0. 들어가며

이 문서는 주니어 개발자 혹은 개발자가 되기를 희망하는 입문자 수준을 대상으로 작성한 문서 입니다. 질문을 '잘'하기 위해 알아야 할 사항에 대해 설명합니다.

1. 질문을 '잘'해야 하는 이유!

개발을 하다보면 마땅한 방법이 떠오르지 않거나 내가 알고 있는 방법으로 문제가 해결되지 않을 때가 있습니다. 그럴 때 우리는 누군가에게 질문을 합니다. 그리고 그 질문에 답할 수 있는 사람은 보통 나보다 숙련된 개발자(이하 시니어) 일테고요. 문제는 시니어는 질문자보다 복잡하고 중요한 문제에 매달려 있을 확률이 크다는 겁니다. 시니어는 질문자의 문제를 깊게 생각해줄 여유가 없습니다! 혹은 이미 반복되는 질문에 지쳐있을 수 있습니다. 그래서 정리되지 않은 스무고개식 질문은 시니어를 지치게 해요. 그래서 질문을 명확하게 잘 하는 일이 중요합니다.

1.1. 온라인에서 특히 질문을 '잘'해야 하는 이유

온라인 게시판을 통해 질문과 답변을 주고받을 경우 질문을 잘 하는 것은 특히 중요합니다. 온라인은 질문과 답변사이의 시간 텀이 매우 크게 발생합니다. 운이 좋아 몇분내에 답변이 달릴 수 있지만 2~3일 뒤에 답변이 달릴 수도 있습니다. 때로는 아예 답변을 받지 못할수도 있지요. 그런데 그렇게 긴 기다림끝에 달린 답변이 질문의 내용을 파악할 수 없으니 추가 정보를 요청하는 답변이라면 어떨까요? 질문자가 이 답변을 확인하고 다시 보충 질문을 해야하는데 답변자가 이 질문을 다시 확인할지 아닐지는 장담할 수 없습니다. 마냥 기다리는 수밖에 없습니다. 이런 시간낭비가!

2. 질문 '잘' 하기 왜이리 어려울까?

결론부터 말씀드리면, '내가 알고 있는것을 상대도 알고있다'는 착각 때문입니다. 질문자는 질문을 하기 전에, 1) 목표를 세우고, 2) 코드를 작성한 뒤 3) 의도한 대로 동작하지 않음을 발견하고 4) 질문 합니다. 질문자는 1)부터 4)의 과정을 모두 거쳤지만 답변해야 하는 사람은 어떤가요? 이 과정을 하나도 거치지 않았죠! 보통 처음 질문하시는 분들은 3) 만 제시하곤 합니다. 온라인으로 질문을 종종 해보신 분들은 2), 3)을 제시하며 질문을 합니다. 그래야 사람들이 답변한다는 걸 경험적으로 알기 때문이에요.

답변자는 질문자가 남긴 단서들을 역으로 유추하며 문제의 원인을 찾습니다. 그러나 목적을 알 수 없는 코드를 해석하는 과정은 매우 지루한 과정이고 기력을 소모합니다 그러니 질문을 할땐 1)부터 3)까지 모두 제시하며 질문하는 것이 좋습니다. 어떤 문제를 해결하기 위해 코드를 작성하였고 어떤 시도를 했는지, 그리고 어떤 부분에서 문제가 발생했는지, 문제의 원인이 무엇이라고 생각하고 이를 해결하기 위해 어떤 시도를 했는지, 모두 알려주세요! 이렇게 잘 정리한 질문은 누구나 답변을 달고 싶어집니다!

2.1. 답이 있는 질문은 YES/NO로 답변 할 수 있게 정리하자

질문에는 크게 '답이 있는 질문'과 '답이 없는 질문'이 있습니다. 답이 있는 질문이란 답변자가 YES/NO 로 대답할 수 있는 질문입니다. 이렇게 YES/NO로 답할 수 있도록 질문을 정리하면 시니어가 답변을 할 확률이 큽니다.

답이 없는 문제란 정답이 없고 해답이 있는 문제를 말합니다. 상황에 따라 답이 달라질 수 있는 문제죠. 이런 답이 없는 문제의 경우에는 본인의 상황을 잘 설명하고, 왜 이 선택을 하였는지 '의도'를 설명하며 질문하는것이 중요합니다. 문제의 맥락을 이해하지 못하면 상대가 엉뚱한 대답을 하거나 잘못된 해답을 제시할 수 있습니다.

3. 질문하기 전에 구글링!

앞서 설명한 것처럼 온라인에서 질문하고 답변을 받는 과정은 정성과 시간을 필요로 합니다. 그런데 혼자서 구글링으로 문제를 해결할 수 있다면? 많은 시간을 아낄 수 있습니다! 그래서 구글링을 잘 하는것이 중요합니다. 다음은 몇가지 구글링 팁입니다.

3.1. 검색요령

영어로 검색하세요.

- 'how to ~' 형태로 검색하면 더 나은 결과를 조회할 수 있습니다.
- 예) how to fix nosuchbeandefinitionexception

오류 메시지를 '복붙'하여 검색하세요.

- 오류 메시지가 뜨면 겁먹지 말고 중요한 키워드가 있는지 찾으세요!
- 예) NoClassDefFoundError

구글 검색 옵션을 적극 활용하세요

- site: 특정 사이트에 존재하는 문서내에서만 검색.
  - 예) 하이버네이트 site:inflearn.com
  - 해석) '하이버네이트' 키워드를 인프런 사이트 내에서만 검색.
- "keyword": 큰따옴표, 특정 키워드 반드시 포함하여 조회하기
  - 예) 하이버네이트 "스펙"
  - 해석) 하이버네이트로 검색하되 "스펙" 키워드가 반드시 포함된 결과를 검색.
- -: 마이너스, 불필요한 키워드 제거하기
  - 예) 스프링 -용수철 -봄
  - 해석) 스프링으로 검색하되 '용수철', '봄' 키워드는 제외.
- ~: 물결표, 물결표에 지정한 카테고리 결과만 검색.
  - 예) 재규어 ~자동차

- 해석) 재규어 중 '자동차'와 카테고리만 검색.

4. 질문하기전에 실험!

책으로, 선생님에게 들어 알고 있다 여겼던 많은 지식들. 막상 사용하려면 떠오르지 않는 경우 있으시죠? 알고만 있는 지식은 지혜로 바뀌지 않습니다. 지식은 경험을 통해 지혜가 됩니다. 그러니 궁금한 부분은 직접 실험을 해보세요. 가설을 세우고, 검증 방법을 생각해보세요. 코드를 작성하고 직접 눈으로 확인하세요. 이렇게 경험으로 배운 지혜는 오랜시간이 지나도 절대 잊혀지지 않습니

5. 정리

1) 질문은 구체적으로 명료하게 작성하세요.

2) 답이 있는 질문은 YES/NO로 대답할 수 있게 질문을 정리해 주세요.

3) 질문하기 전에 구글링을 하세요.

4) 직접 실험해보세요.

5) 누군가의 질문에 답변자가 되어주세요. 누군가에게 설명할 때 가장 많이 배웁니다. 그리고 질문 잘하는 방법도 알 수 있으니 일석이조!

6. 좋은 질문 예시

1) https://www.inflearn.com/questions/337571

2) https://www.inflearn.com/questions/352598