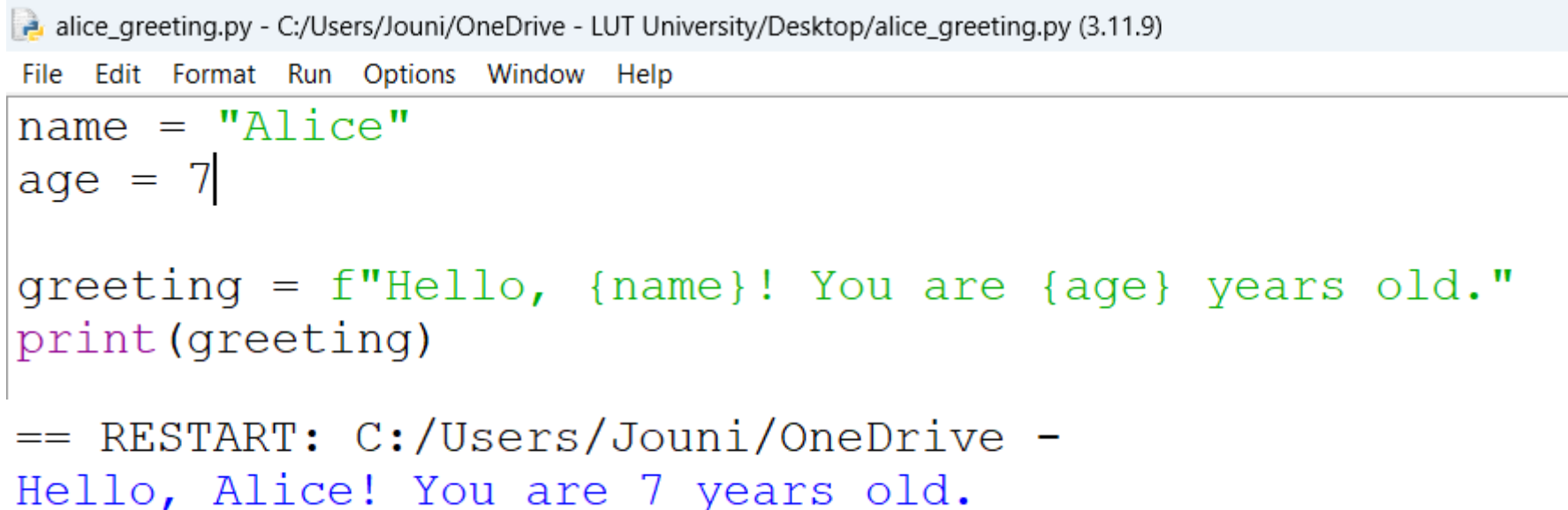


# f-strings

Week 7

# What are f-strings?

- f-strings are a simple way to format strings in Python.
- An **f-string** starts with an `f` (or `F`) followed by a pair of curly brackets `{}`.
- Inside the curly brackets, you can place expressions that will be evaluated and inserted into the string.

A screenshot of a Python IDE window titled 'alice\_greeting.py - C:/Users/Jouni/OneDrive - LUT University/Desktop/alice\_greeting.py (3.11.9)'. The window has a menu bar with 'File', 'Edit', 'Format', 'Run', 'Options', 'Window', and 'Help'. The code editor contains the following Python code:

```
name = "Alice"
age = 7

greeting = f"Hello, {name}! You are {age} years old."
print(greeting)
```

Below the code editor, the output of the program is displayed in a blue font:

```
== RESTART: C:/Users/Jouni/OneDrive -
Hello, Alice! You are 7 years old.
```

# Specifying decimal places:

f-string.py - C:/Users/Jouni/OneDrive - LUT University/Desktop/f-string.py (3.11.9)

File Edit Format Run Options Window Help

```
value = 123.456789
formatted = f"{value:.2f}"
print(formatted)
```

==== RESTART:

123.46

Notice that f-string rounds correctly!

# Adding commas as thousand separators

f-string.py - C:/Users/Jouni/OneDrive - LUT University/Desktop/f-string.py (3.11.9)

File Edit Format Run Options Window Help

```
value = 1234567.89
formatted = f"{value:,}"
print(formatted)
```

==== RESTART: C

1,234,567.89

# Pad and align decimal numbers

f-string.py - C:/Users/Jouni/OneDrive - LUT University/Desktop/f-string.py (3.11.9)

File Edit Format Run Options Window Help

```
value1 = 123.456
value2 = 98765.2345
```

```
# Right-aligned with a width of 10
```

```
print(f"{value1:10.2f}")
```

```
print(f"{value2:10.2f}")
```

```
# The number 10 specifies the total width of the
# formatted string, including the decimal point
# and the digits before and after it
```

```
= RESTART: C:
```

```
123.46
```

```
98765.23
```

# Formatting integers

f-string.py - C:/Users/Jouni/OneDrive - LUT University/Desktop/f-string.py (3.11.9)

File Edit Format Run Options Window Help

```
value = 123
```

```
print(f"{value:5d}") # Width 5 digits
```

```
print(f"{value:05d}") # Zero-padded to 5 digits
```

```
===== R
```

```
123
```

```
00123
```

# Binaries and hexadecimals

f-string.py - C:\Users\Jouni\OneDrive - LUT University\Desktop\Week7\f-string.py (3.11.9)

File Edit Format Run Options Window Help

```
value = 169
formatted_bin = f"{value:b}" # Binary
formatted_hex = f"{value:x}" # Hexadecimal

print(formatted_bin)
print(formatted_hex)
|
```

```
===== RESTA
1010|1001
a9
```