todos seus caracteres são letras e/ou números.

Aprendendo ainda mais sobre Strings

Validação por tipo de conteúdo



The screenshot shows a Python IDE with a file named `validacao_conteudo1.py`. The code defines three strings and checks if they are numeric using `isalnum()`. The first string is entirely alphanumeric and returns `True`. The second string contains spaces and punctuation and returns `False`. The third string contains spaces and returns `False`. The output window at the bottom shows the execution path and the resulting boolean values.

```
1 texto = "EstaStringPossuiApenasLetrasENumeros1234567890"
2 print(texto.isalnum())
3
4 texto = "Esta String possui letras, números, espaços, vírgulas, ponto, 1, 2, 3, 4, 5."
5 print(texto.isalnum())
6
7 texto = "Esta string possui letras e espaços"
8 print(texto.isalnum())
9
```

Run: validacao_conteudo1 x

C:\Users\evaldo\AppData\Local\Programs\Python\Python37\python.exe C:/Evaldo/PCloud/Udemy/MeusCursos
True
False
False



Aprendendo ainda mais sobre Strings

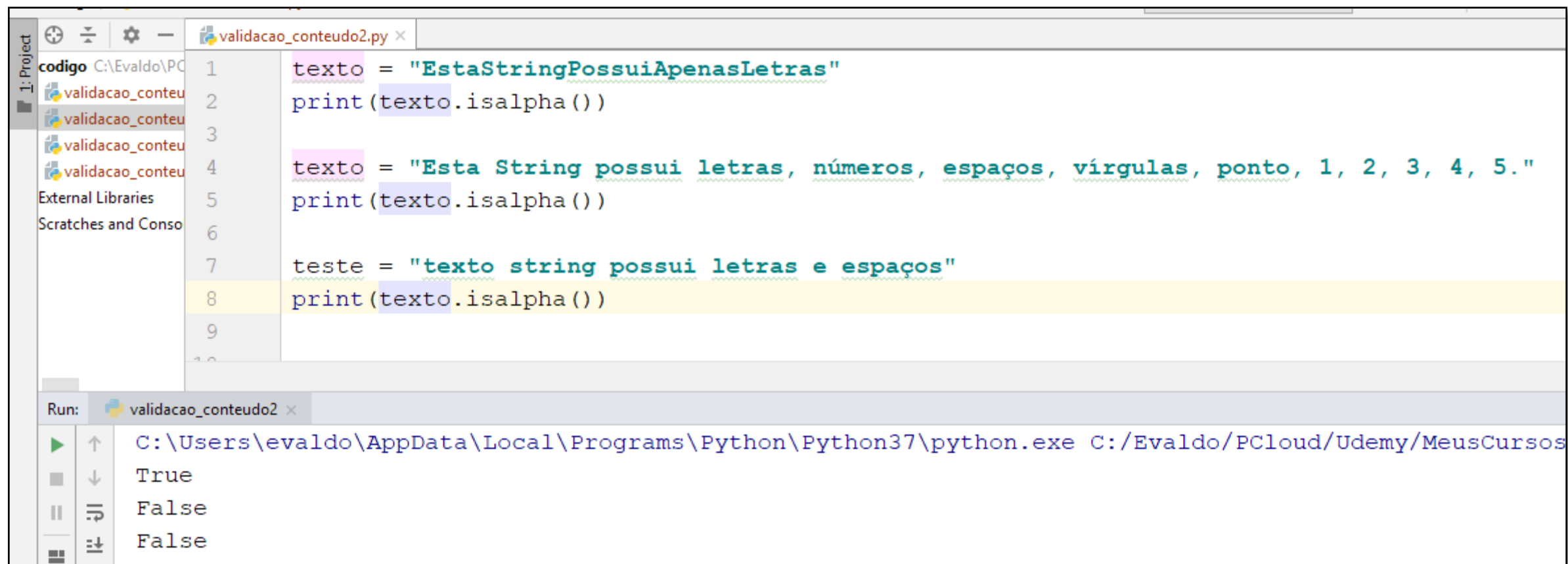
Validação por tipo de conteúdo

O método *isalpha* retorna verdadeiro se a string não for vazia e se todos seus caracteres forem letras (incluindo vogais acentuadas).



