README.md

Back-End Developer Interview Questions

This page has been translated to Chinese by monklof.

I started writing down this list as a personal reminder of topics I had the chance to discuss with colleagues and friends, and that I wanted to deepen...

I'm not a big fan of asking technical questions in job interviews: I rather prefer to sit together with candidates in front of some real code, hands on the keyboard, facing a real problem, and have a full day of pair programming, hopefully rotating with all the other team members. Yet, I feel some technical questions could be a good starting point to begin an engaging and nice conversation, and this can be useful to get a deeper knowledge of each others.

This repo collects a number of back end related questions that can be used when vetting potential candidates. It is by no means recommended to use every single question on the same candidate: that would take hours, and would have no sense at all, as they cover a too broad set of topics for a single developer's to possibly know. Browse the section you find more relevant for your context, and pick the questions that give you more ideas on the conversation to have.

Notice

Most of the questions are open-ended, and some of them just don't have a *right* or a *wrong* answer. On the contrary, they are intended to be used as the starting point for a conversation that hopefully tells you more about the person's capabilities than a straight answer would. Personally, I would even choose the questions whose answers are not yet clear to me.

Again, I stress that just asking questions is hardly sufficient. Complete the interview with a long pair programming session with your candidates: it is one of the best opportunities to know each others' style and approach and to let candidates know some details about their future day job.

This project is admittedly inspired by Front-end Job Interview Questions by @darcyclarke

Where are the answers?

Sooner or later I will complete it with the relative answers. Feel free to contribute, it would be highly appreciated!

Table of Contents

    - Death by Meetings
    - Late Projects
    - Agile Manifesto
    - If I were the CTO
    - PMs
    - Team Organization
    - Turn Over
    - Qualities
    - 3 Things about Code
    - 1 Month's Revolution

- Questions about logic and algorithms

    - FIFO with LIFO
    - Stack Overflow
    - Tail Recursive n!
    - REPL
    - Defragger
    - Mazes
    - Memory Leaks
    - PRNG
    - Garbage Collecting
    - Queues
    - Simple Web Server
    - Sorting Huge Files
    - Duplicates

- Questions about Software Architecture

    - No Cache
    - Event-Driven Architecture
    - Readibility
    - Emergent and Evolutionary
    - Scale-Out, Scale-Up
    - Failures User Sessions
    - CQRS
    - n-tier

    - Unicode
    - Defending Monoliths
    - Professional Developers
    - It's an art
    - People who like this also like...
    - Corporations vs Startups
    - I'm proud of

- Questions based on snippets of code

    - Beware the closure
    - Type Erasure
    - Memory Leak
    - Kill the switch
    - Kill the if
    - Kill the if-chain

- Bill Gates Style Questions

    - Mirrors
    - Clones
    - Revert
    - Quora
    - Cobol
    - 10 years
    - Fire me
    - From scratch
    - Telling lies
    - Your past self

[↑] Questions about Design Patterns:

Globals Are Evil

Why are global and static objects evil? Can you show it with a code example?
Resources

Inversion of Control

Tell me about Inversion of Control and how it improves the design of code.
Resources

Law of Demeter

The Law of Demeter (the Principle of Least Knowledge) states that each unit should have only limited knowledge about other units and it should only talk to its immediate friends (sometimes stated as "don't talk to strangers").
Would you write code violating this principle, show why it is a bad design and then fix it?
Resources

Active-Record

Active-Record is the design pattern that promotes objects to include functions such as Insert, Update, and Delete, and properties that correspond to the columns in some underlying database table. In your opinion and experience, which are the limits and pitfalls of the this pattern?
Resources

Data-Mapper

Data-Mapper is a design pattern that promotes the use of a layer of Mappers that moves data between objects and a database while keeping them independent of each other and the mapper itself. On the contrary, in Active-Record objects directly incorporate operations for persisting themselves to a database, and properties corresponding to the underlying database tables. Do you have an opinion on those patterns? When would you use one instead of the other?
Resources

Billion Dollar Mistake

Why is it often said that the introduction of `null` is a "billion dollar mistake"? Would you discuss the techniques to avoid it, such as the Null Object Pattern introduced by the GOF book, or Option types?
Resources

Inheritance vs Composition

Many state that, in Object-Oriented Programming, composition is often a better option than inheritance. What's you opinion?
Resources

Anti-corruption Layer

What is an Anti-corruption Layer?
Resources

Singleton

Singleton is a design pattern that restricts the instantiation of a class to one single object. Writing a Thread-Safe Singleton class is not so obvious. Would you try?

