

Parsing Log Files



Maaike van Putten

Trainer & Software Developer

@brightboost | www.brightboost.nl

A woman with dark skin, curly hair, and round glasses is looking down at a laptop screen. She is wearing a light-colored button-down shirt. The background is blurred, showing shelves and books.

Log files can grow really big...

Overview



We'll use Python Regex to do the following:

- Parse log files for data types
- Search log files for error codes
- Live logging filter for errors
- Split entries of a log file







Automate Log Handling

Regex can be used to assist in the automated handling of log files by recognizing different patterns in the ever-growing log file.



Demo

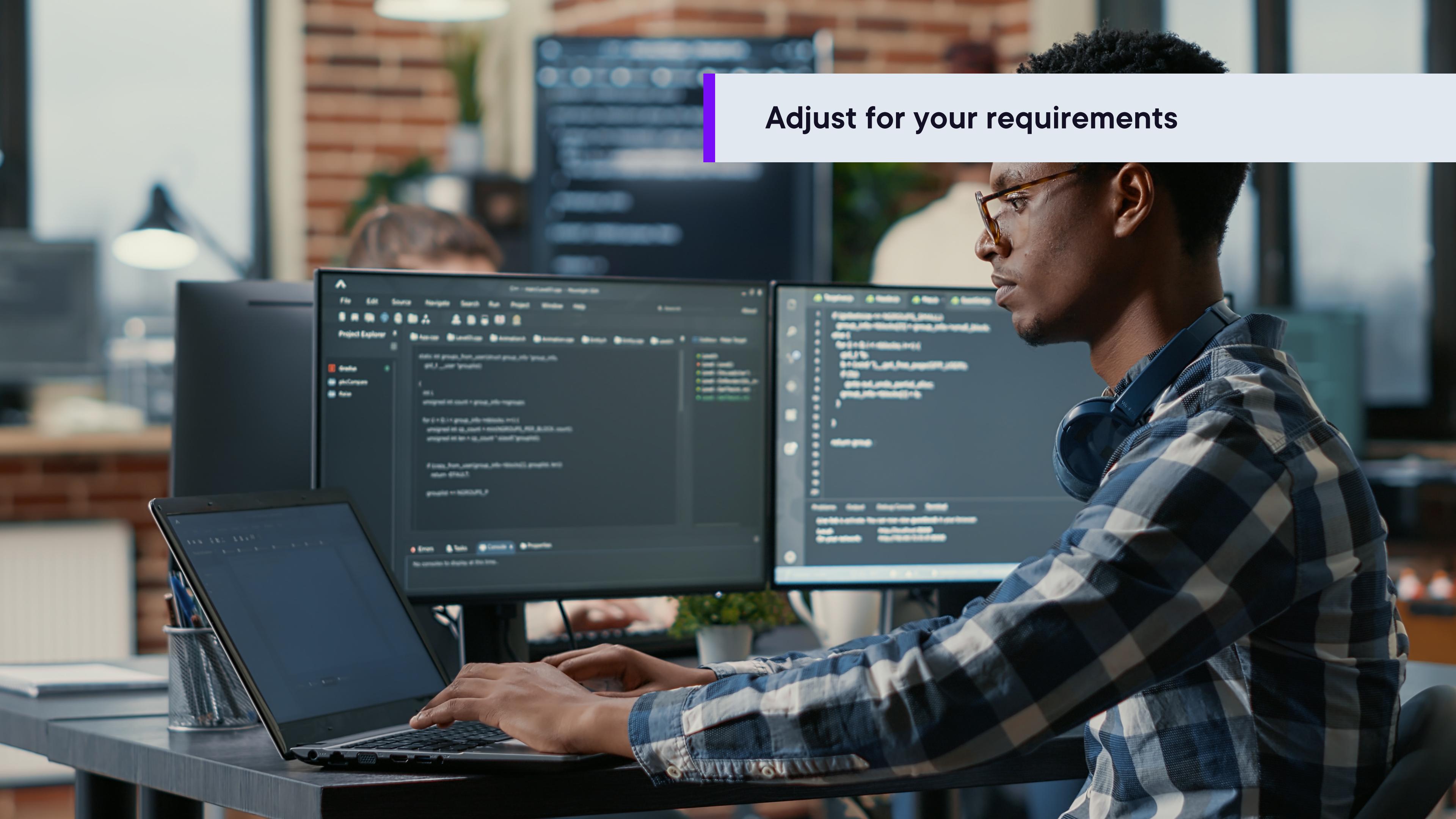


Get all the IP addresses from our log file

Steps:

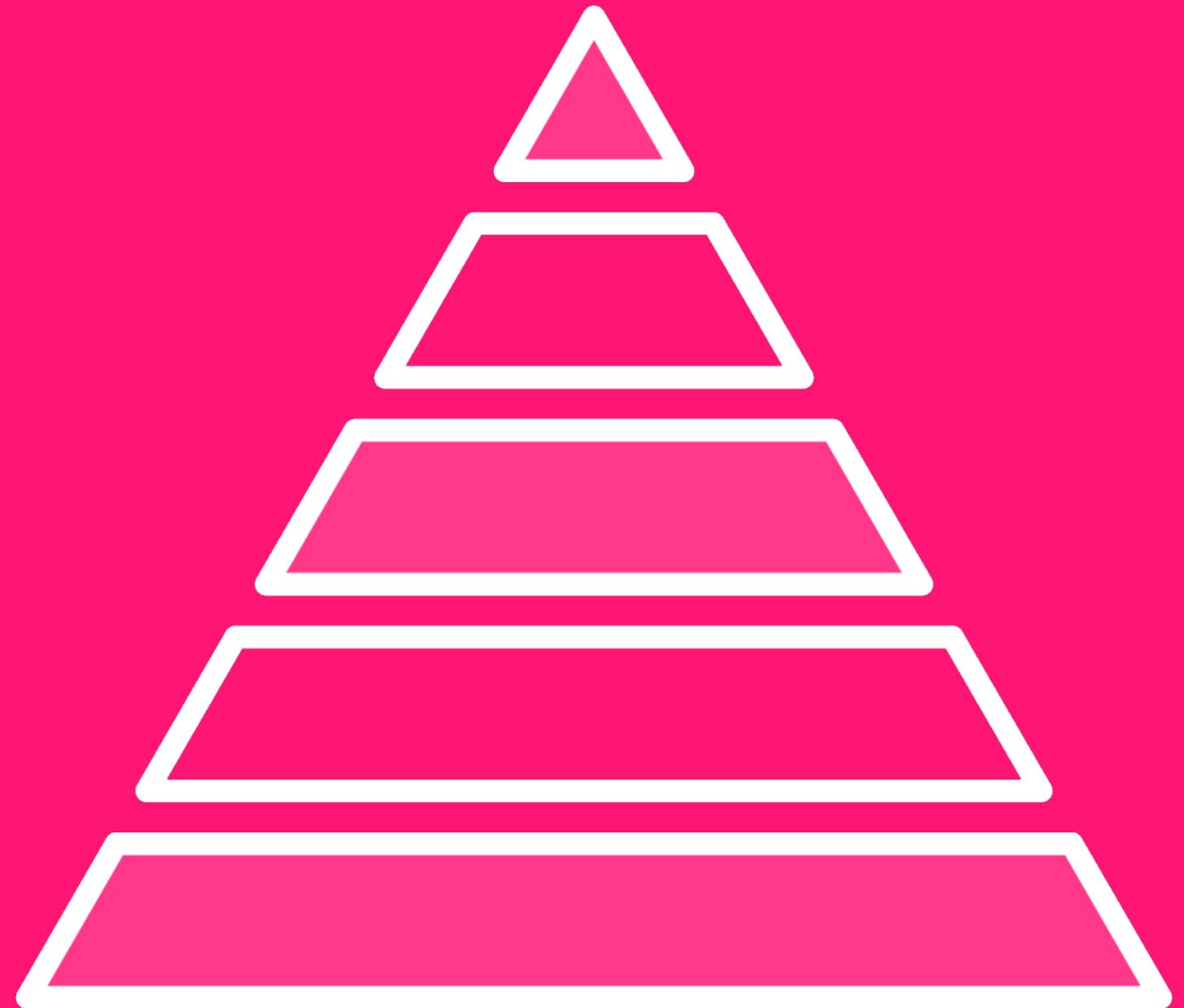
- Define a function that takes a log line as input
- Define a regular expression pattern for IP addresses
- Use the `.findall` function to find all IP addresses
- Open the log file and process each line





Adjust for your requirements





Different Logging Levels

Depending on the type of event, we want to choose a different log level.



