

# Extracting date and time components

DATA ANALYSIS IN SPREADSHEETS



James Chapman

Curriculum Manager, DataCamp

# Data analysis process



Formulate  
Problem



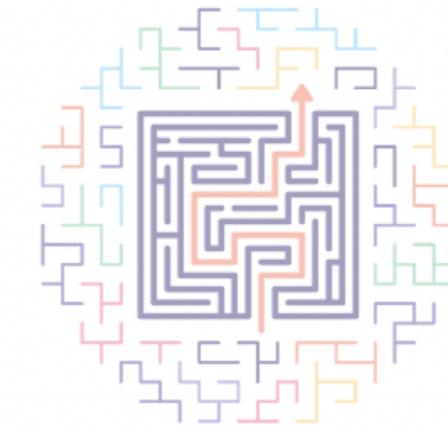
Collect and  
Store Data



Explore Data



Clean Data



Analyze Data



Present  
Findings

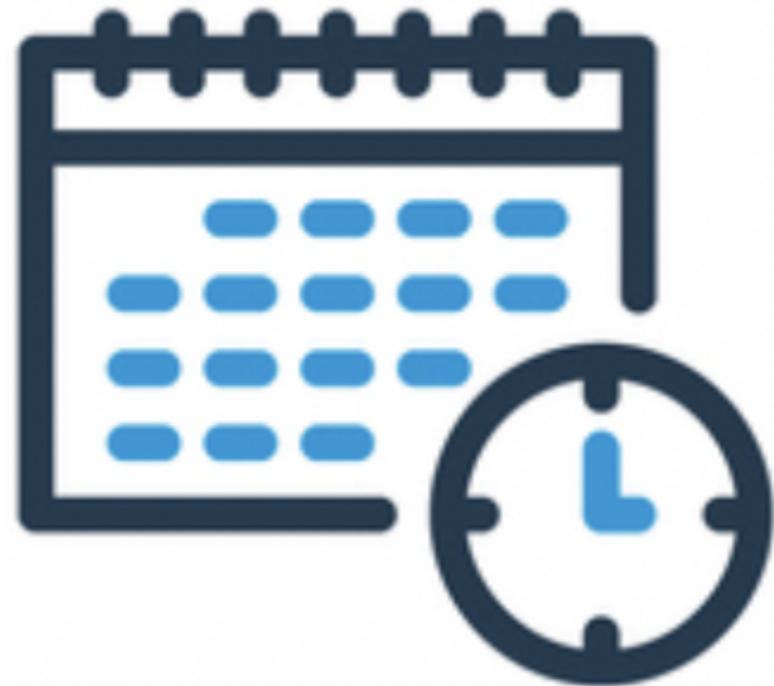
# Cleaning and preparing data

- **80/20 rule:** 80% cleaning, 20% analyzing
- A **clean dataset**...
  - can be easily processed during analysis
  - will return valid conclusions
  - save more time during analysis