Data Abstraction

The ability to change implementation without affecting clients is called Data Abstraction. Produce an example violating this property, then fix it.
Resources

Dont't Repeat Yourself

Write a snippet of code violating the Don't Repeat Yourself (DRY) principle. Then, fix it.
Resources

Dependency Hell

How would you deal with Dependency Hell?
Resources

Goto is Evil

Is goto evil? You may have heard of the famous paper "Go To Statement Considered Harmful" by Edsger Dijkstra, in which he criticized the use of the `goto` statement and advocated structured programming instead. The use of `goto` has always been controversial, so much that even Dijkstra's letter was criticized with articles such as "'GOTO Considered Harmful' Considered Harmful". What's your opinion on the use of `goto` ?
Resources

Robustness Principle

The robustness principle is a general design guideline for software that recommends "*be conservative in what you send, be liberal in what you accept*". It is often reworded as "*be a tolerant reader and a careful writer*". Would you like to discuss the rationale of this principle?

Separation of Concerns

Separation of Concerns is a design principle for separating a computer program into distinct areas, each one addressing a separate concern. There are a lot of different mechanisms for achieving Separation of Concerns (use of objects, functions, modules, or patterns such as MVC and the like). Would you discuss this topic?

[↑] Questions about Code Design:

High Cohesion, Loose Coupling

It is often said that one of the most important goals in Object-Oriented Design (and code design in general) is to have High Cohesion and Loose Coupling. What does it mean? Why is it that important and how is it achieved?

Index 0

Why do array indexes start with '0' in most languages?

TDD

How do tests and TDD influence code design?

DRY violation

Write a snippet of code violating the Don't Repeat Yourself (DRY) principle. Then, explain why it is a bad design, and fix it.

Cohesion vs Coupling

What's the difference between cohesion and coupling?

Refactoring

What is refactoring useful for?

Code comments

Are comments in code useful? Some say they should be avoided as much as possible, and hopefully made unnecessary. Do you agree?

Design vs Architecture

What is the difference between design and architecture?

Early Testing

In TDD, why are tests written before code?

Multiple Inheritance

C++ supports multiple inheritance, and Java allows a class to implement multiple interfaces. What impact does using these facilities have on orthogonality? Is there a difference in impact between using multiple inheritance and multiple interfaces? Is there a difference between using delegation and using inheritance? [This question is from The Pragmatic Programmer, by Andrew Hunt and David Thomas]

Domain Logic in Stored Procedures

What are the pros and cons of holding domain logic in Stored Procedures?

OOP took over the world

In your opinion, why has Object-Oriented Design dominated the market for so many years?

Bad Design

What would you do to understand if your code has a bad design?

[↑] Questions about Languages:

3 worst defects

Tell me the 3 worst defects of your preferred language

Functional Programming

Why is there a rising interest on Functional Programming?

Closures

What is a closure, and what is useful for? What's in common between closures and classes?

Generics

What are generics useful for?

High-Order Functions

What are higher-order functions? What are they useful for? Write one, in your preferred language.

Loops and Recursion

Write a loop, then transform it into a recursive function, using only immutable structures (i.e. avoid using variables). Discuss.

Functions as First-Class Citizens

What does it mean when a language treats functions as first-class citizens?

Anonymous Functions

Show me an example where an anonymous function can be useful.

Static and Dynamic typing

There are a lot of different type systems. Let's talk about static and dynamic type systems, and about strong and weak ones. You surely have an opinion and a preference about this topic. Would you like to share them, and discuss why and when would you promote one particular type system for developing an enterprise software?

Namespaces

What are namespaces useful for? Invent an alternative.

