

funbox

Квалификационные задания для **Java** разработчиков

О заданиях

Данные задания были разработаны для облегчения проверки знаний соискателей. Задания разработаны таким образом, что вы можете их выполнить в любое удобное для вас время и в обстановке, которую считаете оптимальной.

Теперь подробнее о заданиях. Задания разделены на 3 категории.

***Level 1** позволяет оценить базовые знания, насколько вы, знаете базовые основы языка, основы программирования и насколько хорошо владеете программами необходимыми для разработки продуктов.*

***Level 2** содержит вопросы, позволяющие оценить глубину ваших знаний в той или иной части языка программирования, а так же узнать о знании смежных технологий.*

*Ну и последняя часть, это **Level 3** в которой собраны вопросы и задачи по разработке масштабируемых высоконагруженных систем, а так же вопросы на знание низкоуровневого устройства технологий.*

Ответы на задания вы можете составить в свободной форме и прислать нам вместе с вашим резюме (не забыв указать вакансию *Java разработчика*) на почтовый ящик wanted@fun-box.ru. После проверки заданий, мы обязательно сообщим вам о нашем решении.

В общем, это всё. Перейдя на следующую страницу, вы увидите наши вопросы и задания.

Level I

Вопросы данной категории не имеют однозначного ответа который можно было нагуглить или найти в википедии. Если вы еще в ответах укажите, как, при каких обстоятельствах сталкивались с проблемой на собственном опыте и как решили - это будет несомненным плюсом, и будет засчитано как **COMBO X8**.

Q1

Опишите основные плюсы разработки на Java. Как Вы производите сборку (build)?

Q2

Какие технологии Java Enterprise Edition вы чаще всего используете? В чем сложность их использования?

Q3

Расскажите о плюсах использования паттерна MVC. В каких случаях не стоит его использовать? Какие в нем минусы?

Q4

Расскажите об используемых Вами фреймворках (программных каркасах). В чем их плюсы? Для каких задач лучше использовать существующий фреймворк, а когда лучше все написать самому?

Q5

Сборка мусора. Какие проблемы с ней связаны? Какие решения вы бы предложили?

Q6

В чем плюсы использования SVN/CVS/GIT? Какие сложности при работе с ним у вас возникали?

Q7

При работе в команде, каким бы местам в разработке, вы бы уделили большее внимание? Какие бы соглашения (Coding Conventions) вам бы помогли в командной разработке?

Q8

Использование баг-трекеров. В чем плюсы? Расскажите о проблемах использования вами баг-трекеров.

Level II

Вопросы данной категории содержат технические вопросы, в некоторых вам даже придется попробовать себя в роли компилятора. Если вы сможете в ваших ответах на данные вопросы указать, что же еще происходит на низком уровне или почему так происходит - это будет засчитано как **COMBO X16**.

Q1

Объясните почему происходит следующее: Расскажите, в каких случаях, какой контейнер сервлетов лучше использовать:

- Resin
- Tomcat
- Jetty
- WebSphere
- GlassFish
- JBoss

Q2

Расскажите о использовании Java Message Service (JMS), какие проблемы могут возникнуть при работе с ним?

Q3

Каковы плюсы использования Enterprise Java Beans (EJB)? Какие альтернативные технологии можно использовать вместо EJB?

Q4

Есть большая продакшен система. Поступает информация, что одна из основных частей (ORM) начала выдавать ошибки. Вам нужно эти ошибки исправить. ORM система работает с базой C-Store, используя C++ код, через JNI.

- Опишите как Вы начнете анализ места генерации ошибок.
- Какие варианты временного устранения неполадки (костыля) Вы можете предложить?

Q5

Нужно написать прослойку между почтовым сервером и front-end приложением (Flash AS3 Application). Опишите следующие моменты:

- Какой формат обмена данными вы бы использовали, для минимального трафика-обмена (по умолчанию считаем, front-end сможет читать абсолютно любой формат)?
- В чем плюсы выбранного вами формата?
- Какие бы технологии (сервера/фреймворки/утилиты) вы использовали?

Level III

В данной категории содержатся задачи и описание реально возникающих проблем. Постарайтесь придумать и описать ваше решение данных задач. Чем более детализовано будет решение, тем лучше! Вы можете искать решения в интернете, гуглить, читать википедию и так далее, но помните, что вероятнее всего в будущем вам придется столкнуться с такими задачами. Если вы решите все эти задачи, то вы - **TRUE hardcore Java Developer**.

Q1

Есть проект, суть которого в продаже автомобилей. Требования у заказчиков такие: версионность данных(как Wikipedia), возможность расширения моделей данных (можно добавить к описанию автомобиля кастомное свойство, например наличие модинга). Опишите следующие моменты:

- Какую базу данных лучше всего использовать?
- Как реализовать версионность в данном случае?
- Как реализовать возможность расширения моделей?
- Какова будет конечная структура базы данных?
- Какие сложности могут возникнуть в реализации проекта?

Q2

Планируется проект, рассчитанный на большое количество информации, для этого изначально планируется использовать более 6 серверов с MySQL базами данных (есть возможность докупить любое количество серверов). Опишите следующие моменты:

- Как распределить нагрузку между всеми серверами?
- Как реализовать максимальную стабильность работы серверов?
- Как можно снизить загрузку серверов?
- Оптимально ли использовать MySQL? Каковы плюсы и минусы использования?

Q3

Поступило предложение заказчика, на создания аналога сервиса микроблогинга Twitter. На вас ложится задача разработки первичной версии архитектуры проекта. По-умолчанию считаем, что заказчик готов предоставить неограниченные средства. Опишите следующие моменты:

- Какую конфигурацию программной части вы бы составили для проекта (Операционная система, языки программирования, база данных, фреймворки или сторонние разработки)? Опишите в чем плюсы вашей конфигурации.
- Какие слабые стороны возможны у данного проекта? Какие решения Вы можете предложить?
- Опишите схему внутренней работы проекта.

Спасибо за время потраченное на выполнение заданий!