Sample interview questions

Tell me about yourself?

Prepare a story about yourself in advance, it may contain 3 parts:

1. What you were doing in the past or how you started programming
2. What are you working on now and
3. What are your plans for the future

Tell me about your most inspiring project?

Here we need a story about some project you have done with an accent on some technical challenges and complexities. For most of the interviewers your project is boring and challenges are not challenging, but this is your story and you should clearly explain what was challenging and inspiring for you. Example: at my previous job I significantly contributed to a decoupling of legacy poorly documented monolith and its migration to the cloud. Reading, understanding legacy code and some outdated technologies was challenging and not easy, but I managed to deep dive, understand the code base and deprecate old code and build new functionality.

Similar questions:

1. Tell me what was the impact you have created at your previous job?

Tell me something cool about yourself, which is not in your resume?

Here we expect to hear some extracurricular activity, for example: participation in Open Source project, education, teaching experience, volunteering (with some technical skills), some tech project you do at home for fun.

Are you a go to person?

Here we want to know how far you support your colleagues and help them to grow professionally. A go-to person is someone who knows a bit more than others plus ready to share the knowledge. It is a tricky question because some knowledge holders in the team tend to become a so called “bus factor”, so to say if they a hit by the bus the knowledge is lost. So it is important to underline that you share the knowledge but you let other people do the work on their own and you do not try to do all work by yourself. Helping people and doing their job are very different things.

Why should we hire you?

The question may sound stupid: the company is looking for an engineer and here you are letting them choose, but in fact they want to see whether you took time to understand what the company is about, make up your mind about their tasks and what the company wants to make and thought how you can help them to achieve their goals. Therefore, check company's website before interview, understand what is their product, think what your everyday tasks will be and explain at the interview how will you do this. An example: *I see that you are expanding your services to different countries, which means the load on existing services may increase and in case you have monolithic services, it may become difficult to maintain their stability. A migration to microservices is an option. I have an experience in designing RESTFull APIs, building microservices, producing high quality code, covered by tests. I believe this is what you need and where I can be of great help.*

How do you learn?

Companies want to hire people, who are curious and constantly looking forward in improving their skills. Therefore, you should provide a solid explanation how do you learn. Ideally, you should mention learning from more experienced colleagues, taking online courses on Yandex, Coursera, Udemy, Linkedin Learning plus classical books in your area: Effective Java, Clean Code, Design Patterns by Gang of Four, etc. If you say that you took an online course, be ready to say which one exactly, why you took, what you have learned from it in few sentences. It is extremely important to be precise here. "I am watching Youtube videos and read blogs" is the wrong answer. If you are going to say you read Effective Java, read it before the interview at least couple of time.

Why do you want to work for us?

This is similar to some questions above, but here it is enough to say that you like what the company is doing (your values comply with company values) and you think that your skills matches company's expectations.

Where do you see yourself in 5 years?

This one considered as the suppiest question of all time, but it is very easy to answer it wrong and get rejected. If you apply as a junior, the company does not expect you to become a CTO in 5 years. They may want to hire you because they have tasks for you. It is correct to say that you would like to help the company to move ahead and succeed plus you expect the company will let you grow professionally so that you learn from more experienced colleagues. In five years I would like to become a far more technically experienced engineer.

When you work on a project or in the team, how do you know that you are successful?

It is a big question. Below is a sample schema for answering this. You should define success first. Let’s say, success for you is that you are a good team player and technically good engineer, which delivers on time according to stakeholders expectations. How do you know that you are a good team player? I welcome continuous feedback on my work by asking my colleagues how I am doing, I participate in regular 360 feedback rounds, I am constantly working on fixing feedback comments on areas where I may need to improve. On the other hand I have my own KPIs: how many features/projects I do per year? How many comments I receive to my Pull Requests in average? How long does it take for me to develop a feature? How successful are projects which I deliver as a part of the team? I measure all this and make conclusions so to understand whether I am good or not good enough.

You see that a colleague of you opens a low quality pull request and does not react to your comments. What will you do? WIll you just thumb up and let it merge or will you push the peer to fix your comments?

Such questions in general check your attitude towards conflict situation. It is expected to hear from you that you do not go bossy or aggressive. At first, you should try to understand why your colleague is not responsive. Is his code really that bad? Are you sure that you are right in your comments? If so, try to talk to person by asking him direct questions about problems in the code and see his reaction. It might be the case that your comments are not clear enough. Or your colleague cannot formulate his answer because of language barrier or cultural differences and you need to change the approach. It is also fine to attract a 3 party mediator like an engineering lead, principle engineer, ask the team for a help. Stay calm, do not blame anyone, no finger pointing, only discussion and patient coaching (asking questions) and search for a solution, which is the best for the company and the customers.

You have a planned project deadline by the end of the week but you recognized that you need another 3 weeks. What will you do?

The wrong answer in short is “I will work extra hours to fit the deadline”. The correct answer is to accept the situation, calm down and get learnings out of the situation by asking yourself questions like: what could possibly went wrong so that we failed the deadline? should we try to plan better next time? Were we too optimistic about project complexity and our capacity as a team? How can we improve this? When answered all that you approach the customer/manager, share the analysis of the situation and come with new estimates. Important: do not blame people, colleagues or managers. Accept that there is a problem and analyze the situation. You contributed to every failure and it has already happened, so blaming anyone will not help you to fix the missed deadline but will rather distract the relationships in the team.

Tell about a challenge at your former workplace you have overcome and personal achievements you are proud of

Real relevant case. For example you had proposed and implemented the solution that helped to increased development efficiency/quality. Or you managed to grasp the new technology/language in a short term because of new business requirements and so on. But it should be real case and you have to think of the details in advance.

If your boss (technical lead, architect) is proposing the solution that is not optimal in a long-term in your opinion, what would you do?

The right answer - express your opinion in constructive grounded way (maybe it would require several discussion iterations to prove your point of view). But if boss still does not agree - accept his position. Wrong answer - say “I don’t care, it’s his responsibility, I would not say anything”.

Your attitude to overtimes?

The right answer - I understand that sometimes objective situation may require overtimes and I am ok with it. But if it happens too often - it’s symptom of organizational or process issues, that should be fixed at management level.

Please describe your dream job place

Actually you have to be sincere, acceptable answer would be - interesting job content and technology stack relevant to my skills, good team atmosphere of trust and support, room for professional development, desired salary level, to receive a constructive feedback on my performance and so on.
Wrong answer - than more money than better.

Please describe the top 3 things that are the most important for you at work

Similar to question above, but there could be something specific, like family friendly environment with possibility to work home office. Also you can mention what is not acceptable for you like - overtime on regular basis.

Why do you want to change job?

Here the rule is - don’t say bad things about your current employment, disclosure sensitive details or to blame. The answer should be neutral, for example - my current long-term project is going to its end and your offer does look interesting and promising, so I think - maybe it’s a right time to change something in my career?

Describe things you want to improve in yourself.

Here the mistake would be - to be too much sincere. This is a trap to get into confession of own doubts and fears nobody is interested in. The answer should be short and neutral like - I have some issues with time management and/or estimations. But I constantly work on this, to be as much as possible on time and give realistic estimation to rely on.