Language Interoperability

Talk about interoperability between Java and C# (in alternative, choose 2 other arbitrary languages)

Hate of Java

Why do many software engineers not like Java?

Good and Bad Languages

What makes a good language good and a bad language bad?

Rerefential Transparency

Write two functions, one referentially transparent and the other one referentially opaque. Discuss.

Stack and Heap

What is a stack and what is a heap? What's a stack overflow?

Functions as First Class Citizens

Why is it important that in a language functions are first class citizens?

Pattern Matching

Some languages, especially the ones that promote a functional approach, allow a technique called pattern matching. Do you know it? How is pattern matching different from switch clauses?

Exceptions

Why do some languages have no exceptions by design? What are the pros and cons?

Variant and Contravariant Inheritance

If `Cat` is an `Animal`, is `TakeCare<Cat>` a `TakeCare<Animal>`?

Constructors and Interfaces

In Java, C# and many other languages, why are constructors not part of the interface?

Node.js

In the last years there has been a lot of hype around Node.js. What's your opinion on using a language that was initially conceived to run in the browser in the backend?

Java and time-traveling

- Pretend you have a time machine and pretend that you have the opportunity to go to a particular point in time during Java's (or C#, Python, Go or whatever) history, and talk with some of the JDK architects. What would you try to convince them of? Removing checked exceptions? Adding unsigned primitives? Adding multiple-inheritance?

[↑] Questions about Web development:

3rd Party Cookies

Why are first-party cookies and third-party cookies treated so differently?

API Versioning

How would you manage Web Services API versioning?

SPAs

From a backend perspective, are there any disadvantages or drawbacks on the adoption of Single Page Applications?

Statelessness

Why do we usually put so much effort for having stateless services? What's so good in stateless code and why and when is statefulness bad?

REST vs SOAP

REST and SOAP: when would you choose one, and when the other?

MVC and MVVM

In web development, Model-View Controller and Model-View-View-Model approaches are very common, both in the backend and in the frontend. What are they, and why are they advisable?

[↑] Questions about Databases:

DB Migrations

How would you migrate an application from a database to another, for example from MySQL to PostgreSQL? If you had to manage that project, which issues would you expect to face?
Resources

NULL is special

Why do databases treat null as a so special case? For example, why does `SELECT * FROM table WHERE field = null` not match records with null `field` in SQL?
Resources

ACID

ACID is an acronym that refers to Atomicity, Consistency, Isolation and Durability, 4 properties guaranteed by a database transaction in most database engines. What do you know about this topic? Would you like to elaborate?
Resources

Schema Migrations

How would you manage database schema migrations? That is, how would you automate changes to database schema, as the application evolves, version after version?
Resources

Lazy Loading

How is lazy loading achieved? When is it useful? What are its pitfalls?
Resources

N+1 Problem

The so called "N + 1 problem" is an issue that occurs when code needs to load the children of a parent-child relationship with a ORMs that have lazy-loading enabled, and that therefore issue a query for the parent record, and then one query for each child record. How to fix it?
Resources

Slowest Queries

How would you find the most expensive queries in an application?
Resources

Normalization

In your opinion, is it always needed to use database normalization? When is it advisable to use denormalized databases?
Resources

Blue/Green Deployment

Of of the Continuous Integration's techniques is called Blue-Green Deployment: it consists in having two production environments, as identical as possible, and in performing the deployment in one of them while the other one is still operating, and than in safely switching the traffic to the second one after some convenient testing. This technique becomes more complicated when the deployment includes changes to the database structure or content. I'd like to discuss this topic with you.
Resources

[↑] Questions about NoSQL:

Eventual Consistency

What is eventual consistency?

CAP Theorem

Brewer's Theorem, most commonly known as the CAP theorem, states that in the presence of a network partition (the P in CAP), a system's designer has to choose between consistency (the C in CAP) and availability (the A in CAP). Can you think about examples of CP, AP and CA systems?

NoSQL

How would you explain the recent rise in interest in NoSQL?

NoSQL and Scalability

How does NoSQL tackle scalability challenges?

Document and Relational DBs

When would you use a document database like MongoDB instead of a relational database like MySQL or PostgreSQL?

