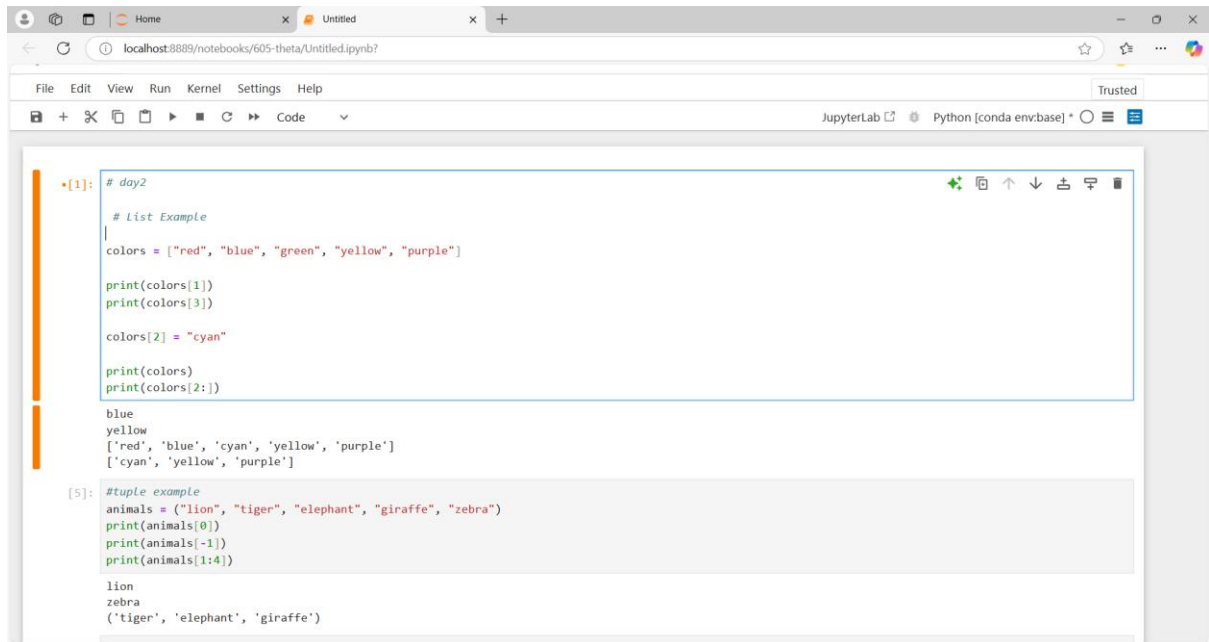


Create a List, tuple and Dictionary with 5 elements in it and how to access few elements based on the index. Try with different examples.



The image shows a JupyterLab interface with a single code cell. The code defines a list named 'colors' with five elements: 'red', 'blue', 'green', 'yellow', and 'purple'. It then prints the element at index 1, the element at index 3, updates the element at index 2 to 'cyan', and prints the entire list and the element at index 2. The output shows the list after the update: ['red', 'blue', 'cyan', 'yellow', 'purple'] and the value 'cyan'.

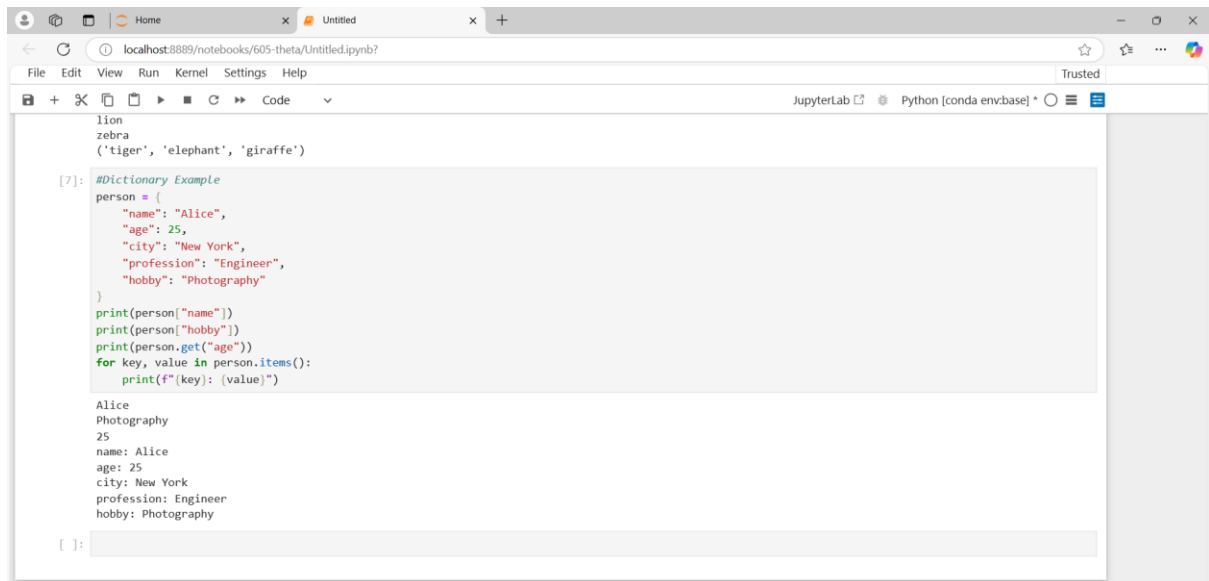
```
[1]: # day2
# List Example
|
colors = ["red", "blue", "green", "yellow", "purple"]

print(colors[1])
print(colors[3])

colors[2] = "cyan"

print(colors)
print(colors[2:])

blue
yellow
['red', 'blue', 'cyan', 'yellow', 'purple']
['cyan', 'yellow', 'purple']
```



The image shows a JupyterLab interface with a single code cell. The code defines a dictionary named 'person' with five key-value pairs: 'name' (Alice), 'age' (25), 'city' (New York), 'profession' (Engineer), and 'hobby' (Photography). It then prints the values for 'name' and 'hobby', prints the value for 'age' using the .get() method, and iterates over the items to print each key-value pair. The output shows the values 'Alice' and 'Photography', the value '25', and the full dictionary representation: {'name': 'Alice', 'age': 25, 'city': 'New York', 'profession': 'Engineer', 'hobby': 'Photography'}.

```
[7]: #Dictionary Example
person = {
    "name": "Alice",
    "age": 25,
    "city": "New York",
    "profession": "Engineer",
    "hobby": "Photography"
}

print(person["name"])
print(person["hobby"])
print(person.get("age"))
for key, value in person.items():
    print(f"{key}: {value}")

Alice
Photography
25
name: Alice
age: 25
city: New York
profession: Engineer
hobby: Photography
```