**1. Определение понятия «Интернет Вещей»**

**Интернет Вещей (IoT)** — это система, в которой устройства, подключённые к интернету, взаимодействуют друг с другом и обмениваются данными. Примеры: умные часы, термостаты, системы умного дома.

**2. Состав типового IoT-сервиса**

1. **Устройства** — собирают данные или управляют физическим миром (например, датчики температуры).
2. **Сеть** — передаёт данные (Wi-Fi, 4G, Bluetooth).
3. **Облачная платформа** — обрабатывает и хранит данные.
4. **Приложение** — предоставляет пользователю информацию или позволяет управлять устройствами.

Пример: термостат собирает данные о температуре → отправляет их в облако → приложение на телефоне показывает температуру.

**3. Архитектура типовой платформы «Интернета Вещей»**

1. **Уровень устройств** — датчики и устройства (например, камера наблюдения).
2. **Уровень связи** — каналы передачи данных (например, MQTT или HTTP).
3. **Облачный уровень** — хранение и обработка данных.
4. **Уровень приложений** — пользовательский интерфейс (например, мобильное приложение).

**4. Виды облачных платформ в IoT**

1. **Платформы общего назначения**: AWS IoT, Microsoft Azure IoT.
2. **Специализированные платформы**: GE Predix для промышленного IoT.
3. **Open-source платформы**: ThingsBoard, Eclipse IoT.

**5. Виды аналитических платформ в IoT**

1. **Реального времени**: обрабатывают данные сразу (например, Apache Kafka).
2. **Исторические**: анализируют накопленные данные (например, Hadoop, Power BI).
3. **Прогнозирующие**: используют AI/ML для прогнозов (например, TensorFlow).

**6. Язык разметки данных XML**

**XML (eXtensible Markup Language)** — это текстовый формат для структурирования данных.

**Синтаксис**:

<person>

<name>John</name>

<age>30</age>

</person>

**Основные области применения**:

* Обмен данными между приложениями.
* Конфигурационные файлы.

**7. Язык разметки данных JSON**

**JSON (JavaScript Object Notation)** — это компактный текстовый формат для обмена данными.

**Синтаксис**:

{

"person": {

"name": "John",

"age": 30

}

}

**Основные области применения**:

* REST API.
* Хранение данных в NoSQL базах.

**8. Язык разметки данных YAML**

**YAML (YAML Ain't Markup Language)** — человекочитаемый формат для конфигураций.

**Синтаксис**:

person:

name: John

age: 30

**Основные области применения**:

* Конфигурационные файлы (например, в Docker, Kubernetes).

**9. Язык разметки данных HTML**

**HTML (HyperText Markup Language)** — стандартный язык для создания веб-страниц.

**Синтаксис**:

<html>

<body>

<h1>Hello, World!</h1>

</body>

</html>

**Основные области применения**:

* Создание интерфейсов для веб-приложений.

**10. Протокол MQTT**

**MQTT (Message Queuing Telemetry Transport)** — лёгкий протокол передачи данных, оптимизированный для устройств с ограниченными ресурсами.

**Принцип работы**:

1. Устройства (клиенты) подключаются к брокеру MQTT.
2. Клиенты отправляют данные (publish) или подписываются (subscribe) на определённые темы.
3. Брокер передаёт сообщения всем подписанным клиентам.

**Пример**:

* Датчик температуры публикует данные в тему house/temperature.
* Приложение получает эти данные через подписку на тему.

**11. Команды протокола MQTT. Подписка и топики. Брокераж сообщений**

* **Команды MQTT**:
  1. **CONNECT**: подключение клиента к брокеру.
  2. **PUBLISH**: публикация сообщения в топик.
  3. **SUBSCRIBE**: подписка клиента на топик.
  4. **UNSUBSCRIBE**: отписка клиента от топика.
  5. **DISCONNECT**: завершение подключения.
* **Подписка и топики**:  
  **Топик** — это канал передачи сообщений, структурированный иерархически (например, home/kitchen/temp).  
  Клиенты могут:
  1. Подписаться на конкретный топик (home/kitchen/temp).
  2. Использовать шаблоны для подписки (например, home/# — все подуровни в home).
* **Брокераж сообщений**:  
  Брокер (сервер) принимает сообщения от издателей и пересылает их подписчикам.

**12. Основные угрозы безопасности IoT-сервисов и методы борьбы с ними**

1. **Угрозы**:
   * Кража данных.
   * Взлом устройств.
   * DDoS-атаки.
2. **Методы борьбы**:
   * Шифрование данных (например, TLS).
   * Аутентификация устройств.
   * Регулярное обновление прошивки.

**13. Национальная и международная стандартизация в области IoT**

* **Международные стандарты**:
  + ISO/IEC 30141: архитектура IoT.
  + IEEE P2413: стандарты интероперабельности.
* **Национальные стандарты (Россия)**:
  + ГОСТ Р 58875-2020: IoT для умных городов.