[↑] Questions about code versioning:

Branching in HG and in Git

Why is branching with Mercurial or git easier than with SVN?

DVCS

What are the pros and cons of distributed version control systems like Git over centralized ones like SVN?

GitFlow and GitHubFlow

Could you describe GitHub Flow and GitFlow workflows?

Rebase

What's a rebase?

Merging in HG and in Git

Why are merges easier with Mercurial and Git than with SVN and CVS?

[↑] Questions about Concurrency:

Why?

Why do we need concurrency, anyway? Explain.

Testing Concurrency

Why is testing multithreaded/concurrent code so difficult?

Race Conditions

What is a race condition? Code an example, using whatever language you like.

Deadlocks

What is a deadlock? Would you be able to write some code that is affected by deadlocks?

Process Starvation

What is process starvation? If you need, let's review its definition.

Free Algorithm

What is a wait free algorithm?

[↑] Questions about Distributed Systems:

Testing Distributed Systems

How would you test a distributed system?

Async Communication

When would you apply asynchronous communication between two systems?

Pitfalls of RPC

What are the general pitfalls of remote procedure calls?

Design of Distributed Systems

If you are building a distributed system for scalability and robustness, what are the different things you'd think of if you are working in a closed and secure network environment versus when you are working in a geographically distributed and public system?

Fault Tolerance

How would you manage fault tolerance in a web application? What about in a desktop one?

Failures

How would you deal with failures in a distributed system?

Network Partitions

Let's talk about the several approaches to reconciliation after network partitions.

Fallacies of Distributed Computing

What are the fallacies of distributed computing?

Request/Reply vs Publish/Subscribe

When would you use request/reply and when publish/subscribe?

Implement Transactions

Suppose the system you are working on does not support transactionality. How would you implement it from scratch?

[↑] Questions about Software Lifecycle and Team Management:

Agility

What is agility?

Legacy Code

How would you deal with legacy code?

Legacy Code ELI5

Say I'm your project manager, and I'm no expert in programming. Would you try explaining to me what legacy code is and why should I care about code quality?

Sell me Kanban

I'm the CEO of your company. Explain to me Kanban and convince me to invest in it.

Agile vs Waterfall

What is the biggest difference between Agile and Waterfall?

Death by Meetings

Being a team manager, how would you deal with the problem of having too many meetings?

Late Projects

How would you manage a very late project?

Agile Manifesto

"*Individuals and interactions over processes and tools*" and "*Customer collaboration over contract negotiation*" comprise half of the values of the Agile Manifesto. Discuss

If I were the CTO

Tell me what decisions would you take if you could be the CTO of your Company.

PMs

Are program managers useful?

Team Organization

Organize a development team using flexible schedules (that is, no imposed working hours) and "take as you need" vacation policy

Turn Over

How would you manage a very high turn over and convince developers not to leave the team, without increasing compensation? What could a company improve to make them stay?

Qualities

What are the top 3 qualities you look for in colleagues, beyond their code?

3 Things About Code

What are the top 3 things you wish non-technical people knew about code?

1 month's revolution

Imagine your company gives you 1 month and some budget to improve your and your colleagues' daily life. What would you do?

[↑] Questions about logic and algorithms:

FIFO with LIFO

Make a FIFO queue using only LIFO stacks. Then build a LIFO stack using only FIFO queues.

Stack Overflow

Write a snippet of code affected by a stack overflow.

Tail Recursive n!

Write a tail-recursive version of the factorial function.

REPL

Using your preferred language, write a REPL that echoes your inputs. Evolve it to make it an RPN calculator.

Defragger

How would you design a "defragger" utility?

Mazes

Write a program that builds random mazes.

Memory Leaks

Write a sample program that produces a memory leak.

PRNG

Generate a sequence of unique random numbers.

Garbage Collecting

Write a simple garbage collection system.

Queues

Write a basic message broker, using whatever language you like.

Simple Web Server

Write a very basic web server. Draw a road map for features to be implemented in the future.

Sorting Huge Files

How would you sort a 10GB file? How would your approach change with a 10TB one?

Duplicates

How would you programmatically detect file duplicates?

