

Automated API Testing Cheat Sheet (English Version)

1. requests Basics

GET

```
response = requests.get("https://api.example.com/items")
assert response.status_code == 200
```

POST

```
payload = {"username": "admin", "password": "1234"}
response = requests.post("https://api.example.com/login", json=payload)
```

2. Authenticated API

Basic Auth

```
from requests.auth import HTTPBasicAuth

response = requests.get("https://httpbin.org/basic-auth/user/pass",
auth=HTTPBasicAuth("user", "pass"))
```

Bearer Token

```
headers = {"Authorization": "Bearer YOUR_TOKEN"}
requests.get("https://api.example.com/profile", headers=headers)
```

3. Data-Driven Testing (JSON)

login_data.json

```
[
  { "username": "emilys", "password": "emilyspass" },
  { "username": "wronguser", "password": "emilyspass" }
]
```

Using pytest

```
def load_data(path):
    with open(path) as f:
        return [(d["username"], d["password"]) for d in json.load(f)]

@pytest.mark.parametrize("user, pwd", load_data("login_data.json"))
```

```
def test_login(user, pwd): ...
```

4. HTML Report (pytest-html)

Generate report

```
pytest test_file.py --html=report.html --self-contained-html --metadata tester jamei --metadata  
project API
```

Set report title

```
--report-title="Login API Test Report"
```

5. Inspect Response

Print response body

```
print(response.text)
```

Check JSON field

```
data = response.json()  
assert "token" in data
```

6. Login validation logic

```
if response.status_code == 200:  
    assert "accessToken" in response.json()  
else:  
    assert response.status_code in [400, 401]
```

7. Collect test list

```
pytest test_file.py --collect-only
```