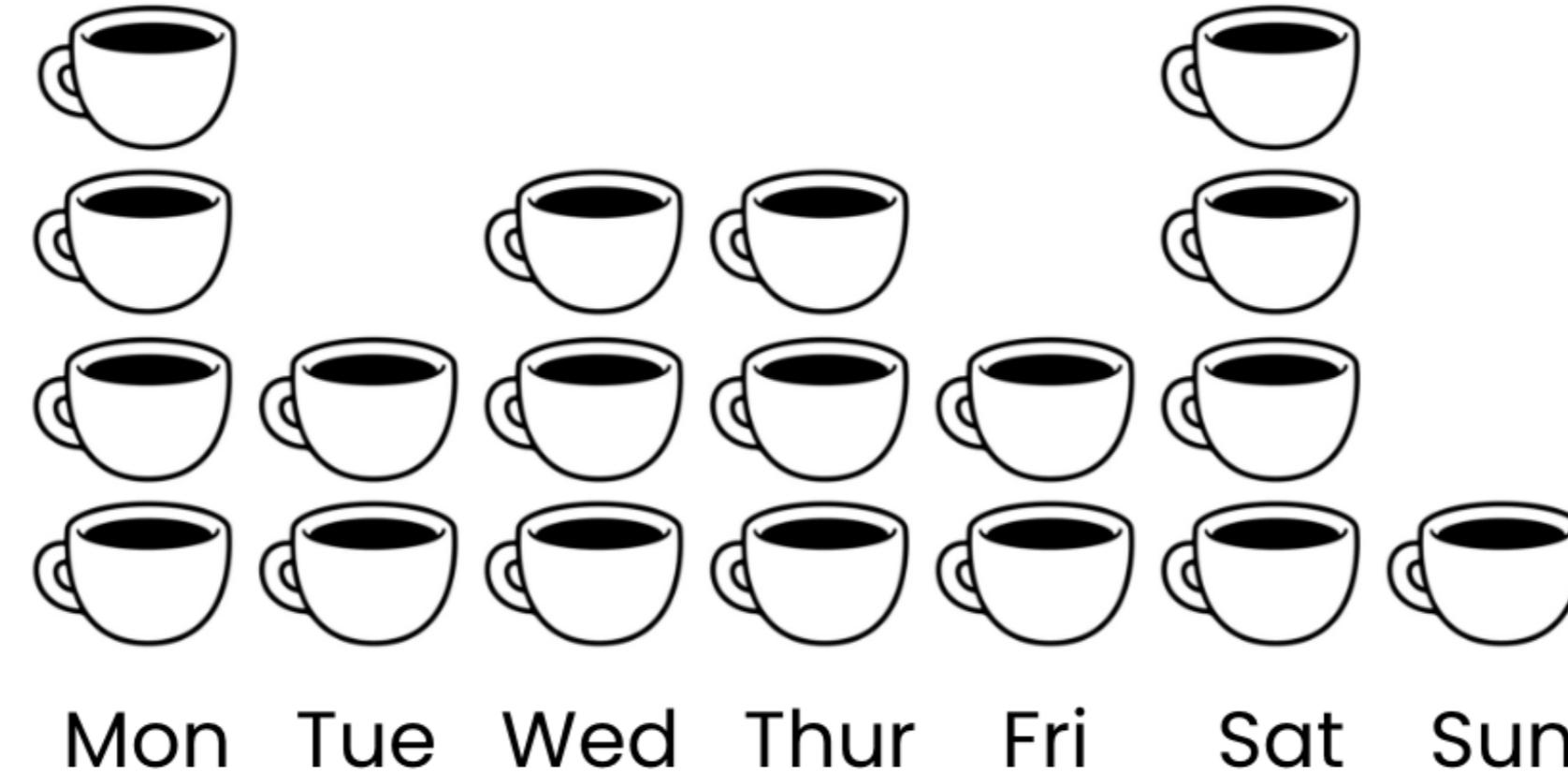
# Dates and times

- Collected for **measurements over time**
- **Continuous** data: can take *any* value
- **Discrete** data: can take one of a *finite* number of categories



# Discrete dates and times

**Question:** How do coffee purchases vary over days of the week?



# Extracting the year component

## Syntax:

- YEAR(date)

	A	B	C
1	Date	Year	
2	1969-07-16		
3	1969-11-14		
4	1971-01-31		
5	1971-07-26		
6	1972-04-16		
7	1972-12-07		

# Extracting the year component

## Syntax:

- YEAR(date)

	A	B	C
1	Date	Year	
2	1969-07-1?	=YEAR(A2)	
3	1969-11-14		
4	1971-01-31		
5	1971-07-26		
6	1972-04-16		
7	1972-12-07		

# Extracting the year component

## Syntax:

- YEAR(date)

	A	B	C
1	Date	Year	
2	1969-07-16	1969	
3	1969-11-14		
4	1971-01-31		
5	1971-07-26		
6	1972-04-16		
7	1972-12-07		

# Extracting the year component

## Syntax:

- `YEAR(date)`

	A	B	C
1	<b>Date</b>	<b>Year</b>	
2	1969-07-16	1969	
3	1969-11-14	1969	
4	1971-01-31	1971	
5	1971-07-26	1971	
6	1972-04-16	1972	
7	1972-12-07	1972	

# Extracting the month component

## Syntax:

- `MONTH(date)`

	A	B	C
1	Date	Month	
2	1969-07-16	= MONTH(A2)	
3	1969-11-14		
4	1971-01-31		
5	1971-07-26		
6	1972-04-16		
7	1972-12-07		

# Extracting the month component

## Syntax:

- `MONTH(date)`

Other desirable month formats:

- E.g., Jan and January

	A	B	C
1	Date	Month	
2	1969-07-16	7	
3	1969-11-14	11	
4	1971-01-31	1	
5	1971-07-26	7	
6	1972-04-16	4	
7	1972-12-07	12	

# Extracting the short month name component

## Syntax:

- `TEXT(number, format)`

	A	B	C
1	Date	Month	
2	1969-07-1?	= TEXT(A2	
3	1969-11-14		
4	1971-01-31		
5	1971-07-26		
6	1972-04-16		
7	1972-12-07		