[↑] Questions about Software Architecture:

No Cache

When is a cache not useful or even dangerous?

Event-Driven Architecture

Why does Event-Driven Architecture improve scalability?

Readibility

What makes code readable?

Emergent and Evolutionary

What is the difference between emergent design and evolutionary architecture?

Scale-Out, Scale-Up

Scale out vs scale up: how are they different? When to apply one, when the other?

Failures User Sessions

How to deal with failover and user sessions?

CQRS

What is CQRS (Command Query Responsibility Segregation)? How is it different from the oldest Command-Query Separation Principle?

n-tier

The so called "multitier architecture" is an approach to design a client–server system aimed to keep physically and logically separated presentation, application processing, data management and other functions. The most widespread of the multitier architectures is the three-tier architecture. Would you discuss the pros and cons of such an approach?

Scalability

How would you design a software system for scalability?

C10K

Someone gave the name "The "C10k problem" to the problem of optimising network sockets to handle over 10.000 open connections at once. While handling 10.000 concurrent clients is not the same as handling 10.000 open connection, the context is similar. It's a tough challenge anyway, and no one is expected to know every single detail to solve it. It may be interesting to discuss the strategies you know to deal with that problem. Would you like to try?

P2P

How would you design a decentralized (that is, with no central server) P2P system?

CGI

You may recall that Common Gateway Interface (CGI) is a standard protocol for web servers to execute programs (CGI scripts) that execute as Command-line programs on a server, and that dynamically generate HTML pages when invoked by a HTTP request. Perl and PHP used to be common languages for such scripts. In CGI, a HTTP request generally causes the invocation of a new process on the server, but FastCGI, SCGI and other approaches improved the mechanism, raising the performance, with techniques such as preforking processes. Can you discuss why CGI became obsolete, and was instead replaced with other architectural approaches?

Vendor Lock-in

How would you defend the design of your systems against vendor Lock-in?

Pub/Sub

What are the disadvantages of the publish-subscribe pattern at scale?

CPUs

What's new in CPUs since the 80s, and how does it affect programming?

Performance

In which part of the lifecycle of a software performance should be taken in consideration, and how?

DDOS

How could a denial of service arise not maliciously but due to a design or architectural problem?

Performance and Scalability

What's the relationship between performance and scalability?

Tight Coupling

When is it OK (if ever) to use tight coupling?

Cloud Readiness

What characteristic should a system have to be cloud ready?

Emergent Architecture

Does unity of design imply an aristocracy of architects? Putting it simple: can good design emerge from a collective effort of all developers?

Design, Architecture, Functionality, Aesthetic

What's the difference between design, architecture, functionality and aesthetic? Discuss.

[↑] Questions about Service Oriented Architecture and Microservices:

Long-lived Transactions

Why, in a SOA, long-lived transactions are discouraged and sagas are suggested instead?

SOA and Micro Services

What are the differences between SOA and microservice?

Versioning and Breaking Changes

Let's talk about web services versioning, version compatibility and breaking changes.

Sagas and compensations

What's the difference between a transaction and a compensation operation in a saga, in SOA?

Too Micro

When is a microservice too micro?

Micro Services Architecture

What are the pros and cons of microservice architecture?

[↑] Questions about Security:

Security by Default

How do you write secure code? In your opinion, is it one of the developer's duties, or does it require a specialized role in the company? And why?

Don't invent Cryptography

Why is it said that cryptography is not something you should try to invent or design yourself?

2-FA

What is two factor authentication? How would you implement it in an existing web application?

Sensible Data in Logs

If not carefully handled, there is always a risk of logs containing sensitive information, such as passwords. How would you deal with this?

SQL Injection

Write down a snippet of code affected by SQL injection and fix it.

Detect SQL Injection

How would it be possible to detect SQL injection via static code analysis? I don't expect you to write an algorithm capable of doing this, as it is probably a huge topic, but let's discuss a general approach.

XSS

What do you know about Cross-Site Scripting? If you don't remember it, let's review online its definition and let's discuss about it.

Cross-Site Forgery Attack

What do you know about Cross-Site Forgery Attack? If you don't remember it, let's review online its definition and let's discuss about it.