Aprendendo ainda mais sobre Strings

Validação por tipo de conteúdo



The screenshot shows a Python IDE with a file named `validacao_conteudo2.py`. The code defines three strings and checks if they are alphabetic using the `isalpha()` method. The first string is a single word, the second contains various punctuation and numbers, and the third contains spaces. The output window shows the results: `True` for the first, and `False` for the other two.

```
1 texto = "EstaStringPossuiApenasLetras"
2 print(texto.isalpha())
3
4 texto = "Esta String possui letras, números, espaços, vírgulas, ponto, 1, 2, 3, 4, 5."
5 print(texto.isalpha())
6
7 teste = "texto string possui letras e espaços"
8 print(texto.isalpha())
9
```

Run: validacao_conteudo2 x

```
C:\Users\evaldo\AppData\Local\Programs\Python\Python37\python.exe C:/Evaldo/PCloud/Udemy/MeusCursos
True
False
False
```

Aprendendo ainda mais sobre Strings

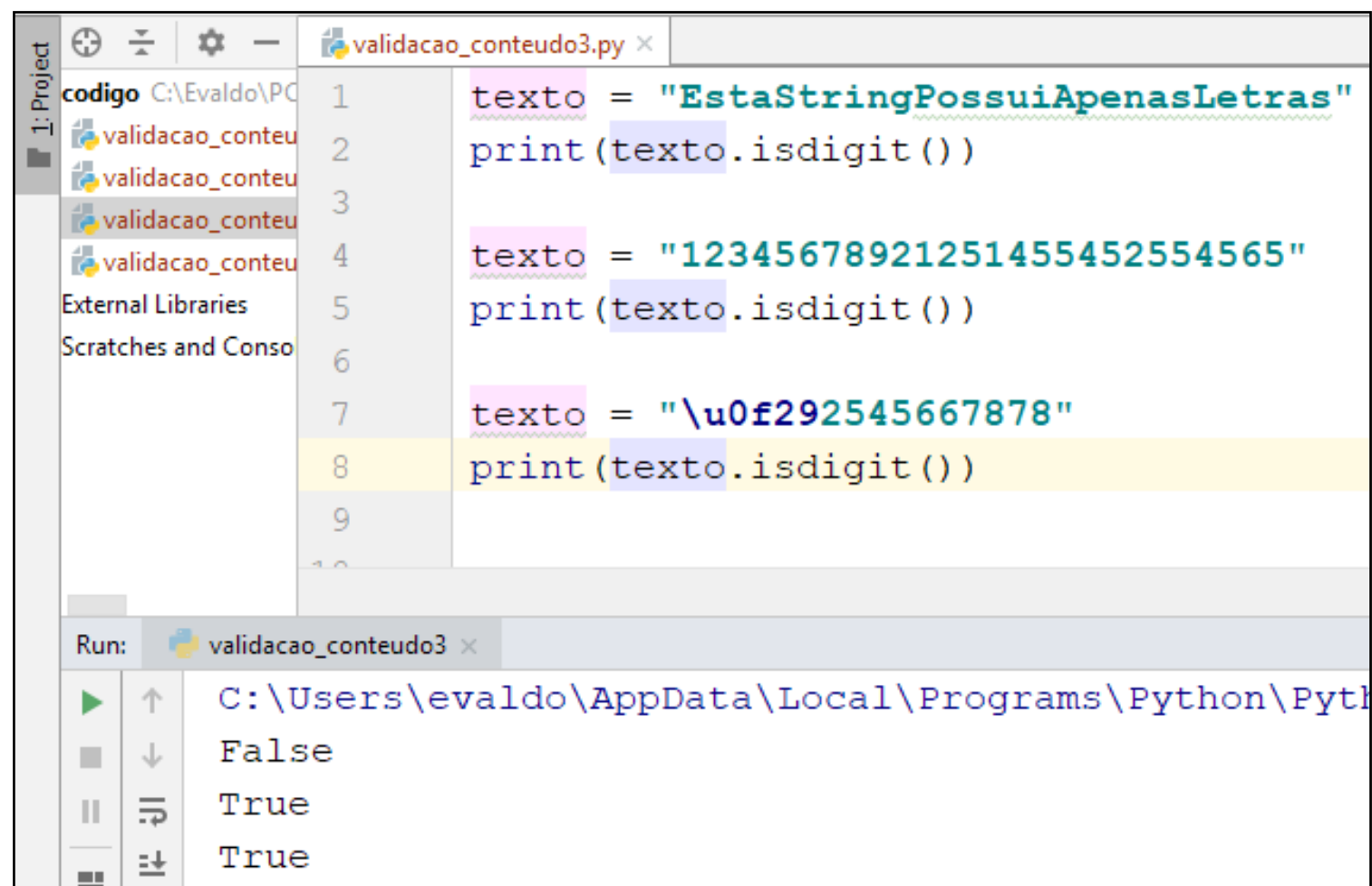
Validação por tipo de conteúdo

O método *isdigit* retorna verdadeiro se a string não for vazia e se todos seus caracteres forem numéricos em Unicode. Além dos números de 0 a 9, *isdigit* reconhece outros caracteres numéricos, como por exemplo, o 9 tibetano que é *Unicode* (\u0f29).



