

# Wprowadzenie do zaawansowanych technik programowania

# Serwer

**Serwer** – program komputerowy świadczący usługi na rzecz odpowiednich programów zazwyczaj uruchomionych na innych komputerach podłączonych do sieci komputerowej.

# Jak programy wyglądały kiedyś?

- Każdy z klientów musiał zainstalować sobie program u siebie na sprzęcie, np. z płyty CD.
- Programy skryptowe obsługiwane przez konsole

# Jak programy wyglądają teraz?

- Jedno centralne miejsce które obsługuje wielu klientów, niezależnie jakimi urządzeniami odpytują o dane

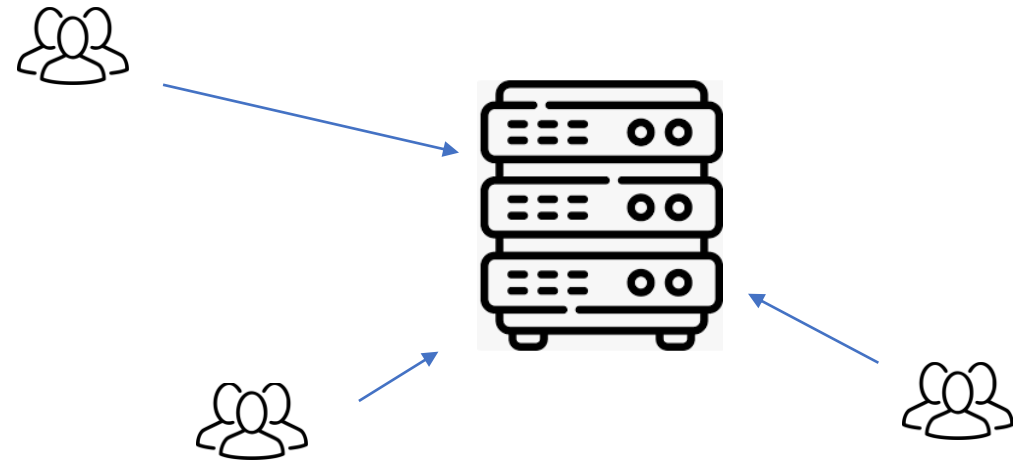
# Przykłady serwerów

- Serwery aplikacji
- Serwery poczty elektronicznej
- Serwery SQL
- Serwery wydruku
- ...

# Jak komunikować się z Serwerem

## HTTP (Hypertext Transfer Protocol)

- jest to **protokół** komunikacji pomiędzy klientem a serwerem
- Określa w jaki sposób treści są udostępniane, przekształcane i w jaki sposób ma je odczytać serwer
- Protokół bezstanowy – nie przechowuje żadnych danych



# Składowe zapytania HTTP

- URL
- Żądanie HTTP
- Odpowiedź HTTP
- Czasowniki HTTP
- Nagłówki HTTP
- Statusy HTTP

# URL – adres naszego zasobu

<http://superserwer.com:8080/testowy/adres/strony?programista=python>

Część adresu	Wartość	Opis
scheme	http	Określenie protokołu komunikacji, najczęściej http lub https
host	Superserwer.com	Wielkość liter nie ma znaczenia. Nazwa domeny internetowej lub adres IP
Port	8080	Numer portu na jakim nasłuchuje nasz serwer. Domyślnie dla protokołu. HTTP – 80 HTTPS – 433
Path	/testowy/adres/strony	Część URL która określa zasób
Query	?programista=python	Zawiera dodatkowe dane identyfikujące dany zasób jako <i>klucz=wartość</i> . Może występować ich wiele. Łącznik: &

# Żądanie HTTP

- Czasownik HTTP
- Nagłówki HTTP
- Ciało wiadomości – Opcjonalnie jeżeli istnieje

# Odpowiedź HTTP

- Wersja Protokołu i Status odpowiedzi
- Nagłówki HTTP
- Ciało wiadomości – Opcjonalnie jeżeli istnieje



# Metody (czasowniki) HTTP

- **GET** - pobranie zasobu wskazanego przez URI
- **HEAD** – pobieranie informacji o zasobie – podobne do GET ale odpowiedź nie może zawierać body
- **PUT** – aktualizacja danego zasobu
- **POST** – tworzenie nowego zasobu
- **DELETE** – usuwanie zasobu
- **OPTIONS** – pobieranie informacji na temat możliwych komunikacji dla danego zasobu

# Statusy odpowiedzi HTTP

*Statusy to informacja czy nasze wysłane żądanie się powiodło*

- 1xx – Informacja że zapytanie zostało przyjęte i jest przetwarzane (bardzo rzadko się stosuje)
- 2xx – Grupa statusów informująca że zapytanie zostało poprawnie przetworzone (200 OK, 201 Created, 202 Accepted, 204 No Content)
- 3xx – Grupa statusów informująca o tym że musi zostać wykonana dodatkowa akcja w celu otrzymania odpowiedzi – np. adres zasobu został zmieniony (301 Moved Permanently)
- 4xx – Statusy informujące o błędach **klienta**. (400 Bad Request, 403 Forbidden, 404 Not Found)
- 5xx – Grupa statusów informujących o błędach serwera (500 Internal Server Error, 503 Service Unavailable)