HTTPS

How does HTTPS work?

MITM Attack

What's a man-in-the-middle Attack, and why does HTTPS help protect against it?

Stealing Sessions

How can you prevent the user's session from being stolen? Chances are you remember what session or cookie hijacking is, otherwise let's read its Wikipedia page together.

[↑] General Questions:

Why FP?

Why does functional programming matter? When should a functional programming language be used?

Browsers

How do companies like Microsoft, Google, Opera and Mozilla profit from their browsers?

TCP Sockets

Why does opening a TCP socket have a large overhead?

Encapsulation

What is encapsulation important for?

Real-time systems

What is a real-time system and how is it different from an ordinary system?

Real-time and memory allocation

What's the relationship between real-time languages and heap memory allocation?

Immutability

Immutability is the practice of setting values once, at the moment of their creation, and never changing them. How can immutability help write safer code?

Mutable vs Immutable

What are the pros and cons of mutable and immutable values.

Object-Relational Impedance Mismatch

What's the Object-Relational impedance mismatch?

Sizing a Cache

Which principles would you apply to define the size of a cache?

TCP and HTTP

What's the difference between TCP and HTTP?

Client-Side vs Server-Side

What are the tradeoffs of client-side rendering vs. server-side rendering?

Reliable and non-reliable channels

How could you develop a reliable communication protocol based on a non-reliable one?

Million-Dollar Mistake

Tony Hoare who invented the null reference once said "*I call it my billion-dollar mistake*" since it led to "*innumerable errors, vulnerabilities, and system crashes, which have probably caused a billion dollars of pain and damage in the last forty years*". Imagine you want to remove the possibility to have null references in your preferred language: how would you achieve this goal? What consequences could this have?

[↑] Open Questions:

Resistance to Change

Why do people resist change?

Threading ELI5

Explain threads to your grandparents

Innovation and Predictability

As a software engineer you want both to innovate and to be predictable. How those two goals can coexist in the same strategy?

Good Code

What makes good code good?

Stradming

Explain streaming and how you would implement it.

1 Week Improvement

Say your company gives you one week you can use to improve your and your colleagues' lifes: how would you use that week?

Learnt this week

What did you learn this week?

Aestheric

There is an aesthetic element to all design. The question is, is this aesthetic element your friend or your enemy?

Last 5 books

List the last 5 books you read.

Introducing CI/CD

How would you introduce Continuous Delivery in a successful, huge company for which the change from Waterfall to Continuous Delivery would be not trivial, because of the size and complexity of the business?

Reinvent the Wheel

When does it make sense to reinvent the wheel?

Not Invented Here

Let's have a conversation about "*reinventing the wheel*", the "*not invented here syndrome*" and the "*eating your own food*" practice

Next Thing to Automate

What's the next thing you would automate in your current workflow?

Code is Hard

Why is writing software difficult? What makes maintaining software hard?

Green Fields and Brown Fields

Would you prefer working on green field or brown field projects? Why?

Type "Google.com"

What happens when you type google.com into your browser and press enter?

While idle

What does an operating system do when it has got no custom code to run, and therefore it looks idle? I would like to start a discussions about interrupts, daemons, background services, polling, event handling and so on.

Unicode

Explain Unicode and database transactions to a 5 year old child.

Defending Monoliths

Defend the monolithic architecture.

Professional Developers

What does it mean to be a "professional developer"?

It's an art

Is developing software an art, a craftsmanship or an engineering endeavour? Your opinion.

People who like this also like...

"People who like this also like... ". How would you implement this feature in an e-commerce shop?

Corporations vs Startups

Why are corporations slower than startups in innovating?

I'm proud of

What have you achieved recently that you are proud of?