Aprendendo ainda mais sobre Strings

Validação por tipo de conteúdo



The screenshot shows a Python IDE with a file named `validacao_conteudo3.py`. The code defines three strings and checks if they are digits using the `isdigit()` method. The first string is "EstaStringPossuiApenasLetras", the second is "12345678921251455452554565", and the third is "\u0f292545667878". The output shows that the first two strings are not digits (False) and the third is a digit (True).

```
1 texto = "EstaStringPossuiApenasLetras"
2 print(texto.isdigit())
3
4 texto = "12345678921251455452554565"
5 print(texto.isdigit())
6
7 texto = "\u0f292545667878"
8 print(texto.isdigit())
9
```

Run: validacao_conteudo3 x

C:\Users\evaldo\AppData\Local\Programs\Python\Python38\python.exe

False

True

True



Aprendendo ainda mais sobre Strings

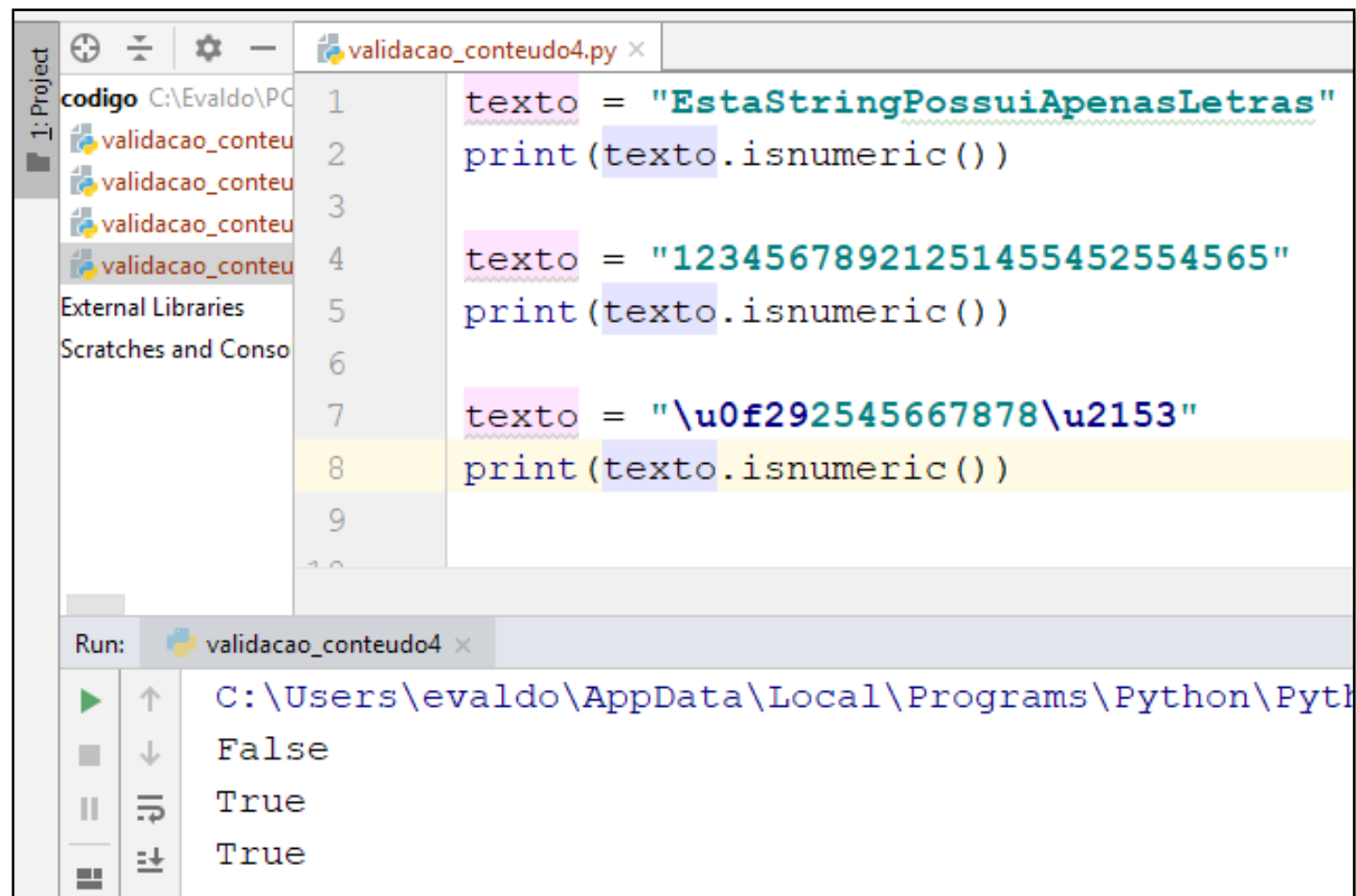
Validação por tipo de conteúdo

O método *isnumeric* retorna verdadeiro se a string não for vazia e se todos seus caracteres forem numéricos (Unicode também). Este método é mais abrangente que o *isdigit* e inclui dígitos e representações numéricas como frações, por exemplo, 1/3 (\u2153).



Aprendendo ainda mais sobre Strings

Validação por tipo de conteúdo



The screenshot shows a Python IDE with a file named `validacao_conteudo4.py`. The code defines three strings and checks if they are numeric using the `isnumeric()` method. The output window shows the results: `False` for the first string, `True` for the second, and `True` for the third.

```
1 texto = "EstaStringPossuiApenasLetras"  
2 print(texto.isnumeric())  
3  
4 texto = "12345678921251455452554565"  
5 print(texto.isnumeric())  
6  
7 texto = "\u0f292545667878\u2153"  
8 print(texto.isnumeric())  
9  
10
```

Run: validacao_conteudo4

```
C:\Users\evaldo\AppData\Local\Programs\Python\Python37-32\python.exe C:\Users\evaldo\AppData\Local\Programs\Python\Python37-32\python.exe C:\Users\evaldo\AppData\Local\Programs\Python\Python37-32\python.exe  
False  
True  
True
```