# Extracting the short month name component

## Syntax:

- `TEXT(number, format)`
- `format → "mmm"`

	A	B	C
1	Date	Month	
2	1969-07-1?	= TEXT(A2, "mmm")	
3	1969-11-14		
4	1971-01-31		
5	1971-07-26		
6	1972-04-16		
7	1972-12-07		

# Extracting the short month name component

## Syntax:

- `TEXT(number, format)`
- `format → "mmm"`

	A	B	C
1	<b>Date</b>	<b>Month</b>	
2	1969-07-16	Jul	
3	1969-11-14		
4	1971-01-31		
5	1971-07-26		
6	1972-04-16		
7	1972-12-07		

# Extracting the short month name component

## Syntax:

- `TEXT(number, format)`
- `format → "mmm"`

	A	B	C
1	<b>Date</b>	<b>Month</b>	
2	1969-07-16	Jul	
3	1969-11-14	Nov	
4	1971-01-31	Jan	
5	1971-07-26	Jul	
6	1972-04-16	Apr	
7	1972-12-07	Dec	

# Extracting the long month name component

## Syntax:

- `TEXT(number, format)`
- `format → "mmmm"`

	A	B	C
1	Date	Month	
2	1969-07-16	= TEXT(A2, "mmmm")	
3	1969-11-14		
4	1971-01-31		
5	1971-07-26		
6	1972-04-16		
7	1972-12-07		

# Extracting the long month name component

## Syntax:

- `TEXT(number, format)`
- `format → "mmmm"`

	A	B	C
1	Date	Month	
2	1969-07-16	July	
3	1969-11-14	November	
4	1971-01-31	January	
5	1971-07-26	July	
6	1972-04-16	April	
7	1972-12-07	December	

# Extracting the weekday component

## Syntax:

- `WEEKDAY(date, [type])`
  - `type` : *the numbering system to use*
    - `1` (default): Start Sunday = 1
    - `2` : Start Monday = 1
    - `3` : Start Monday = 0

	A	B	C
1	Date	Weekday	
2	1969-07-1?	= WEEKDAY(A2)	
3	1969-11-14		
4	1971-01-31		
5	1971-07-26		
6	1972-04-16		
7	1972-12-07		

# Extracting the weekday component

## Syntax:

- `WEEKDAY(date, [type])`
  - `type` : *the numbering system to use*
    - `1` (default): Start Sunday = 1
    - `2` : Start Monday = 1
    - `3` : Start Monday = 0

	A	B	C
1	Date	Weekday	
2	1969-07-16	4	
3	1969-11-14	6	
4	1971-01-31	1	
5	1971-07-26	2	
6	1972-04-16	1	
7	1972-12-07	5	

# Extracting the short weekday name

## Syntax:

- `TEXT(number, format)`
- `format → "ddd"`

	B2	fx	= TEXT(A2, "ddd")
1	Date	Weekday	
2	1969-07-16	Wed	
3	1969-11-14	Fri	
4	1971-01-31	Sun	
5	1971-07-26	Mon	
6	1972-04-16	Sun	
7	1972-12-07	Thu	

# Extracting the long weekday name

## Syntax:

- `TEXT(number, format)`
- `format → "dddd"`

B2  $\downarrow$   $fx$  = `TEXT(A2, "dddd")`

	A	B	C
1	Date	Weekday	
2	1969-07-16	Wednesday	
3	1969-11-14	Friday	
4	1971-01-31	Sunday	
5	1971-07-26	Monday	
6	1972-04-16	Sunday	
7	1972-12-07	Thursday	

# Extracting time components

Also many functions to extract **time components**:

- HOUR(time)
- MINUTE(time)
- SECOND(time)



# **Let's practice!**

**DATA ANALYSIS IN SPREADSHEETS**

# Calculating date intervals

DATA ANALYSIS IN SPREADSHEETS



**James Chapman**

Curriculum Manager, DataCamp

# Recap...

Extract **date** components:

- YEAR()
- MONTH()
- WEEKDAY()
- TEXT()

Extract **time** components:

- HOUR()
- MINUTE()
- SECOND()

This lesson: calculate **durations!**

# When is now?

- Comparing to the **present date**



- Comparing to the **present time**



# TODAY()

= TODAY()

9/26/2022

- Spreadsheet is refreshed...

9/27/2022

# NOW()

= NOW()

9/26/2022 3:24:04

- Spreadsheet is refreshed...

