CoGrammar

# String Handling

SKILLS FOR LIFE
SKILLS BOOTCAMPS

Department for Education

# Data Science Lecture Housekeeping

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly. **(FBV: Mutual Respect.)**
- No question is daft or silly - **ask them!**
- There are **Q&A sessions** midway and at the end of the session, should you wish to ask any follow-up questions. Moderators are going to be answering questions as the session progresses as well.
- If you have any questions outside of this lecture, or that are not answered during this lecture, please do submit these for upcoming Open Classes. You can submit these questions here: **Open Class Questions**

# **Data Science Lecture Housekeeping** cont.

- For all **non-academic questions**, please submit a query: **www.hyperiondev.com/support**
- Report a **safeguarding** incident: **www.hyperiondev.com/safeguardreporting**
- We would love your **feedback** on lectures: **Feedback on Lectures**

# Lecture Objectives

- **A brief recap on Strings and String Methods.**
- **Examples of some more advanced cases of String manipulation / handling.**

# Remembering Strings

```
# We remember strings and how we can manipulate them?
word = "Hello There!"

idx = word[0 : 8 : 2] # [start : end : step]
print(idx)
```

# Remembering Strings

```
# There's also string methods
word = "do you enjoy Python yet?"

new_upper = word.upper()
new_lower = word.lower()
new_capital = word.capitalize()
```

# Remembering Strings

```
# More methods include :
new_strip = word.strip("?")
new_split = word.split(" ")
new_replace = word.replace(" ", "@")
```

# Formatting Strings

Instead of simply concatenating strings together, we have the ability to create more complex, formatted strings. With the power of this method we are able to create formatted strings by placing variables into placeholders.

# Formatting Strings

```
# Formatting Strings goes as follows :
name = "John"
surname = "Python"

print("My name is {} {} and I am a thorough enjoyer of {} \
".format(name, surname, surname))

# An alternate way of writing
print("My name is {0} {1} and I am a thorough enjoyer of {1} \
".format(name, surname))
```

# Format Examples

```
# We could even display decimal values!
value = 68.85673423

output = "Your total on your order will be : £{:.2f}"
print(output.format(value))
# Do keep in mind that the decimal function will round up if possible
```

# F-Strings

There is another means to format strings that was introduced, which is called Literal String Interpolation, or better known as f-strings.

# F-String Example

```
# F-Strings

name = "John"
surname = "Python"

print(f"My name is {name} {surname} and I am a thorough Python enjoyer.")
```

CoGrammar

# Q & A SECTION

Please use this time to ask any questions relating to the topic, should you have any.

# CoGrammar

## Thank you for joining!