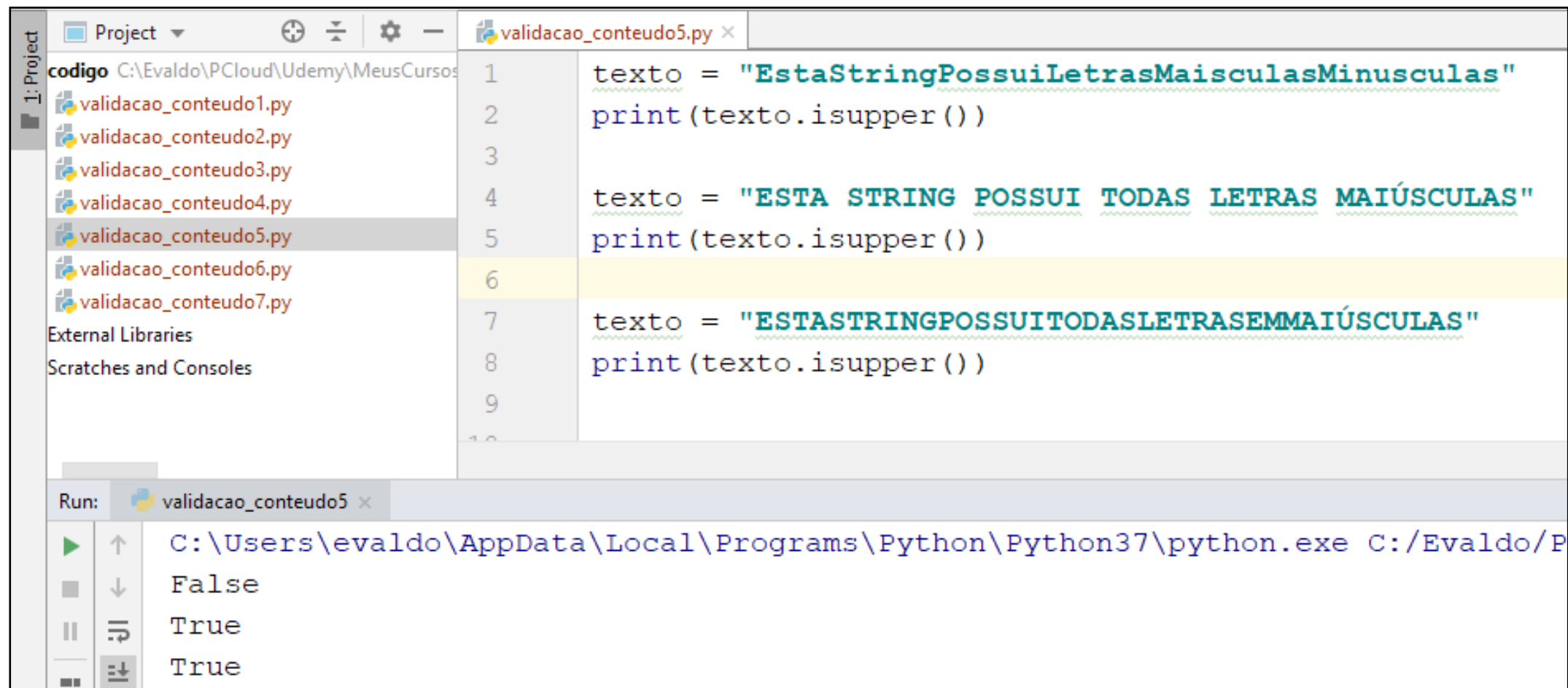
Aprendendo ainda mais sobre Strings

Validação por tipo de conteúdo

O método *isupper* verifica se todos os caracteres da string são maiúsculos.

Aprendendo ainda mais sobre Strings

Validação por tipo de conteúdo



The screenshot shows a Python IDE with a project named '1: Project'. The file explorer on the left lists several Python files: 'validacao_conteudo1.py' through 'validacao_conteudo7.py'. The main editor window displays the code in 'validacao_conteudo5.py'. The code defines three strings and checks if they are all uppercase using the `isupper()` method. The first string is 'EstaStringPossuiLetrasMaiusculasMinusculas', the second is 'ESTA STRING POSSUI TODAS LETRAS MAIÚSCULAS', and the third is 'ESTASTRINGPOSSUITODASLETRASEMMAIÚSCULAS'. The output window at the bottom shows the results of these checks: False for the first string, and True for the second and third strings.

```
1 texto = "EstaStringPossuiLetrasMaiusculasMinusculas"
2 print(texto.isupper())
3
4 texto = "ESTA STRING POSSUI TODAS LETRAS MAIÚSCULAS"
5 print(texto.isupper())
6
7 texto = "ESTASTRINGPOSSUITODASLETRASEMMAIÚSCULAS"
8 print(texto.isupper())
9
```

Run: validacao_conteudo5 x

C:\Users\evaldo\AppData\Local\Programs\Python\Python37\python.exe C:/Evaldo/P
False
True
True



