

# Введение в сетевые протоколы

## Рассматриваемые вопросы

1. Сетевая модель OSI
2. Пакет java.net

# Сетевая модель OSI

Данные	Прикладной доступ к сетевым службам
Данные	Представления представление и кодирование данных
Данные	Сеансовый Управление сеансом связи
Блоки	Транспортный безопасное и надёжное соединение точка-точка
Пакеты	Сетевой Определение пути и IP (логическая адресация)
Кадры	Канальный MAC и LLC (Физическая адресация)
Биты	Физический кабель, сигналы, бинарная передача данных

Прикладной уровень или уровень приложений (application layer) – верхний уровень модели, осуществляет связь пользовательских приложений с сетью.

## Представительский уровень

Представительский уровень или уровень представления данных (presentation layer) преобразует данные в соответствующий формат.

Сеансовый уровень или уровень сессий (session layer) организует сеанс связи между компьютерами.

Транспортный уровень (transport layer) – этот уровень обеспечивает надёжность передачи данных от отправителя к получателю.

Сетевой уровень (network layer) определяет путь, по которому данные будут переданы.



Канальный уровень (data link layer) нужен для взаимодействия сетей на физическом уровне.

## Физический уровень

Физический уровень (physical layer) – самый нижний уровень, непосредственно осуществляющий передачу потока данных.

Сокет - это одна конечная точка двустороннего канала связи между двумя программами, работающими на разных компьютерах в сети. Сокет привязан к номеру порта, так что транспортный уровень может идентифицировать приложение, которому предназначены данные для отправки.

Инициализация объекта класса java.net.Socket:

```
public Socket(String host, int port) throws  
UnknownHostException, IOException
```

Инициализация объекта класса java.net.Socket:

```
Socket socketClient =  
    new Socket ("ИМЯ_СЕРВЕРА", 8030);
```

## Сокет сервера

```
public ServerSocket(int port, int backlog,  
                    InetAddress bindAddr) throws IOException
```

Существует два протокола связи, которые можно использовать для программирования сокетов: **протокол пользовательских дейтаграмм (UDP)** и **протокол управления передачей (TCP)** .

Класс `java.net.DatagramSocket` -  
датаграммные сокет (протокол UDP)



1. Опишите основные сетевые протоколы.
2. Какие классы в Java отвечают за сетевое взаимодействие?