Demo

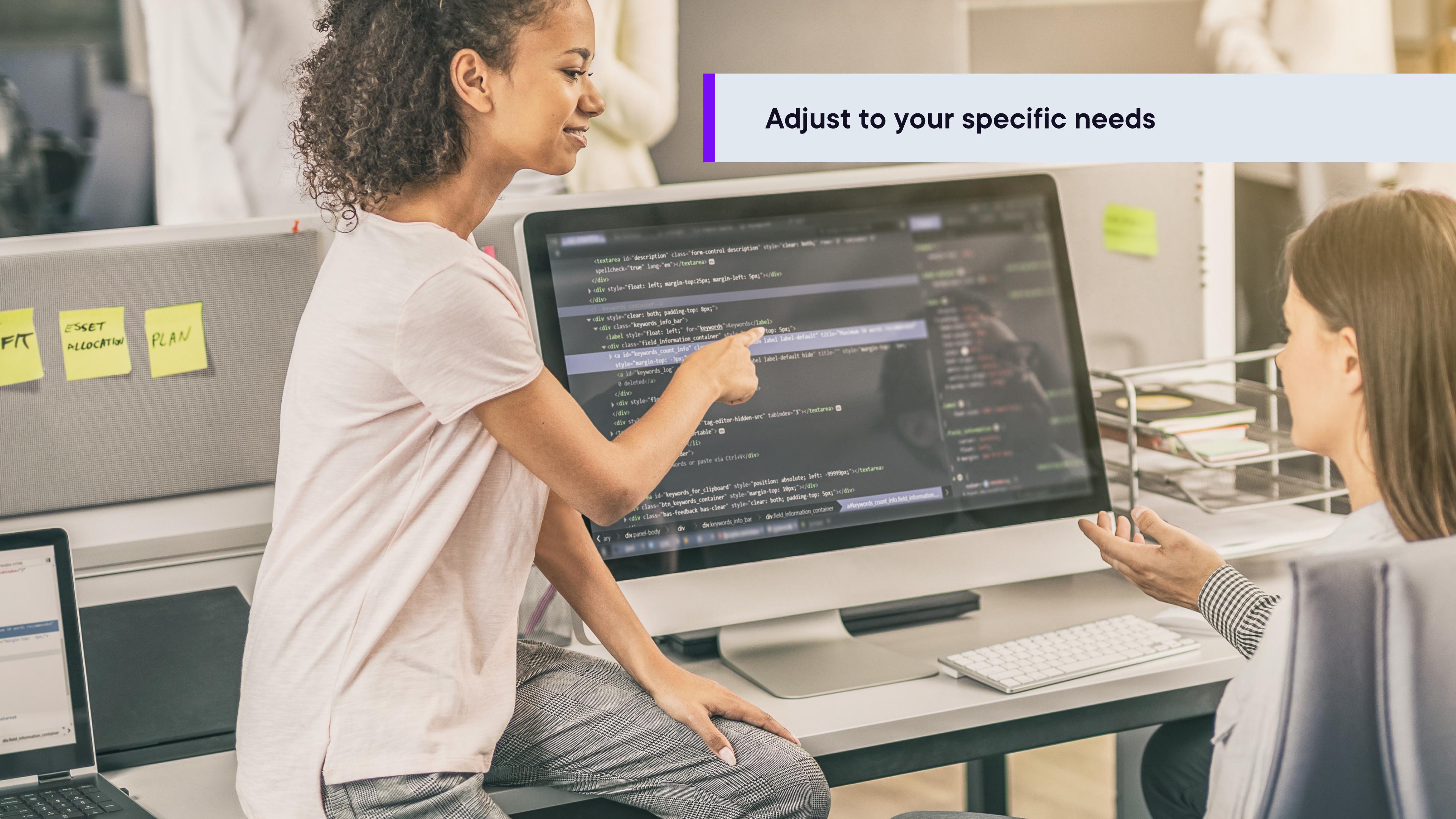


Find different types of log entries

Steps:

- Define a function that takes a log line as input
- Define a pattern that matches a log entry type
- Use the search function to find the log entry type pattern in the log line.
- Extract the log entry type of the match
- Call the function for each line to identify and print the type of log entry





Adjust to your specific needs





Search for specific error codes

Demo



Extract error codes from log files

Steps:

- Define a function that takes a log line and an error code as inputs
- Define a pattern that to find the errors
- Use the search function to find the error code pattern in the log line.
- Call the function for each line and print any matching log entries



A medium shot of a young woman with dark, curly hair and a warm complexion. She is wearing a white, long-sleeved button-down shirt. She is smiling broadly and looking towards the camera. In her right hand, she holds a white ceramic coffee cup with a spoon resting in it. Her left arm is partially visible, showing a gold bracelet and a gold ring. The background is slightly blurred, showing what appears to be a modern interior or a cafe setting.

Adjust to meet your requirements





Process Logs Live

We can detect problems while scanning a live log stream for specific problems and errors.



