

Тест 1 -----

Тема: 3. Система идентификации соединений в ТСП. Статические и динамические порты, сокеты и серверные сокеты. Процедуры установления и разрыва соединения. Состояния соединения.

2. Определение динамического порта. Когда он создается? 2

Динамический порт – порт с номером, выделяемый узлом сети для данного конкретного соединения при установлении его приложением-клиентом. Занятость номеров динамических портов контролируется службой установления соединений.

Статический порт сервера (приемника соединения) в связке IP-адрес:порт может использоваться для адресации службы/приложения. В то же время инициатору соединения (клиенту) нет необходимости идентифицировать себя, поэтому он использует динамический порт, выделяемый узлом сети для данного конкретного соединения. Диапазон динамических портов и их занятость контролируется службой установления соединений.

Если служба или приложение может поддерживать множественные соединения, то связка IP-адрес:статический порт уже не является идентификатором соединения. Поэтому установленное соединение на обоих концах (в обоих сокетах/TCP) содержит четверку локальный IP/порт – удаленный IP/порт, один из которых, как правило, является динамическим.

Динамический порт назначается только на время соединения. После завершения сеанса соединения порт снова становится свободен для использования, хотя в большинстве реализаций просто происходит увеличение номера последнего использованного порта на единицу вплоть до исчерпания всего диапазона динамических портов.

Тест 2 -----

Тема: 6. Понятие тонкого и толстого клиента. Слои: представление, контроллер, бизнес-слой, DAO. Уровни соединения клиента и сервера в типовом приложении. Прикладной протокол как элемент взаимодействия слоев клиент-серверной архитектуры. Проектирование протоколов распределенных систем.

7. Как понятие бизнес-сущность относится к понятиям: модель предметной области, таблица БД, бизнес-объект, DAO-объект? 2

Модель предметной области описывает сущности и связи предметной области, на ее основе: структура базы данных и бизнес-объекты.

Древовидная структура бизнес-объекта представляет: бизнес-сущности, специфицированные в классах связи, универсальный механизм DAO/ORM работы по связям.

Бизнес-объект – класс представления бизнес-сущности в программе, удобный для использования как единое целое. Класс бизнес-объекта наследует или делегирует класс бизнес-сущности и содержит ссылки на связанные с ним классы бизнес-сущностей или бизнес-объектов

Тест 3 -----

Тема: 14. RestAPI. Основные положения. Клиентская библиотека Retrofit.

3. Почему в RestAPI не может быть реализована полиморфная реализации методов доступа к объекту по имени сущности и ее id в БД? По «идейным» соображениям или способу реализации? 2

В RestAPI данные передаются в формате JSON, соответственно полиморфизм передаваемых данных отсутствует в реализации. По этой причине требуется дублирование функций API для родственных классов.

Тест 4 -----

Тема: 7. Проектирование протокола. Стратегия взаимодействия клиента с сервером. Архитектурные аспекты проектирования протокола. Использование конечных автоматов (диаграмм состояний) в протоколах

7. Определите сущность, достоинства и недостатки стратегии: распределенная подписка клиентов на события, принимаемые сервером и создаваемые его службами.

Тест 5 -----

Тема: 10. Классы Socket и ServerSocket, установление и разрыв соединения

4. Почему на каждое открытое соединение серверное приложение использует отдельный поток (Thread)? 1

Так как соединения устанавливаются во время нахождения запросов в очереди, то клиент может начать передачу данных, когда серверное приложение еще не получило сокет из очереди. Этот эффект зависит от реактивности серверного приложения, что определяется его архитектурой. Если оно работает в режиме транзакций: «прием соединения – запрос – ответ – разрыв соединения», то такой механизм позволяет сделать параллельными процессы приема. Если же серверной приложение является многопоточным, то оно обладает высокой реактивностью, поскольку максимально быстро запускает отдельный поток на каждое принятое соединение и не ждет в нем данных.

Тест 6 -----

Тема: 9. Аспекты проектирования протокола. CheckList понимания протокола.