Aprendendo ainda mais sobre Strings

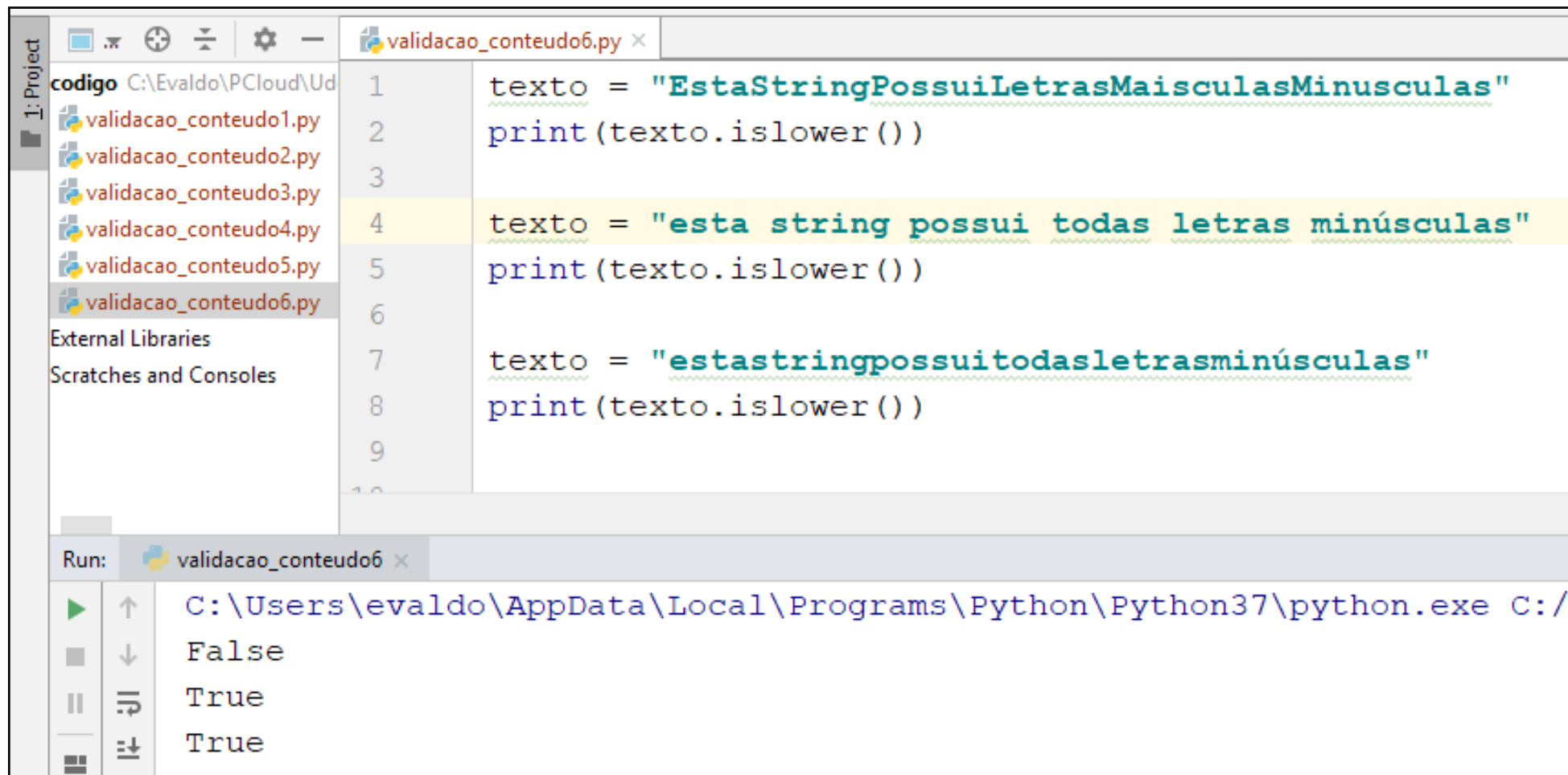
Validação por tipo de conteúdo

O método *islower* verifica se todos os caracteres da string são minúsculos.



Aprendendo ainda mais sobre Strings

Validação por tipo de conteúdo



The screenshot shows a Python IDE with a project named '1: Project'. The file explorer on the left lists several files: 'validacao_conteudo1.py', 'validacao_conteudo2.py', 'validacao_conteudo3.py', 'validacao_conteudo4.py', 'validacao_conteudo5.py', and 'validacao_conteudo6.py'. The active file is 'validacao_conteudo6.py', which contains the following code:

```
1 texto = "EstaStringPossuiLetrasMaisculasMinusculas"
2 print(texto.islower())
3
4 texto = "esta string possui todas letras minúsculas"
5 print(texto.islower())
6
7 texto = "estastringpossuitodasletrasminúsculas"
8 print(texto.islower())
9
```

The output window at the bottom shows the results of running the code:

```
Run: validacao_conteudo6 x
C:\Users\evaldo\AppData\Local\Programs\Python\Python37\python.exe C:/
False
True
True
```