**14. Определение понятия «Устройство измерения»**

**Устройство измерения** — техническое средство, используемое для определения значений физических величин (например, термометр, мультиметр).

**15. Формулировка теоремы Котельникова**

**Теорема Котельникова (Шеннона)**:  
Для точного восстановления сигнала он должен быть оцифрован с частотой, превышающей вдвое его максимальную частоту.

Пример: если максимальная частота сигнала 10 кГц, частота дискретизации должна быть > 20 кГц.

**16. Аналогово-цифровой преобразователь (АЦП)**

**Определение**: устройство, преобразующее аналоговый сигнал в цифровой.

**Функции и задачи**:

* Преобразование непрерывных сигналов в дискретные.
* Оцифровка звука, температуры, напряжения и других величин.

Пример: микрофон записывает звук → АЦП превращает его в цифровые данные.

**17. Цифро-аналоговый преобразователь (ЦАП)**

**Определение**: устройство, преобразующее цифровой сигнал в аналоговый.

**Функции и задачи**:

* Воспроизведение аналогового сигнала из цифрового.
* Управление аналоговыми устройствами (например, громкость динамиков).

Пример: аудиофайл преобразуется в звук через ЦАП.

**18. Понятие «сигнал» и его отличие от данных**

**Сигнал** — физический процесс, передающий информацию (например, электрический импульс).

**Отличие**:

* **Сигнал** — это средство передачи.
* **Данные** — это информация, передаваемая сигналом.

**19. Понятие «данные» и их отличие от сигнала**

**Данные** — закодированные значения, представляющие информацию.

**Отличие**:

* **Данные** — содержат смысл.
* **Сигнал** — лишь форма передачи.

**20. Понятие «информация» и её отличие от данных**

**Информация** — осмысленные данные, полезные для пользователя.

**Отличие**:

* **Данные** — сырая информация.
* **Информация** — данные, интерпретированные и понятые.

**21. Понятие «знание» и его отличие от информации**

**Знание** — информация, усвоенная и используемая для принятия решений.

**Отличие**:

* **Информация** — может быть временной и общей.
* **Знание** — структурированное и применимое в контексте.

**22. Цели и задачи создания платформ «Интернета вещей»**

**Цели**:

* Обеспечение взаимодействия устройств.
* Упрощение разработки IoT-приложений.
* Централизованное управление данными.

**Задачи**:

* Сбор, обработка и хранение данных от устройств.
* Обеспечение безопасности.
* Интеграция с другими системами.

**23. Принципы проектирования программных интерфейсов «Интернета вещей»**

1. **Интероперабельность** — поддержка взаимодействия между разными устройствами.
2. **Лёгкость** — минимальная нагрузка на устройства.
3. **Безопасность** — защита данных.
4. **Масштабируемость** — работа с большим количеством устройств.
5. **Реализация стандартов** — использование протоколов, таких как REST, MQTT.

**24. Понятие структуры данных. Способы описания информационных объектов**

**Структура данных** — способ организации и хранения данных для их эффективного использования.

**Способы описания**:

* **Иерархические**: деревья.
* **Линейные**: массивы, списки.
* **Графовые**: графы.
* **Табличные**: базы данных.

**25. Основные компоненты DOM-модели**

1. **Document** — весь документ.
2. **Element** — элементы HTML (например, <div>).
3. **Attributes** — атрибуты элементов.
4. **Text** — текст внутри элементов.
5. **Nodes** — все части DOM, включая комментарии.

**26. Понятие «устройство» в информационной модели IoT**

**Устройство** — физический объект, подключённый к платформе IoT, который собирает данные или выполняет команды (например, датчик температуры, умная лампа).

**27. Модель объекта контроля в информационной модели IoT**

**Объект контроля** — сущность, за которой ведётся наблюдение или управление с помощью IoT-устройства (например, помещение с умными датчиками).

**28. Понятие «актив» в информационной модели IoT**

**Актив** — управляемый ресурс, который используется или мониторится в системе (например, станок на заводе, автомобиль в логистике).

**29. Информационная модель платформы IoT**

**Информационная модель** — описание структуры данных и взаимосвязей между объектами платформы (устройства, события, сообщения, активы).

**30. Понятие «событие» в информационной модели IoT**

**Событие** — изменение состояния устройства или системы, фиксируемое платформой (например, превышение температуры, включение света).

**31. Понятие «сообщение» в информационной модели IoT**

**Сообщение** — единица данных, передаваемая между устройствами или платформой (например, текущая температура: {"temp": 25}).

**32. Понятие «функциональное действие» в информационной модели IoT**

**Функциональное действие** — команда или операция, выполняемая устройством (например, включение/выключение лампы).

**33. Понятие «цепочка действий» в информационной модели IoT**

**Цепочка действий** — последовательность функциональных действий, выполняемых для достижения цели (например, включение кондиционера → настройка температуры → мониторинг).

**34. Понятие «виджет» в информационной модели IoT**

**Виджет** — графический элемент интерфейса, отображающий данные или позволяющий управлять устройствами (например, кнопка включения, график температуры).