9/26/2022 4:24:04

# Date arithmetic

	A	B	C
1	<b>Date</b>	<b>Interval</b>	
2	1969-07-16		
3	1969-11-14		
4	1971-01-31		
5	1971-07-26		
6	1972-04-16		
7	1972-12-07		
8			

# Date arithmetic

	A	B	C
1	Date	Interval	
2	1969-07-16		
3	1969-11-1	? = A3 - A2	
4	1971-01-31		
5	1971-07-26		
6	1972-04-16		
7	1972-12-07		
8			

# Date arithmetic

	A	B	C
1	Date	Interval	
2	1969-07-16		
3	1969-11-14	121	
4	1971-01-31		
5	1971-07-26		
6	1972-04-16		
7	1972-12-07		
8			

- Subtracting dates returns the interval in **days**

# Date arithmetic

	A	B	C
1	Date	Interval	
2	1969-07-16		
3	1969-11-14	121	
4	1971-01-31	443	
5	1971-07-26	176	
6	1972-04-16	265	
7	1972-12-07	235	
8			

- Subtracting dates returns the interval in **days**

# DATEDIF()

## Syntax:

- `DATEDIF(start_date, end_date, unit)`
  - `end_date > start_date`
  - `unit` : `"Y"` , `"M"` , `"D"` , ...

B3	fx	=DATEDIF(A2, A3, "M")
1	Date	Interval
2	1969-07-16	
3	1969-11-14	3
4	1971-01-31	14
5	1971-07-26	5
6	1972-04-16	8
7	1972-12-07	7
8		

# DATEDIF()

## Syntax:

- `DATEDIF(start_date, end_date, unit)`
  - `end_date > start_date`
  - `unit` : `"Y"`, `"M"`, `"D"`, ...
- Results are **chopped**

B3	▼	fx	=DATEDIF(A2, A3, "Y")
	A	B	C
1	Date	Interval	
2	1969-07-16		
3	1969-11-14	0	
4	1971-01-31	1	
5	1971-07-26	0	
6	1972-04-16	0	
7	1972-12-07	0	
8			

# DATEDIF() and TODAY()

## Syntax:

- `DATEDIF(start_date, end_date, unit)`
  - `end_date` > `start_date`
  - `unit` : `"Y"`, `"M"`, `"D"`, ...
- Results are **chopped**

	A	B	C
1	Date	Interval	
2	1969-07-1?	= DATEDIF(A2, TODAY(), "Y")	
3	1969-11-14		
4	1971-01-31		
5	1971-07-26		
6	1972-04-16		
7	1972-12-07		
8			

# DATEDIF() and TODAY()

## Syntax:

- `DATEDIF(start_date, end_date, unit)`
  - `end_date > start_date`
  - `unit` : `"Y"` , `"M"` , `"D"` , ...
- Results are **chopped**

	A	B	C
1	Date	Interval	
2	1969-07-16		53
3	1969-11-14		53
4	1971-01-31		51
5	1971-07-26		51
6	1972-04-16		50
7	1972-12-07		50
8			

# **Let's practice!**

**DATA ANALYSIS IN SPREADSHEETS**

# Cleaning text data

DATA ANALYSIS IN SPREADSHEETS



**James Chapman**

Curriculum Manager, DataCamp

# Cleaning text data

- Who's your favorite DataCamp instructor?

james chapman

James

Jamez chapnam

James Chapman



# Changing cases - PROPER()

- **Proper case:**
  - Example: *This Is Proper Case*
  - Used for names and addresses

```
= PROPER("neil armstrong")
```

```
Neil Armstrong
```

# Changing cases - LOWER()

- **Lowercase:**
  - Example: *this is lowercase*

```
= LOWER("Neil Armstrong")
```

```
neil armstrong
```

# Changing cases - UPPER()

- **Uppercase:**
  - Example: *THIS IS UPPERCASE*

```
= UPPER("Neil Armstrong")
```

```
NEIL ARMSTRONG
```

# Removing whitespace

- **Extra whitespace:**
  - *Leading* - space before text
  - *Trailing* - space after text
  - *Repeated* - >1 space between characters

```
= TRIM("    Neil    Armstrong    ")
```

```
Neil Armstrong
```

# Combining text data

## Syntax:

```
CONCATENATE(string1, [string2, ...])
```

	A	B	C	D
1	<b>First Name</b>	<b>Last Name</b>	<b>Full Name</b>	
2	Neil	Armstrong		
3	Edwin	Aldrin		
4	Michael	Collins		
5	Charles	Conrad		
6	Alan	Bean		
7	Richard	Gordon		
8				

