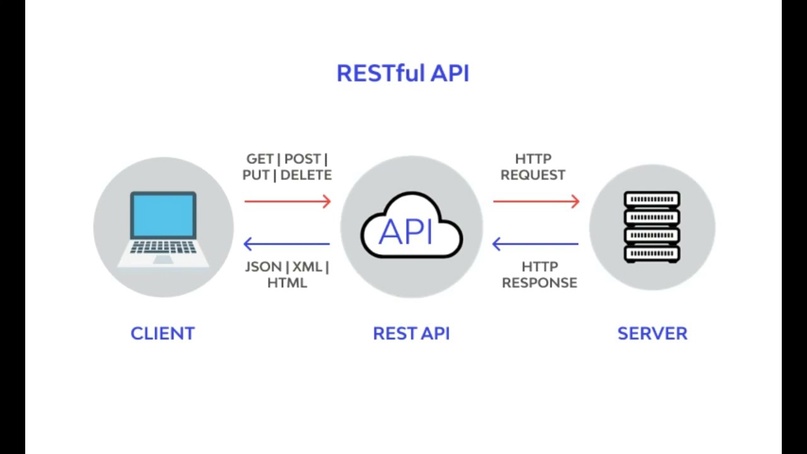
**Тестовый контроль «Основы Rest API»**



**1-15. Вопросы с выбором**

1. **Что такое REST API?**
   1. A) Протокол для передачи файлов
   2. B) Стандарт для создания API, использующий HTTP запросы
   3. C) Фреймворк для работы с базами данных
   4. D) Библиотека для работы с клиентом

**Ответ:** b. B) Стандарт для создания API, использующий HTTP запросы

1. **Какой HTTP-метод используется для получения данных с сервера?**
   1. A) POST
   2. B) GET
   3. C) PUT
   4. D) DELETE

**Ответ:** b. B) GET

1. **Какой HTTP-метод используется для создания нового ресурса на сервере?**
   1. A) POST
   2. B) GET
   3. C) PATCH
   4. D) DELETE

**Ответ:** a. A) POST

1. **Что из ниже перечисленного является характеристикой REST API?**
   1. A) Использование HTTP-методов
   2. B) Передача данных только в формате XML
   3. C) Жесткая привязка к клиентским библиотекам
   4. D) Невозможность работы с внешними сервисами

**Ответ:** a. A) Использование HTTP-методов

1. **Что из следующего является правильным статусом HTTP, который обозначает успешное выполнение запроса?**
   1. A) 404
   2. B) 200
   3. C) 500
   4. D) 301

**Ответ:** b. B) 200

1. **Какой HTTP-метод используется для частичного обновления ресурса?**
   1. A) PATCH
   2. B) GET
   3. C) PUT
   4. D) DELETE

**Ответ:** a. A) PATCH

1. **Что такое RESTful API?**
   1. A) API, использующее асинхронные запросы
   2. B) API, использующее синхронные запросы
   3. C) API, соблюдающее принципы REST, включая использование стандартных HTTP-методов
   4. D) API, использующее только POST-запросы

**Ответ:** c. C) API, соблюдающее принципы REST, включая использование стандартных HTTP-методов

1. **Что такое статус HTTP 404?**
   1. A) Ошибка сервера
   2. B) Ресурс не найден
   3. C) Запрос успешен
   4. D) Проблема с базой данных

**Ответ:** b. B) Ресурс не найден

1. **Каким образом можно передавать параметры в запросах к REST API?**
   1. A) Только через тело запроса
   2. B) Только через URL
   3. C) Через URL или тело запроса
   4. D) Невозможно передавать параметры

**Ответ:** c. C) Через URL или тело запроса

1. **Какой формат данных обычно используется для обмена данными с REST API?**

* A) CSV
* B) JSON
* C) HTML
* D) XML

**Ответ:** B) JSON

1. **Что такое endpoint в контексте REST API?**

* A) Специальный тип запроса
* B) URL, по которому выполняется запрос к API
* C) Протокол для передачи данных
* D) Статусный код ответа

**Ответ:** B) URL, по которому выполняется запрос к API

1. **Что такое метод PUT в контексте REST API?**

* A) Создает новый ресурс
* B) Удаляет ресурс
* C) Обновляет существующий ресурс
* D) Получает данные от сервера

**Ответ:** C) Обновляет существующий ресурс

1. **Какой HTTP-метод используется для удаления ресурса на сервере?**

* A) POST
* B) PUT
* C) DELETE
* D) GET

**Ответ:** C) DELETE

1. **Какой HTTP-метод используется для получения информации о ресурсе, но без его изменений?**

* A) PATCH
* B) PUT
* C) DELETE
* D) GET

**Ответ:** D) GET

1. **Какой статусный код HTTP обозначает успешное создание ресурса?**

* A) 200
* B) 201
* C) 404
* D) 500

**Ответ:** B) 201

**16-25. Открытые вопросы**

1. **Что такое REST API и какие основные принципы лежат в его основе?**

**Открытый ответ:**

**Что такое REST API и какие основные принципы лежат в его основе?**

**REST API** (Representational State Transfer Application Programming Interface) — это набор правил и принципов для создания веб-сервисов, которые позволяют взаимодействовать с ресурсами через HTTP-протокол. Основные принципы REST:

* **Клиент-серверная архитектура**: Клиент и сервер работают независимо друг от друга.
* **Безсостояние (Stateless)**: Каждый запрос от клиента к серверу должен содержать всю необходимую информацию для его обработки.
* **Кэширование**: Ответы сервера могут быть кэшированы клиентом для улучшения производительности.
* **Единообразие интерфейса**: Использование стандартных HTTP-методов (GET, POST, PUT, DELETE и т.д.).
* **Слои (Layered System)**: Архитектура может включать промежуточные серверы, такие как прокси или шлюзы.
* **Код по требованию (Code on Demand, опционально)**: Сервер может отправлять исполняемый код клиенту для выполнения.