8. К какому аспекту реализации протокола (Функциональный, структурный, алгоритмический, процессный) относится утверждение (обоснуйте): задачей канального уровня является достоверная передача кадров между соседними узлами сети 0

Утверждение относится к структурному аспекту реализации протокола. Структурный аспект реализации протокола – все, что связано с передаваемыми данными – форматы сообщений, программные интерфейсы;

Тест 7 -----

Тема: 4. Основные технологические решения в протоколах низких уровней и их использование в прикладном программировании сетевых приложений

11. В протоколе, использующем двоичную сериализацию сообщение, при передаче одного из видов сообщение пропущен метод вывода одного из полей (writeInt). Опишите возможные ошибки синхронизации и средства защиты от них.

Тест 8 -----

Тема: 15. Контекст соединения. Восстановление соединений. Паттерн «сессия».

Взаимодействие клиент-сервер без последствий. Идентификатор сессии (токен)

Паттерн «сессия», его реализация при восстановлении постоянных соединений и при взаимодействии клиент-сервер без последствий

3. Определите термин «идентификатор сессии - токен». Как он связан с контекстом клиента на сервере? 1

Токен – файл авторизации, генерируемый сервером с использованием секретного ключа, и отправляемый пользователю и хранящийся у пользователя. С каждым запросом клиент отправляет токен на сервер для подтверждения запроса. Токен имеет время действия, при получении токена сервер не проверяет, кто является пользователем. Токен более надежен чем cookies.

Тест 9 -----

Тема: 6. Понятие тонкого и толстого клиента. Слои: представление, контроллер, бизнес-слой, DAO. Уровни соединения клиента и сервера в типовом приложении. Прикладной протокол как элемент взаимодействия слоев клиент-серверной архитектуры.

Проектирование протоколов распределенных систем.

1. Перечислите слои клиент-серверного приложения и виды передаваемых между ними объектов (событий) 2

- слой 0: примитивные элементы интерфейса пользователя и устройства, его обеспечивающие: графический экран, отдельные окна приложений, пиксели, графические примитивы, события, связанные с нажатием/отпусканьем клавиш, кнопок мыши, различных датчиков, движков, сенсоров и т.п.;
- слой 1: графические объекты и управляющие элементы оконного приложения. Сюда входят такие компоненты, как меню, кнопки, списки, текстовые поля, а также события, в них происходящие (выбор пунктов меню, нажатие кнопок, выбор элемента списка, потеря и получение фокуса, клик мышью по элементу управления. Этот слой реализуется библиотекой оконных классов;
- слой 2 – бизнес-логика приложения. В свою очередь может состоять из подслоев, реализующих следующие функции:
 - 2.1. – представление (View), в которое включаются все элементы реализации интерфейса пользователя;
 - 2.2. – логика бизнес-модели - описание «долгосрочного» ее поведения (Controller);
 - 2.3. – объекты бизнес-модели и методы их преобразования, не связанные с долгосрочным поведением (Model);
 - 2.4. – средства сборки объектов бизнес-модели из табличных объектов БД;
- слой 3 – база данных (СУБД). В свою очередь, может состоять из 3 подслоев:
 - 3.1. табличные объекты БД, DAO (data access objects) – объекты доступа к данным;
 - 3.2. библиотека программного доступа к БД;
 - 3.3. сама программная компонента БД (СУБД).
- слой 4 – данные на физическом носителе (файлы).

Тест 10 -----

Тема: 2. Транспортный протокол TCP. Форматы сообщений, структуры данных, функциональность и алгоритмы передачи данных и управления потоком.

7. Что такое полужесткое закрытие соединения, какими свойствами оно обладает. 2

При вызове метода `close` на одной стороне:

- все записанные в выходной поток данные отправляются в сеть, после чего на передающей стороне сокет закрывается (на потоке приема срабатывает исключение `java.net.SocketException`);
- на приемной стороне происходит получение EOF во входном потоке, но сокет остается открытым;
- сокет на приемной стороне закрывается программно вызовом `close`, выполняется процедура встречного закрытия.

Итого = 12