# Combining text data

Syntax:

```
CONCATENATE(string1, [string2, ...])
```

	A	B	C	D
1	<b>First Name</b>	<b>Last Name</b>	<b>Full Name</b>	
2	Neil	Armstrong	= CONCATENATE(A2, B2)	
3	Edwin	Aldrin		
4	Michael	Collins		
5	Charles	Conrad		
6	Alan	Bean		
7	Richard	Gordon		
8				

# Combining text data

Syntax:

```
CONCATENATE(string1, [string2, ...])
```

	A	B	C	D
1	<b>First Name</b>	<b>Last Name</b>	<b>Full Name</b>	
2	Neil	Armstrong	NeilArmstrong	
3	Edwin	Aldrin		
4	Michael	Collins		
5	Charles	Conrad		
6	Alan	Bean		
7	Richard	Gordon		
8				

# Combining text data

Syntax:

```
CONCATENATE(string1, [string2, ...])
```

	A	B	C	D
1	<b>First Name</b>	<b>Last Name</b>	<b>Full Name</b>	
2	Neil	Armstrong	= CONCATENATE(A2, " ", B2)	
3	Edwin	Aldrin		
4	Michael	Collins		
5	Charles	Conrad		
6	Alan	Bean		
7	Richard	Gordon		
8				

# Combining text data

Syntax:

```
CONCATENATE(string1, [string2, ...])
```

	A	B	C	D
1	<b>First Name</b>	<b>Last Name</b>	<b>Full Name</b>	
2	Neil	Armstrong	Neil Armstrong	
3	Edwin	Aldrin		
4	Michael	Collins		
5	Charles	Conrad		
6	Alan	Bean		
7	Richard	Gordon		
8				

# Combining text data

Syntax:

```
CONCATENATE(string1, [string2, ...])
```

	A	B	C	D
1	<b>First Name</b>	<b>Last Name</b>	<b>Full Name</b>	
2	Neil	Armstrong	Neil Armstrong	
3	Edwin	Aldrin	Edwin Aldrin	
4	Michael	Collins	Michael Collins	
5	Charles	Conrad	Charles Conrad	
6	Alan	Bean	Alan Bean	
7	Richard	Gordon	Richard Gordon	
8				

# Combining text data - email addresses

Example:

- **firstname.lastname@nasa.com**

	A	B	C	D	E
1	First Name	Last Name	Email		
2	Neil	Armstrong			
3	Edwin	Aldrin			
4	Michael	Collins			
5	Charles	Conrad			
6	Alan	Bean			
7	Richard	Gordon			
8					

# Combining text data - email addresses

Example:

- **firstname.lastname@nasa.com**

	A	B	C	D	E
1	First Name	Last Name	Email		
2	Neil	Armstrong	? = CONCATENATE(LOWER(A2),		
3	Edwin	Aldrin			
4	Michael	Collins			
5	Charles	Conrad			
6	Alan	Bean			
7	Richard	Gordon			
8					

# Combining text data - email addresses

Example:

- **firstname.lastname@nasa.com**

	A	B	C	D	E
1	First Name	Last Name	Email		
2	Neil	Armstrong	? = CONCATENATE(LOWER(A2), ". ",		
3	Edwin	Aldrin			
4	Michael	Collins			
5	Charles	Conrad			
6	Alan	Bean			
7	Richard	Gordon			
8					

# Combining text data - email addresses

Example:

- **firstname.lastname@nasa.com**

	A	B	C	D	E
1	First Name	Last Name	Email		
2	Neil	Armstrong	? = CONCATENATE(LOWER(A2), ". ", LOWER(B2)),		
3	Edwin	Aldrin			
4	Michael	Collins			
5	Charles	Conrad			
6	Alan	Bean			
7	Richard	Gordon			
8					

# Combining text data - email addresses

Example:

- **firstname.lastname@nasa.com**

	A	B	C	D	E
1	First Name	Last Name	Email		
2	Neil	Armstrong	? = CONCATENATE(LOWER(A2), ".", LOWER(B2), "@nasa.com")		
3	Edwin	Aldrin			
4	Michael	Collins			
5	Charles	Conrad			
6	Alan	Bean			
7	Richard	Gordon			
8					

# Combining text data - email addresses

Example:

- **firstname.lastname@nasa.com**

	A	B	C	D	E
1	First Name	Last Name	Email		
2	Neil	Armstrong	neil.armstrong@nasa.com		
3	Edwin	Aldrin			
4	Michael	Collins			
5	Charles	Conrad			
6	Alan	Bean			
7	Richard	Gordon			
8					

# Combining text data - email addresses

Example:

- **firstname.lastname@nasa.com**

	A	B	C	D	E
1	First Name	Last Name	Email		
2	Neil	Armstrong	neil.armstrong@nasa.com		
3	Edwin	Aldrin	edwin.aldrin@nasa.com		
4	Michael	Collins	michael.collins@nasa.com		
5	Charles	Conrad	charles.conrad@nasa.com		
6	Alan	Bean	alan.bean@nasa.com		
7	Richard	Gordon	richard.gordon@nasa.com		
8					

# **Let's practice!**

**DATA ANALYSIS IN SPREADSHEETS**

# Manipulating text data

DATA ANALYSIS IN SPREADSHEETS



**James Chapman**

Curriculum Manager, DataCamp

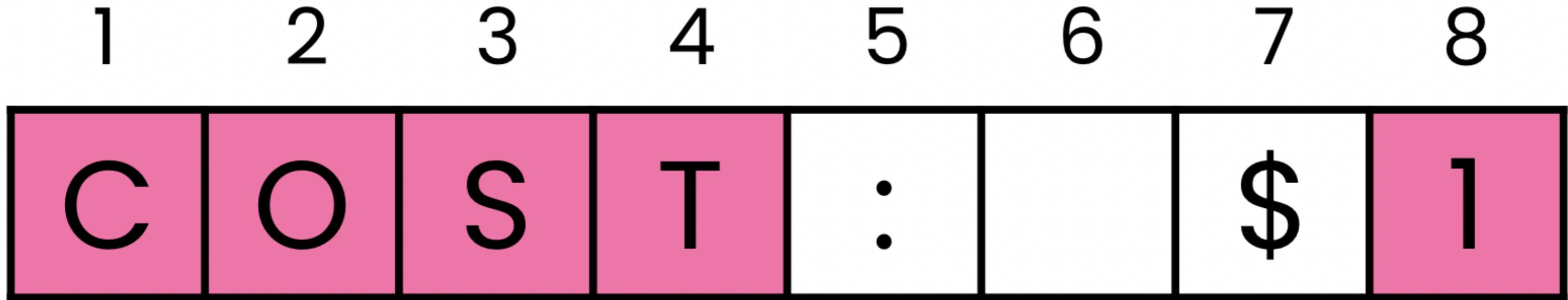
# How long is a string?

- Example: COST: \$1

1	2	3	4	5	6	7	8
C	O	S	T	:		\$	1

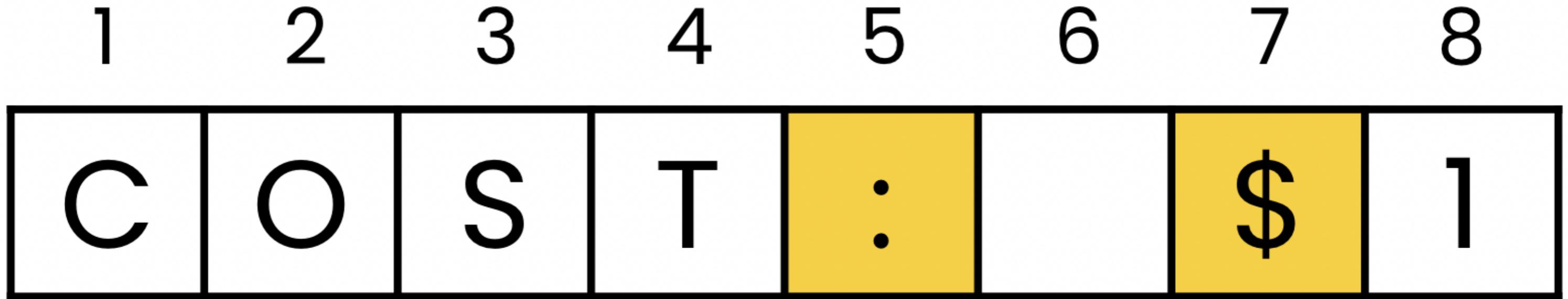
# How long is a string?