[↑] Questions about snippets of code:

Beware the closure

What's the output of this Javascript function?

```
function hookupevents() {
  for (var i = 0; i < 3; i++) {
    document.getElementById("button" + i)
      .addEventListener("click", function() {
        alert(i);
      });
  }
}
```

Type Erasure

About Type Erasure, what's the output of this Java snippet, and why?

```
ArrayList<Integer> li = new ArrayList<Integer>();
ArrayList<Float> lf = new ArrayList<Float>();
if (li.getClass() == lf.getClass()) // evaluates to true
  System.out.println("Equal");
```

Memory Leak

Can you spot the memory leak?

```
public class Stack {
    private Object[] elements;
    private int size = 0;
    private static final int DEFAULT_INITIAL_CAPACITY = 16;
```

```
    public Stack() {
        elements = new Object[DEFAULT_INITIAL_CAPACITY];
    }

    public void push(Object e) {
        ensureCapacity();
        elements[size++] = e;
    }

    public Object pop() {
        if (size == 0)
            throw new EmptyStackException();
        return elements[--size];
    }

    /**
     * Ensure space for at least one more element, roughly
     * doubling the capacity each time the array needs to grow.
     */
    private void ensureCapacity() {
        if (elements.length == size)
            elements = Arrays.copyOf(elements, 2 * size + 1);
    }
}
```

Kill the switch

`if` s and in general conditional statements lead to procedural and imperative programming. Can you get rid of this `switch` and make this snippet more object oriented?

```
public class Formatter {

    private Service service;

    public Formatter(Service service) {
        this.service = service;
    }

    public String doTheJob(String theInput) {
        String response = service.askForPermission();
        switch (response) {
        case "FAIL":
            return "error";
        case "OK":
            return String.format("%s%s", theInput, theInput);
        default:
            return null;
        }
```

```
    }
}
```

Kill the if

Can you get rid of these `if`s and make this snippet of code more object oriented?

```
public class TheService {
    private final FileHandler fileHandler;
    private final FooRepository fooRepository;

    public TheService(FileHandler fileHandler, FooRepository fooRepository) {
        this.fileHandler = fileHandler;
        this.fooRepository = fooRepository;
    }

    public String Execute(final String file) {

        final String rewrittenUrl = fileHandler.getXmlFileFromFileName(file);
        final String executionId = fileHandler.getExecutionIdFromFileName(file);

        if (executionId.equals("") || rewrittenUrl.equals("")) {
            return "";
        }

        Foo knownFoo = fooRepository.getFooByXmlFileName(rewrittenUrl);

        if (knownFoo == null) {
            return "";
        }

        return knownFoo.DoThat(file);
    }
}
```

Kill the if-chain

How would you refactor this code?

```
function()
{
    HRESULT error = S_OK;

    if(SUCCEEDED(Operation1()))
    {
        if(SUCCEEDED(Operation2()))
        {
```

```
            if(SUCCEEDED(Operation3()))
            {
                if(SUCCEEDED(Operation4()))
                {
                }
                else
                {
                    error = OPERATION4FAILED;
                }
            }
            else
            {
                error = OPERATION3FAILED;
            }
        }
        else
        {
            error = OPERATION2FAILED;
        }
    }
    else
    {
        error = OPERATION1FAILED;
    }

    return error;
}
```

Resources

[↑] Bill Gates Style Questions:

This section collects some weird questions along the lines of the Manhole Cover Question.

Mirrors

What would happen if you put a mirror in a scanner?

Clones

Imagine there's a perfect clone of yourself. Imagine that that clone is your boss. Would you like to work for him/her?

Revert

Interview me.

Quora

Why are Quora's answers better than Yahoo Answers' ones?

Cobol

Let's play a game: defend Cobol against modern languages, and try to find as many reasonable arguments as you can.

10 years

Where will you be in 10 years?

Fire me

You are my boss and I'm fired. Inform me.

From scratch

I want to refactor a legacy system. You want to rewrite it from scratch. Argument. Then, switch our roles.

Telling lies

Your boss asks you to lie to the company. What's your reaction?

Your past self

If you could travel back in time, which advice would you give to your younger self?

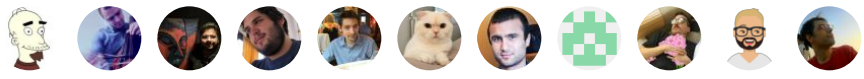
Releases

No releases published

Packages

No packages published

Contributors 17



+ 6 contributors