Aprendendo ainda mais sobre Strings

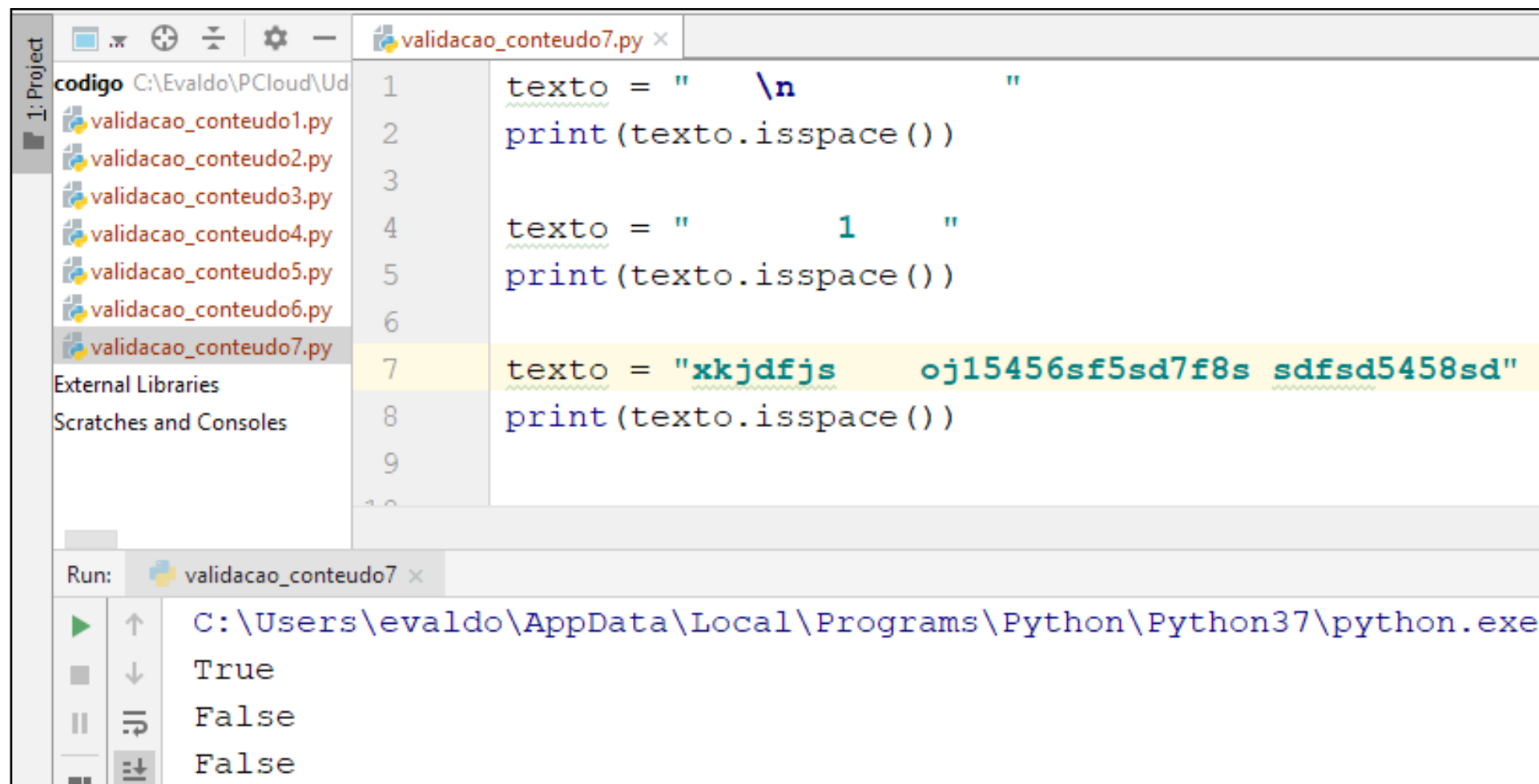
Validação por tipo de conteúdo

O método *isspace* verifica se a string contém apenas caracteres em branco, como espaços, marcas de tabulação (TAB), quebras de linha (LF) ou retorno de carro (CR).



Aprendendo ainda mais sobre Strings

Validação por tipo de conteúdo



The screenshot shows a Python IDE with a project named '1: Project'. The file explorer on the left lists several files: 'codigo C:\Evaldo\PCloud\Ud', 'validacao_conteudo1.py', 'validacao_conteudo2.py', 'validacao_conteudo3.py', 'validacao_conteudo4.py', 'validacao_conteudo5.py', 'validacao_conteudo6.py', and 'validacao_conteudo7.py'. The 'validacao_conteudo7.py' file is open in the editor, showing the following code:

```
1 texto = " \n "  
2 print(texto.isspace())  
3  
4 texto = " 1 "  
5 print(texto.isspace())  
6  
7 texto = "xkjdfjs oj15456sf5sd7f8s sdfsd5458sd"  
8 print(texto.isspace())  
9  
10
```

The output of the code is displayed in the 'Run' console at the bottom, showing the execution path and the results of the `isspace()` method calls:

```
Run: C:\Users\evaldo\AppData\Local\Programs\Python\Python37\python.exe  
True  
False  
False
```