- Example: COST: \$1



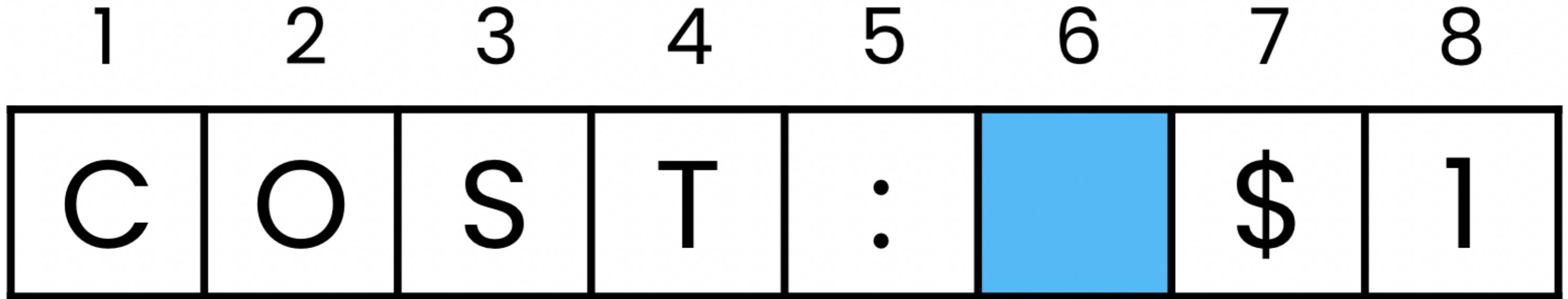
# How long is a string?

- Example: COST: \$1



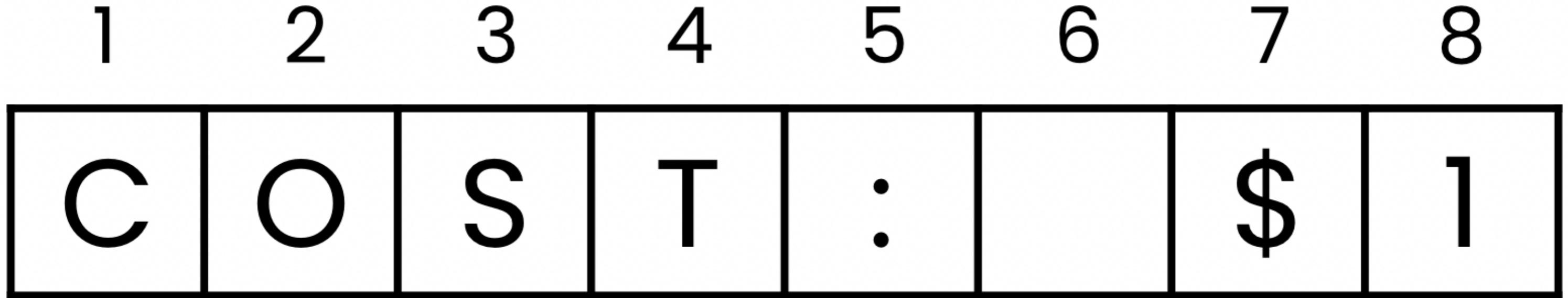
# How long is a string?

- Example: COST: \$1



# How long is a string?

- Example: COST: \$1



```
= LEN("COST: $1")
```

8

# Searching for characters

## Syntax:

```
SEARCH(search_for, text_to_search, [starting_at])
```

- `search_for` : String to search for
- `text_to_search` : Text to search through
- `starting_at` (default = 1): Index to start at

# Searching for characters

Syntax:

```
SEARCH(search_for, text_to_search, [starting_at])
```

Example: Find \$ in COST: \$1

```
= SEARCH("$", "COST: $1")
```

# Extracting text - LEFT() and RIGHT()

Syntax:

```
LEFT(string, [number_of_characters])
```

Syntax:

```
RIGHT(string, [number_of_characters])
```

Example:

```
= LEFT("COST: $1", 4)
```

COST

Example:

```
= RIGHT("COST: $1", 2)
```

\$1

# Bringing it together

Example: Extract the state from the location data

	A	B	C	D
1	<b>Location</b>	<b>State</b>		
2	Orlando,FL			
3	Arlington,VA			
4	Nottingham,England			
5	Vancouver,BC			
6	Los Gatos,CA			
7	Lentate sul Seveso,Lombardy			
8	Phoenix,AZ			

# Bringing it together

Example: Extract the state from the location data

	A	B	C	D
1	<b>Location</b>	<b>State</b>		
2	Orlando,FL	? = RIGHT(		
3	Arlington,VA			
4	Nottingham,England			
5	Vancouver,BC			
6	Los Gatos,CA			
7	Lentate sul Seveso,Lombardy			
8	Phoenix,AZ			

