

# requests

August 20, 2025

## 1 Biblioteka requests jako klient API

```
[ ]: import requests  
     from urllib.parse import urljoin
```

```
[ ]: main_url = "http://localhost:8000"
```

### 1.1 get

root

```
[ ]: response = requests.get(main_url)  
     response
```

```
[ ]: response.json()
```

All tasks

```
[ ]: response = requests.get(urljoin(main_url, "tasks"))  
     response
```

```
[ ]: response.url
```

```
[ ]: response.json()
```

Single task

```
[ ]: task_id = 1  
  
     response = requests.get(urljoin(main_url, f"tasks/{task_id}"))  
     response
```

```
[ ]: response.url
```

```
[ ]: response.json()
```

## 1.2 post

```
[ ]: task = {"description": "Opis zadania", "priority": 1}
```

```
[ ]: response = requests.post(urljoin(main_url, "tasks"), json=task)
response
```

```
[ ]: response.json()
```

## 1.3 put

```
[ ]: updated_task = {"description": "Opis zadania", "priority": 2, "is_completed":  
    ↪ True}
```

```
[ ]: task_id = 3
response = requests.put(urljoin(main_url, f"tasks/{task_id}"),  
    ↪ json=updated_task)
response
```

```
[ ]: response.json()
```

## 1.4 delete

```
[ ]: task_id = 3

response = requests.delete(urljoin(main_url, f"tasks/{task_id}"))
response
```

```
[ ]: response.text
```

## ZADANIE

Wykonaj analogiczne requesty dla userów.

```
[ ]:
```