Demo

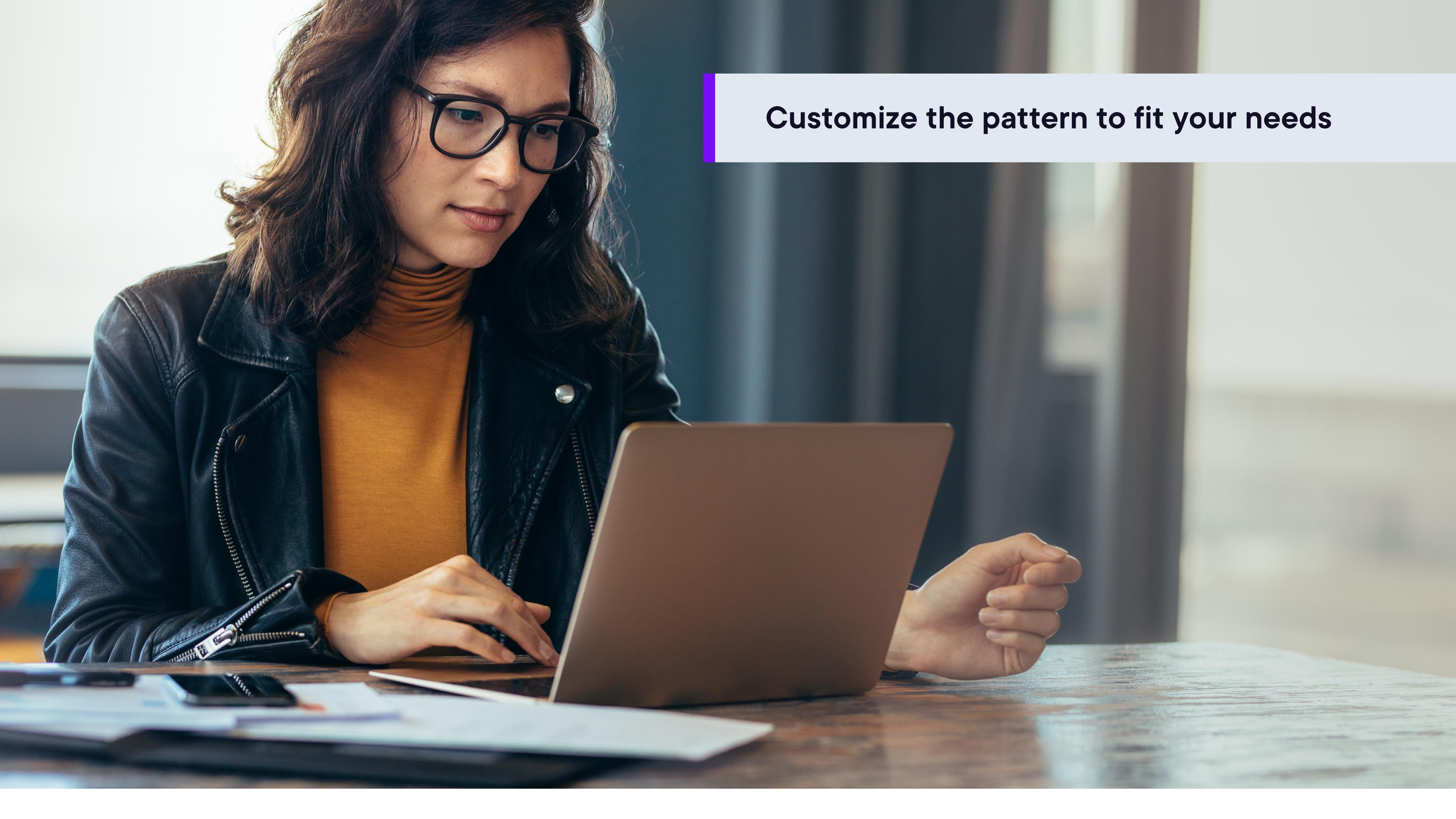


Filter a live log stream for error codes

Steps:

- Define a function that generates a log message with a random log level
- Define a function that takes a log line as input and filters for the error pattern
- Use the search function to match the pattern in the log line.
- Simulate a live logging system
- Call the `filter_errors` function on each generated log entry



A woman with long dark hair and glasses, wearing a black leather jacket over an orange turtleneck, is sitting at a wooden table and working on a silver laptop. She is looking down at the screen. On the table in front of her are some papers and a smartphone. The background is a blurred indoor setting.