1. **Опишите, как работает метод GET в контексте REST API и как его использовать для получения данных.**

**Открытый ответ:**

**Опишите, как работает метод GET в контексте REST API и как его использовать для получения данных.**

Метод **GET** используется для получения данных с сервера. Он не изменяет состояние ресурса на сервере. Параметры запроса передаются в URL. Пример:

GET /users?id=123

Этот запрос вернет данные пользователя с ID 123.

1. **Как в REST API происходит обработка запросов с использованием метода POST? Приведите пример его применения.**

**Открытый ответ:**

**Как в REST API происходит обработка запросов с использованием метода POST? Приведите пример его применения.**

Метод **POST** используется для отправки данных на сервер, обычно для создания нового ресурса. Данные передаются в теле запроса. Пример:

POST /users

Content-Type: application/json

{

"name": "John Doe",

"email": "john.doe@example.com"

}

Этот запрос создаст нового пользователя с указанными данными.

1. **Объясните, чем отличаются методы PUT и PATCH в REST API. В каких случаях следует использовать каждый из них?**

**Открытый ответ:**

**Объясните, чем отличаются методы PUT и PATCH в REST API. В каких случаях следует использовать каждый из них?**

* **PUT**: Используется для полного обновления ресурса. Клиент отправляет полное представление ресурса.
* **PATCH**: Используется для частичного обновления ресурса. Клиент отправляет только измененные данные.

Примеры:

* PUT:

PUT /users/123

Content-Type: application/json

{

"name": "John Doe",

"email": "john.doe@example.com",

"age": 30

}

* PATCH:

PATCH /users/123

Content-Type: application/json

{

"email": "new.email@example.com"

}

1. **Что такое статусный код 500, и в каких случаях он может быть возвращен сервером в ответ на запрос?**

**Открытый ответ:**

**Что такое статусный код 500, и в каких случаях он может быть возвращен сервером в ответ на запрос?**

Статусный код **500 Internal Server Error** указывает на то, что сервер столкнулся с непредвиденной ошибкой, которая помешала выполнению запроса. Это может быть вызвано различными причинами, такими как ошибки в коде сервера, проблемы с базой данных и т.д.

1. **Как происходит передача параметров в REST API, если они находятся в теле запроса? Приведите пример.**

**Открытый ответ:**

**Как происходит передача параметров в REST API, если они находятся в теле запроса? Приведите пример.**

Параметры в теле запроса обычно передаются в формате JSON или XML. Пример:

POST /users

Content-Type: application/json

{

"name": "John Doe",

"email": "john.doe@example.com"

}

1. **Какие преимущества и ограничения имеет использование JSON для обмена данными в REST API?**

**Открытый ответ:**

**Какие преимущества и ограничения имеет использование JSON для обмена данными в REST API?**

**Преимущества JSON**:

* Легко читаем и пишется.
* Поддерживается большинством языков программирования.
* Меньше объем данных по сравнению с XML.

**Ограничения JSON**:

* Нет встроенной поддержки схем данных.
* Менее строгий формат, что может привести к ошибкам.

1. **Объясните, что такое аутентификация в REST API и как она может быть реализована с помощью токенов.**

**Открытый ответ:**

**Объясните, что такое аутентификация в REST API и как она может быть реализована с помощью токенов.**

**Аутентификация** — это процесс проверки подлинности пользователя. В REST API часто используется токен-аутентификация, где клиент получает токен после успешной аутентификации и использует его для доступа к защищенным ресурсам. Пример:

POST /login

Content-Type: application/json

{

"username": "user",

"password": "pass"

}

Сервер возвращает токен, который клиент использует в заголовке Authorization для последующих запросов:

GET /users

Authorization: Bearer <token>

1. **Каковы основные отличия между REST API и SOAP API?**

**Открытый ответ:**

**Каковы основные отличия между REST API и SOAP API?**

* **REST**: Легковесный, использует стандартные HTTP-методы, данные обычно передаются в формате JSON или XML.
* **SOAP**: Более строгий протокол, использует XML для передачи данных, поддерживает WSDL для описания сервисов.

1. **Какие средства для тестирования REST API вы бы порекомендовали и почему?**

**Открытый ответ:**

**Какие средства для тестирования REST API вы бы порекомендовали и почему?**

* **Postman**: Удобный интерфейс, поддержка автоматических тестов, коллекции запросов.
* **Insomnia**: Легковесный, поддержка работы с окружениями, автоматические тесты.
* **Curl**: Командная строка, гибкость, подходит для автоматизации.
* **JMeter**: Для нагрузочного тестирования, поддержка сложных сценариев.

Эти инструменты помогают упростить процесс тестирования и отладки REST API.

**Вопросы на сопоставление**

1. **Сопоставьте HTTP-методы с их назначением.**

**Методы:**

1. GET
2. POST
3. PUT
4. PATCH
5. DELETE

**Назначения:**

A) Используется для получения информации с сервера без изменения данных.  
B) Используется для частичного обновления существующего ресурса.  
C) Используется для удаления ресурса с сервера.  
D) Используется для создания нового ресурса на сервере.

E) Используется для полного обновления существующего ресурса.

**Ответ:**

1. **GET** - A) Используется для получения информации с сервера без изменения данных.
2. **POST** - D) Используется для создания нового ресурса на сервере.
3. **PUT** - E) Используется для полного обновления существующего ресурса.
4. **PATCH** - B) Используется для частичного обновления существующего ресурса.
5. **DELETE** - C) Используется для удаления ресурса с сервера.