Aprendendo ainda mais sobre Strings

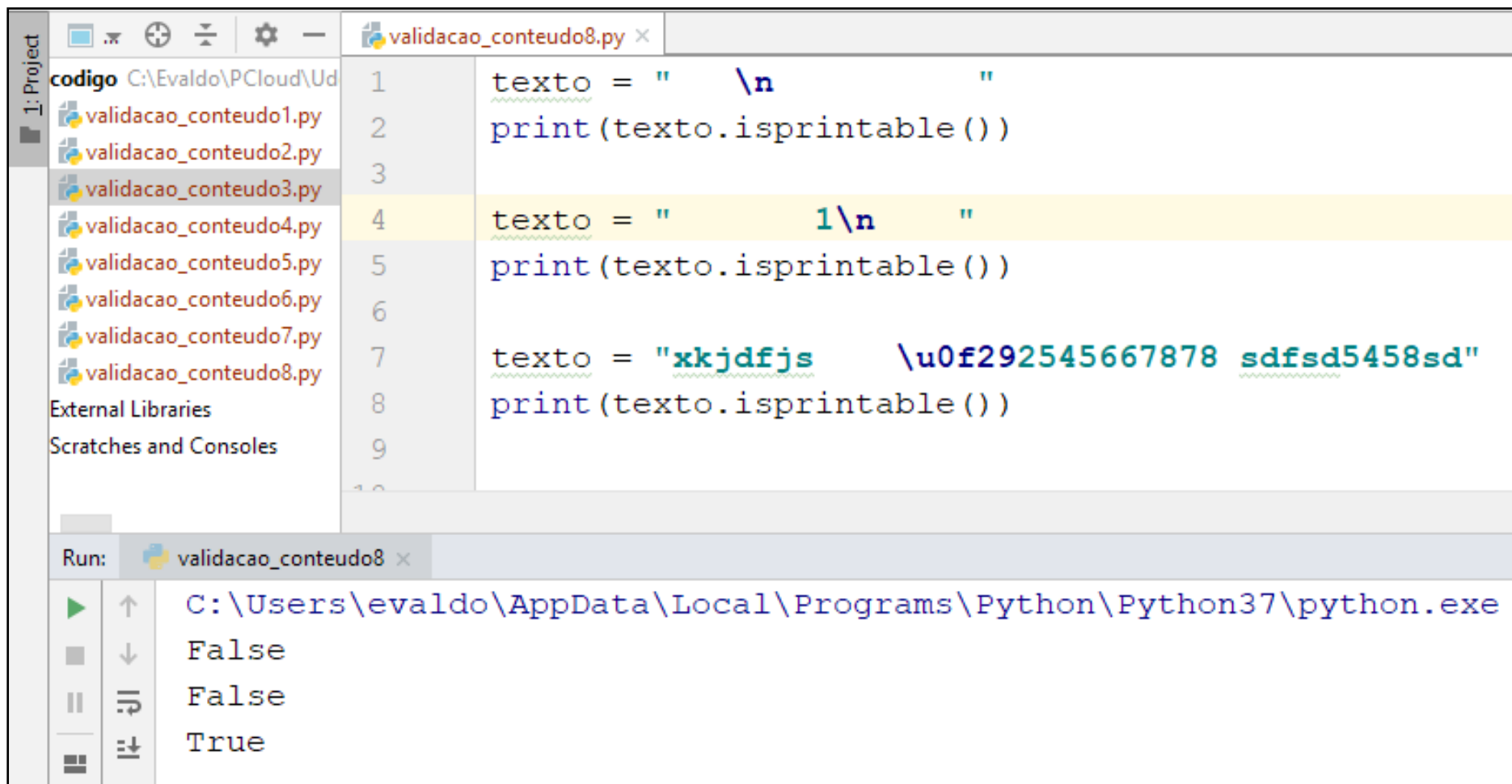
Validação por tipo de conteúdo

O método *isprintable* verifica se a string pode ser impressa na tela. Esse método retorna falso se algum caracter que não pode ser impresso for encontrado na string (por exemplo, o caracter de enter “\n”).



Aprendendo ainda mais sobre Strings

Validação por tipo de conteúdo



The screenshot displays the Wing IDE interface. On the left, a project explorer shows a directory named '1: Project' containing several files: 'codigo C:\Evaldo\PCloud\Ud', 'validacao_conteudo1.py', 'validacao_conteudo2.py', 'validacao_conteudo3.py', 'validacao_conteudo4.py', 'validacao_conteudo5.py', 'validacao_conteudo6.py', 'validacao_conteudo7.py', and 'validacao_conteudo8.py'. Below these are 'External Libraries' and 'Scratches and Consoles'. The main editor window shows the code for 'validacao_conteudo8.py' with line numbers 1 through 9. The code is as follows:

```
1 texto = " \n "  
2 print(texto.isprintable())  
3  
4 texto = " 1\n "  
5 print(texto.isprintable())  
6  
7 texto = "xkjdfjs \u0f292545667878 sdfsd5458sd"  
8 print(texto.isprintable())  
9
```

At the bottom, the 'Run' panel shows the execution of 'validacao_conteudo8'. The command used is 'C:\Users\evaldo\AppData\Local\Programs\Python\Python37\python.exe'. The output of the program is displayed in three lines: 'False', 'False', and 'True', corresponding to the three print statements in the code.

FIM