Customize the pattern to fit your needs





Processing a Text Log File

Sometimes we need to turn the contents of the log file into objects

This can be done with Python Regex



Demo

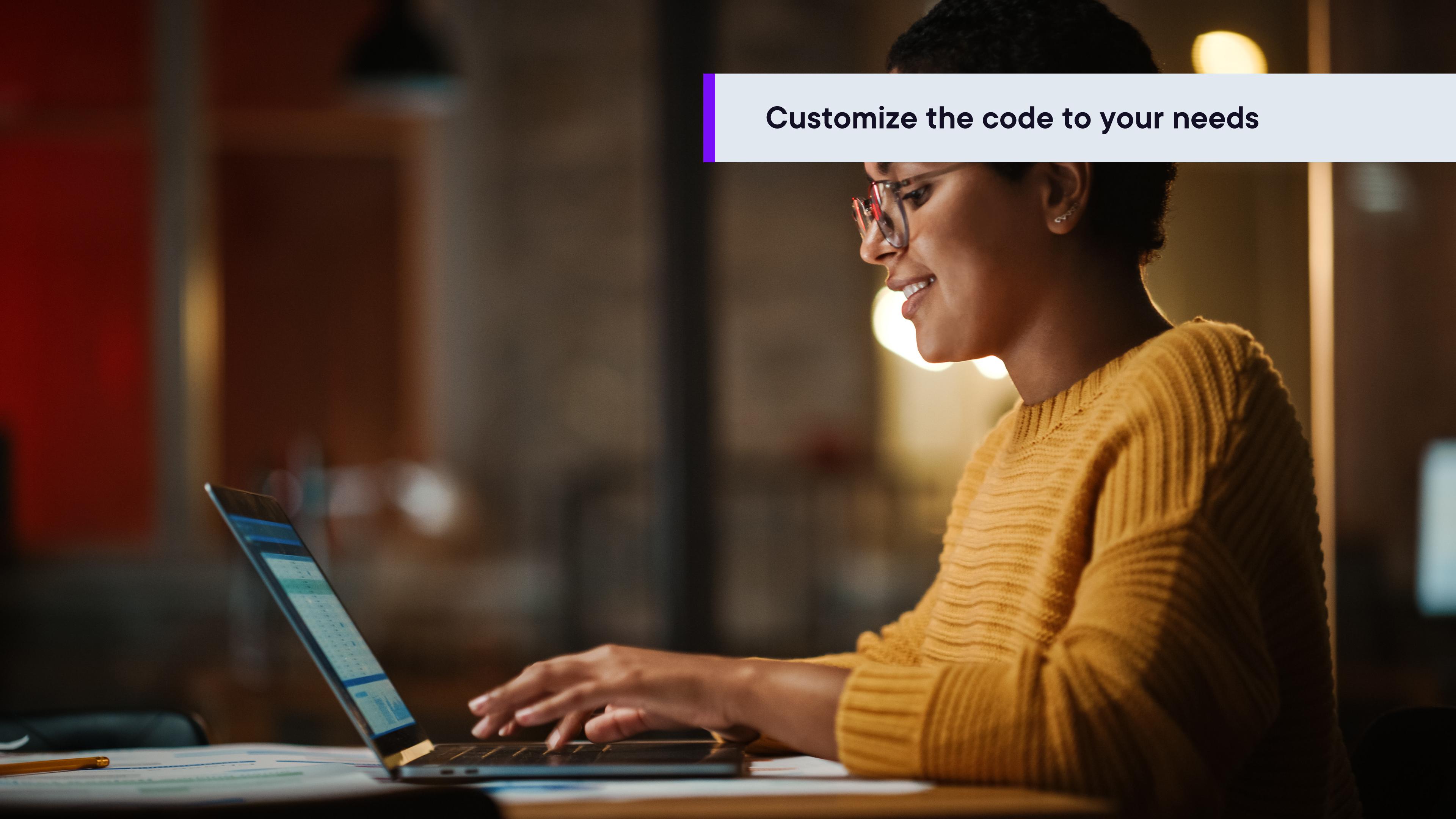


Split entries of a log file using the split function

Steps:

- Create a function `split_log_entry` that takes a log line as input.
- Define a pattern that matches one or more whitespace characters
- Use the `split` function to split the log line into parts based on the pattern
- Extract several components
- Call the `split` function for each line



A woman with dark hair and glasses, wearing a yellow ribbed sweater, is sitting at a desk and working on a laptop. She is looking down at the screen, which displays some code or data. The background is dark, with a warm light source visible behind her. A purple vertical bar is positioned on the left side of the text area.

Customize the code to your needs