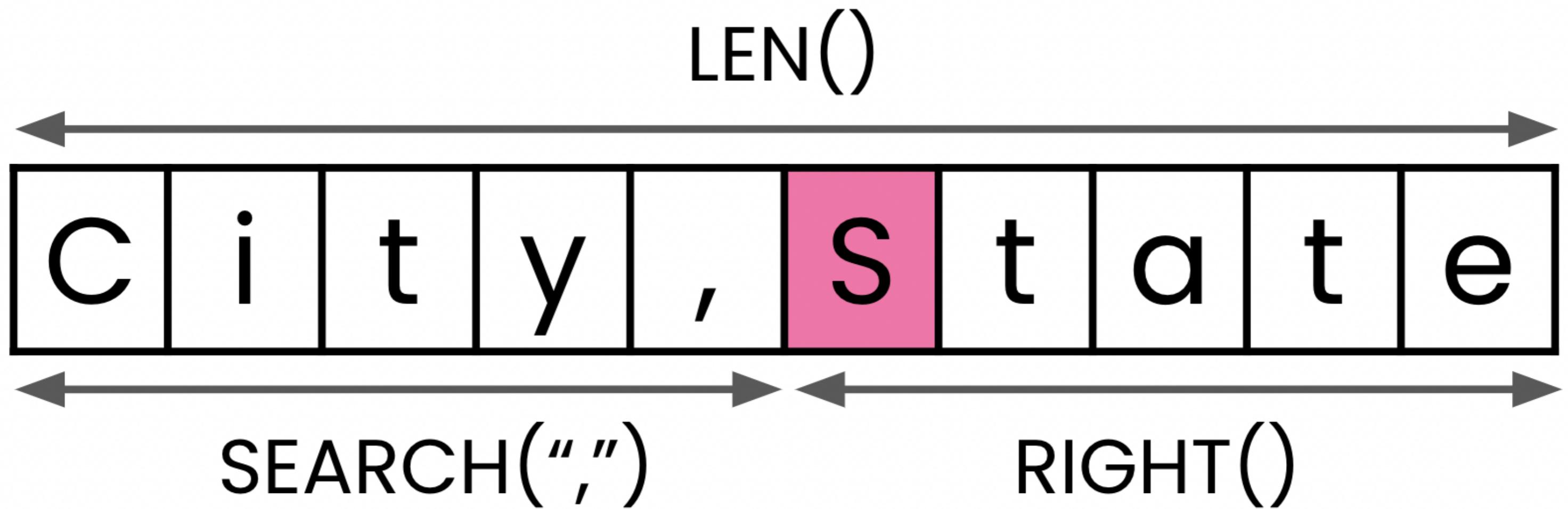
# Bringing it together

## Syntax:

```
RIGHT(string, [number_of_characters])
```

	A	B	C	D
1	<b>Location</b>	<b>State</b>		
2	Orlando,FL	? = RIGHT(A2,		
3	Arlington,VA			
4	Nottingham,England			
5	Vancouver,BC			
6	Los Gatos,CA			
7	Lentate sul Seveso,Lombardy			
8	Phoenix,AZ			

# Bringing it together



# Bringing it together

## Syntax:

```
SEARCH(search_for, text_to_search, [starting_at])
```

	A	B	C	D
1	<b>Location</b>	<b>State</b>		
2	Orlando,FL	? = RIGHT(A2, LEN(A2)-SEARCH(", ", A2))		
3	Arlington,VA			
4	Nottingham,England			
5	Vancouver,BC			
6	Los Gatos,CA			
7	Lentate sul Seveso,Lombardy			
8	Phoenix,AZ			

# Bringing it together

## Syntax:

```
SEARCH(search_for, text_to_search, [starting_at])
```

	A	B	C	D
1	<b>Location</b>	<b>State</b>		
2	Orlando, FL	FL		
3	Arlington, VA			
4	Nottingham, England			
5	Vancouver, BC			
6	Los Gatos, CA			
7	Lentate sul Seveso, Lombardy			
8	Phoenix, AZ			

# Bringing it together

## Syntax:

```
SEARCH(search_for, text_to_search, [starting_at])
```

	A	B	C	D
1	<b>Location</b>	<b>State</b>		
2	Orlando, FL	FL		
3	Arlington, VA	VA		
4	Nottingham, England	England		
5	Vancouver, BC	BC		
6	Los Gatos, CA	CA		
7	Lentate sul Seveso, Lombardy	Lombardy		
8	Phoenix, AZ	AZ		

# Substituting characters

## Syntax:

```
SUBSTITUTE(text_to_search, search_for, replace_with, [occurrence_number])
```

- `text_to_search` : the text to search through
- `search_for` : the string to search for
- `replace_with` : the replacement string
- `occurrence_number` : which occurrence should be substituted

# Substituting characters

## Syntax:

```
SUBSTITUTE(text_to_search, search_for, replace_with, [occurrence_number])
```

## Example:

```
= SUBSTITUTE("I like DataCamp!", "like", "love")
```

I love DataCamp!

# **Let's practice!**

**DATA ANALYSIS IN SPREADSHEETS**