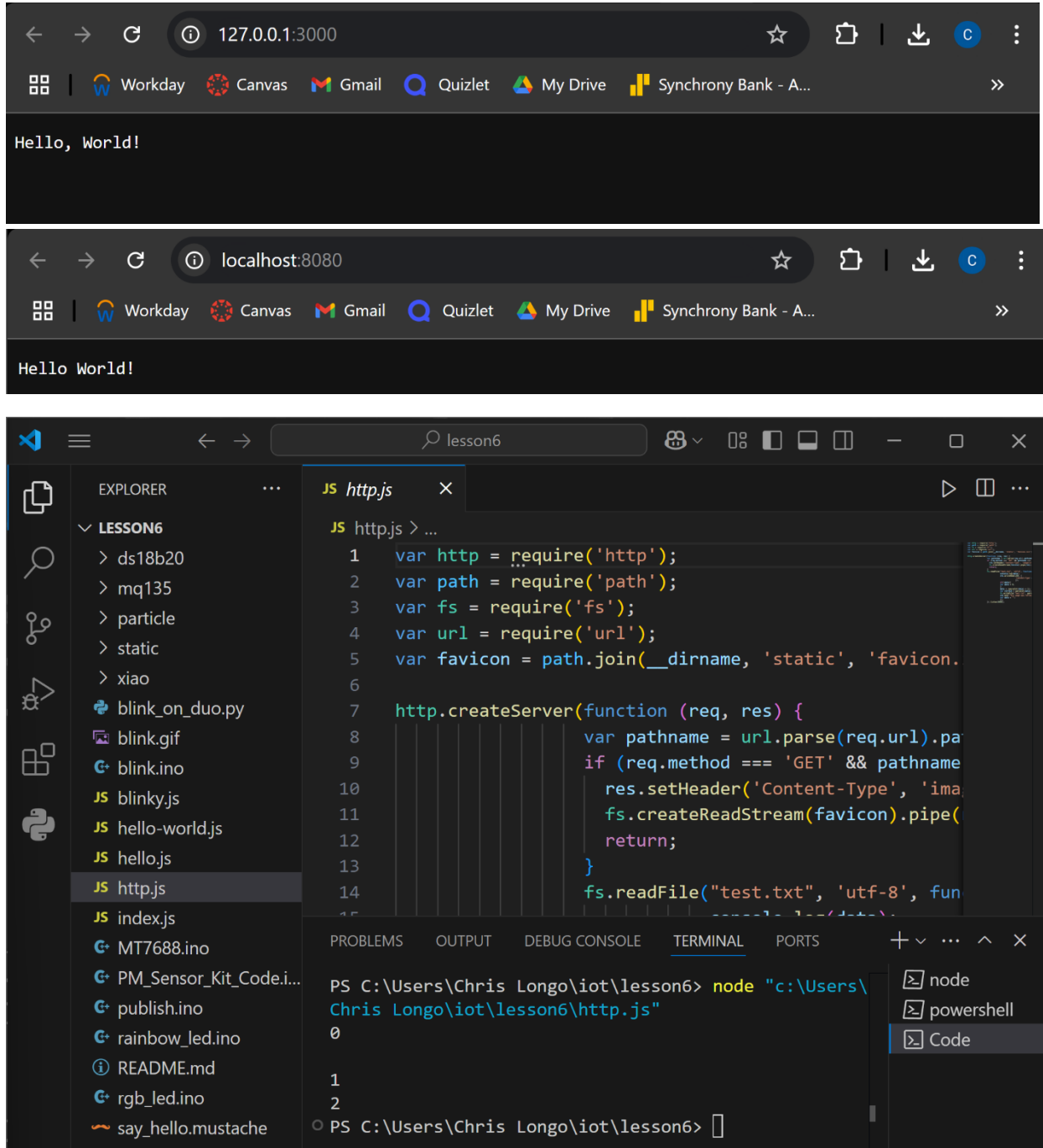
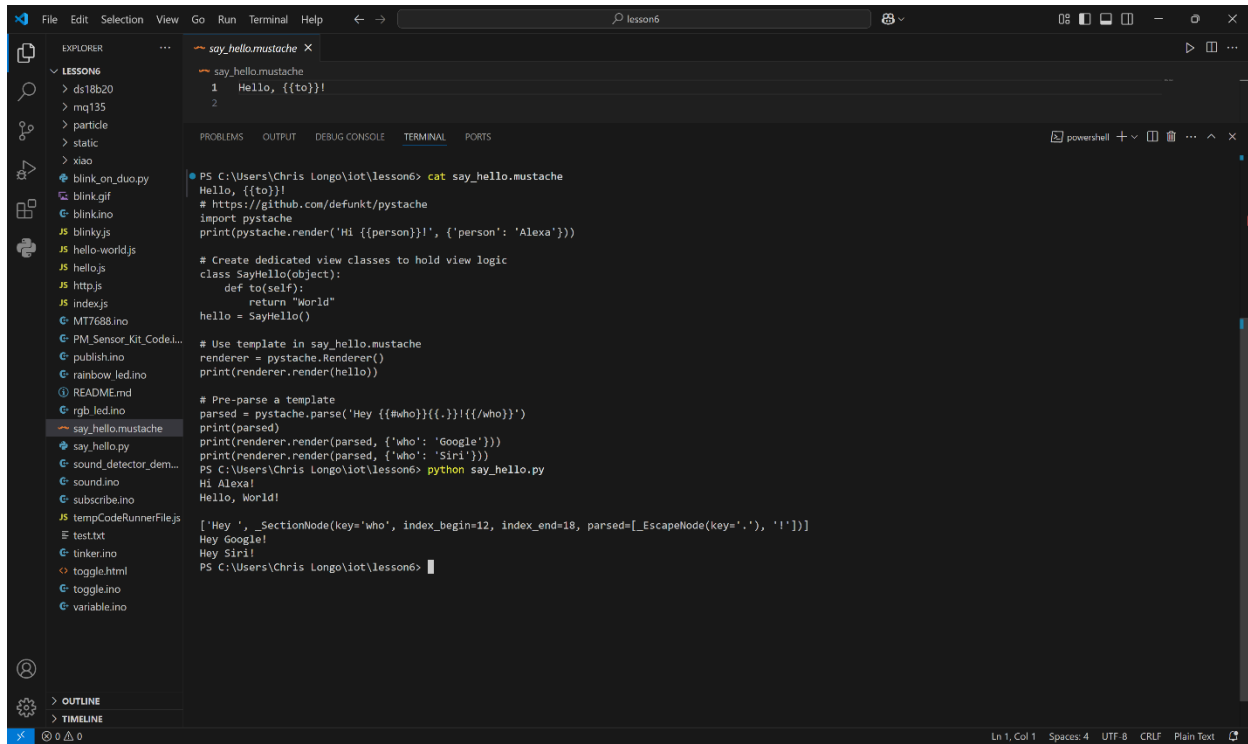


Lab 6 – Node.js and Pystache

Node.js webpage and implementation:



Pystache Implementation:



```
say_hello.mustache
1 Hello, {{to}}!
2

PS C:\Users\Chris Longo\iot\lesson6> cat say_hello.mustache
Hello, {{to}}!
# https://github.com/defunkt/pystache
import pystache
print(pystache.render('Hi {{person}}!', {'person': 'Alexa'}))

# Create dedicated view classes to hold view logic
class SayHello(object):
    def to(self):
        return "World"
    hello = SayHello()

# Use template in say_hello.mustache
renderer = pystache.Renderer()
print(renderer.render(hello))

# Pre-parse a template
parsed = pystache.parse('Hey {{#who}}{{.}}{{/who}}')
print(parsed)
print(renderer.render(parsed, {'who': 'Google'}))
print(renderer.render(parsed, {'who': 'Siri'}))
PS C:\Users\Chris Longo\iot\lesson6> python say_hello.py
Hi Alexa!
Hello, World!

['Hey ', _SectionNode(key='who', index_begin=12, index_end=18, parsed=[_EscapeNode(key='.', 'I')])]
Hey Google!
Hey Siri!
PS C:\Users\Chris Longo\iot\lesson6>
```