**JEE**

1. Что входит в JEE?

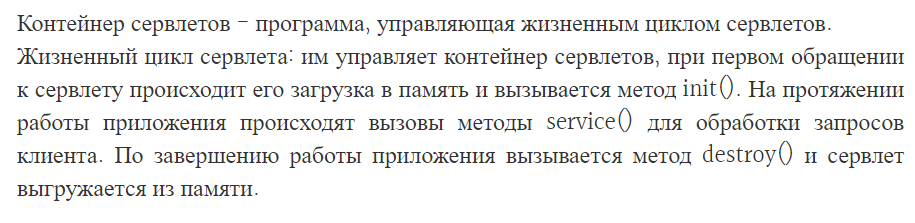
Java EE или **Java Enterprise Edition** представляет платформу для создания корпоративных приложений на языке Java. Прежде всего это сфера веб-приложений и веб-сервисов.

Java EE состоит из набора API и среды выполнения. Некоторые из API:

* **Java Servlets**. Сервлеты представляют специальные модули, которые обрабатывают запросы от пользователей и отправляют результат обработки.
* **JavaServer Pages (JSP)**. Также модули на стороне сервера, которые обрабатывают запросы. Удобны для генерации большого контента HTML. По сути предствляют собой страницы с кодом HTML/JavaScript/CSS с вкраплениями кода на Java
* **Enterprise JavaBeans (EJB)** представляют классы, которые хранят бизнес-логику.
* **Contexts and Dependency Injection (CDI)** предоставляет механизм для внедрения и управления зависимостями в другие объекты.
* **JSON Processing (JSON-P)** позволяет работать со строками JSON в Java
* **JSON Binding (JSON-B)** предоставляет функционал для сериализации и десериализации JSON в объекты Java.
* **WebSocket** позволяет интегрировать WebSocket в приложения на Java.
* **Java Message Service (JMS)** - API для пересылки сообщений между двумя и более клиентами.
* **Security API** - API для стандартизации и упрощения задач обеспечения безопасности в приложениях на Java.
* **Java API for RESTful Web Services (JAX-RS)** - API для применения архитектуры REST в приложениях.
* **JavaServer Faces (JSF)** предоставляет возможности для создания пользовательского интерфейса на стороне сервера.

Эти и ряд других API сообственно и образуют то, что называется Java EE. Стоит отметить, что также в среде веб-разработки на Java популярна еще одна технология Spring. Фреймворк Spring не является частью Java EE и может использоваться как альтернативный подход к созданию веб-приложений на языке Java.

1. Что такое сервер приложений? Что такое веб-сервер, в чём его отличие от сервера приложений? Привести примеры веб-сервера и сервера приложений.
2. Что такое контейнер сервлетов? Что такое сервлет?



1. Методы сервлета. Жизненный цикл сервлета.
2. Что такое jsp. Жизненный цикл.
3. Что такое сессия? Жизненный цикл.
4. Что такое request? Из чего состоит? Жизненный цикл.
5. Сookies. Как можно достать Сookies, а если Сookies удалить, можно ли достать сессию?
6. Что нужно написать в браузерной строке, чтобы обратиться к сервлету? Можно ли из браузерной строки напрямую вызвать метод сервлета?
7. Чем отличается методы POST и GET. Если не указать напрямую, какой из этих методов выполнится по умолчанию?
8. Как сделать redirect незаметно для пользователя?
9. Какие scopes (области видимости) переменных существуют в JSP?
10. В чём различие forward и redirect?
11. Отличия getAttribute() от getParameter() в сервлете.
12. Из чего состоит url?
13. Отличие jsp:include от директивы include.
14. Применение классов HttpServletRequestWrapper и HttpServletResponseWrapper.
15. Что делает RequestDispatcher.include()?
16. В какой последовательности выполняются сервлет фильтры?
17. Как обрабатывается тег с телом?
18. JSTL.
19. Что нужно написать в строке браузера, чтобы обратиться к хосту, на котором установлен tomcat, развёрнуто приложение, в котором есть несколько сервлетов? Как обратиться к конкретному сервлету? Что такое www? Где нужно указывать порт?
20. Можно ли в web.xml определить сервлет без указания url паттерна и как к нему обратиться?
21. Что такое HTTP? Отличия HTTP 1.0 и HTTP 2.
22. Cохраняет ли http протокол своё состояние.
23. Как сервер понимает, что для пользователя создана сессия и не нужно её создавать?
24. Отличие авторизации от аутентификации.
25. Сервер приложений. Веб-сервер. Отличия.
26. Пошагово рассказать, что происходит, когда пользователь нажимает на кнопку. (с формы поля сетаются в request, потом вызывается контейнер сервлетов, потом он как-то по request понимает, куда нужно идти дальше (в дескриптор развертывания, а их может быть несколько, нужно как-то понимать в какой).
27. Какие бывают WebService?
28. RESTful сервис. Принципы работы.