BYB - task_25 GOOGLE ANSWERS DOCUMENT

Helder Paixao
helder.paixao@outlook.com

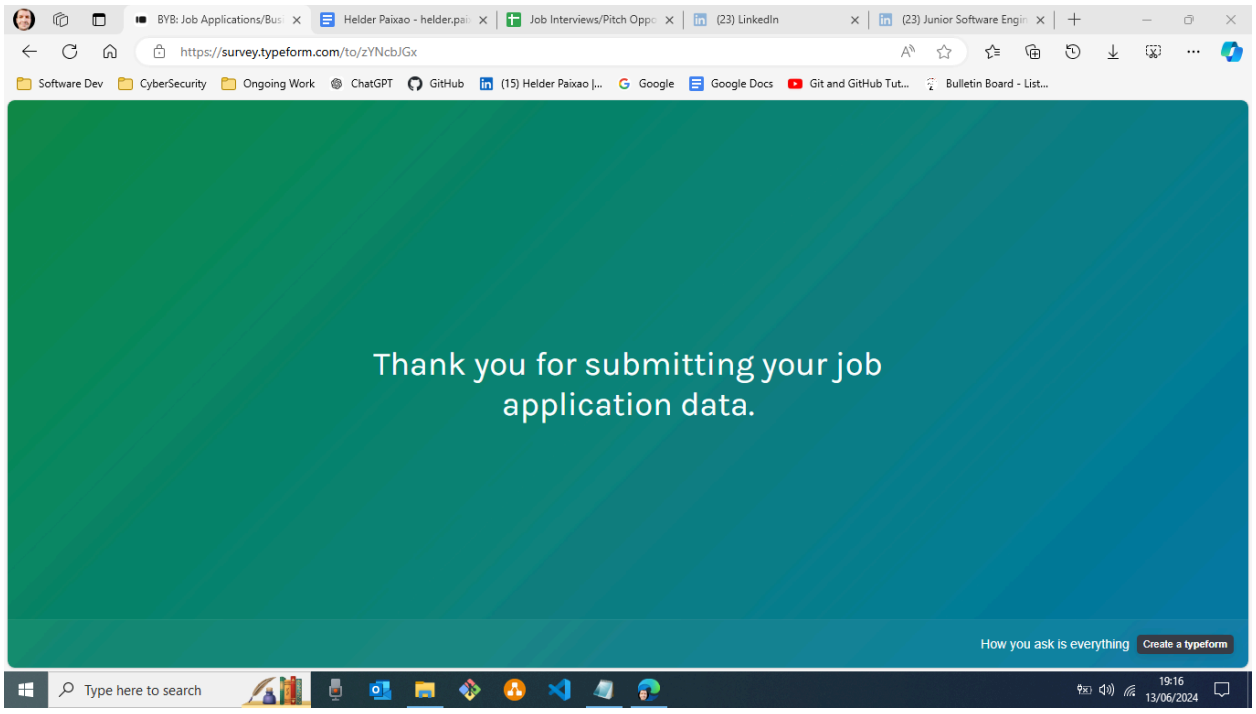
Practical Task 1

Opportunities Tracker - publicly accessible link:

⇒ Job Interviews/Pitch Opportunities Tracker

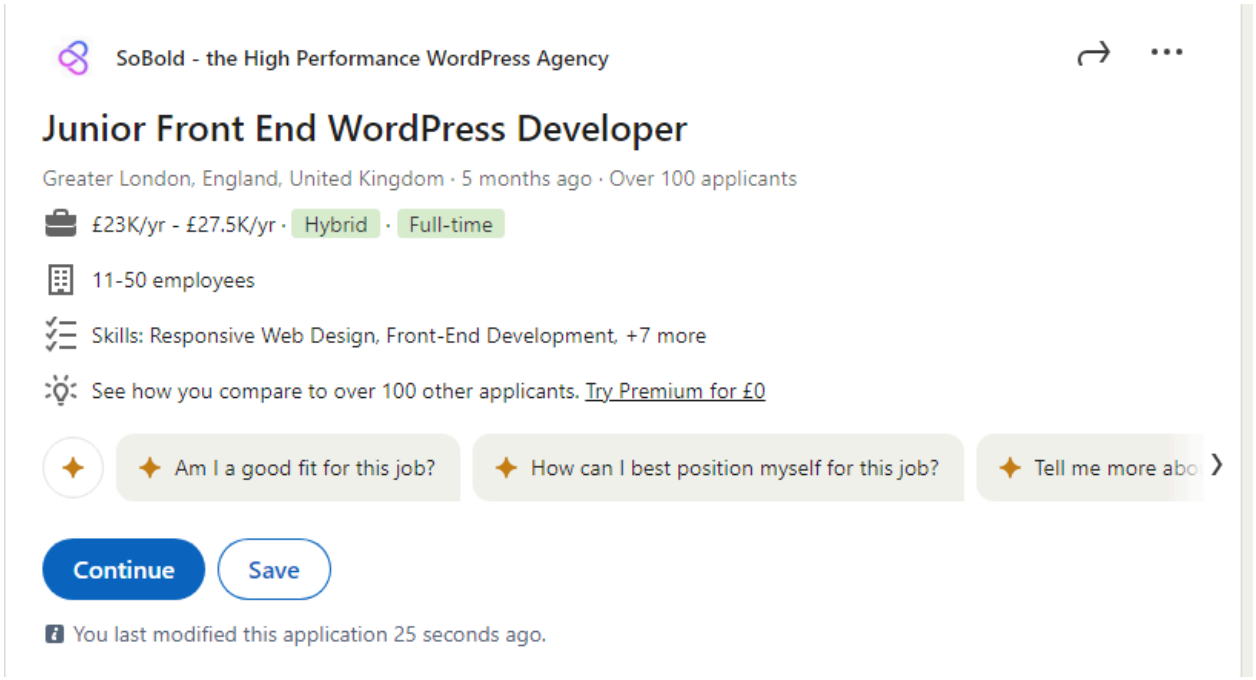
[Helder Paixao | LinkedIn]

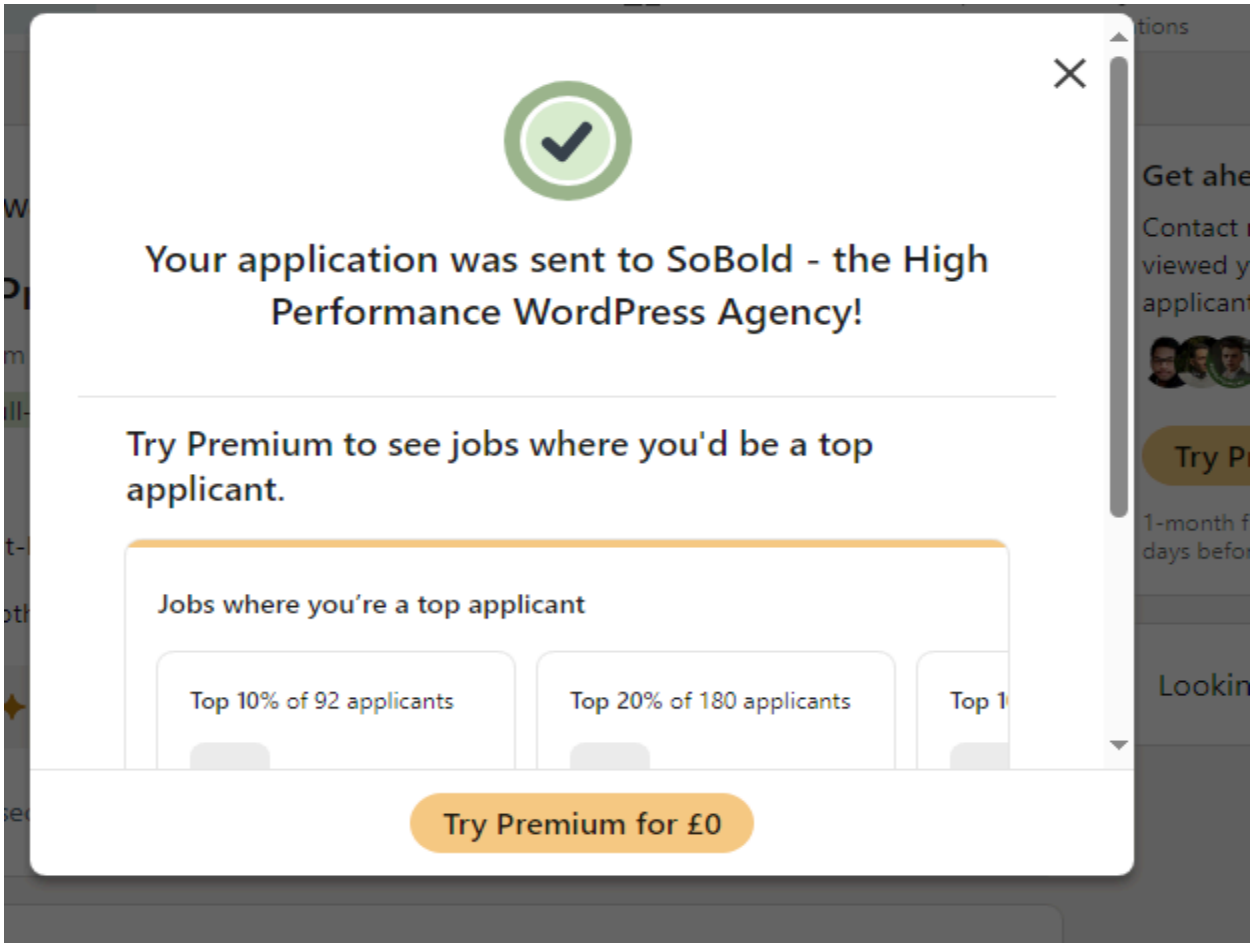
Five junior tech roles currently being advertised:



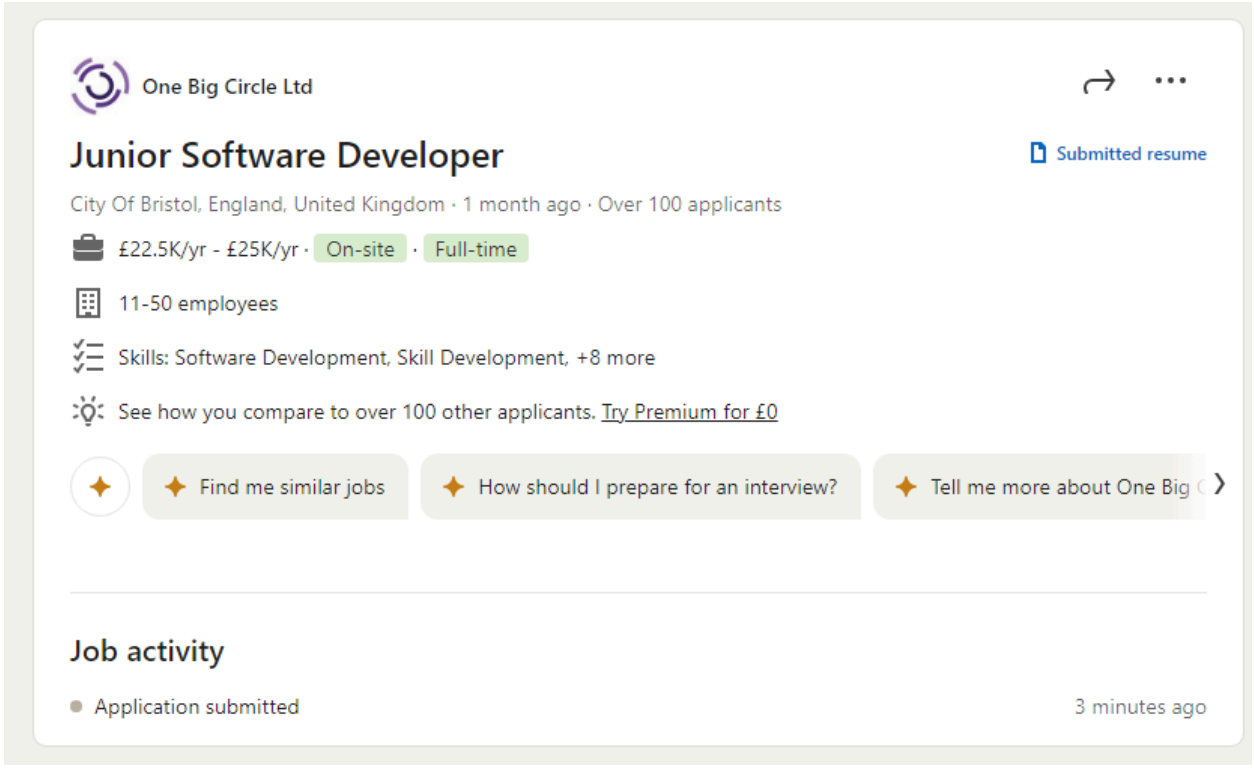
https://survey.typeform.com/to/zYNcbJGx

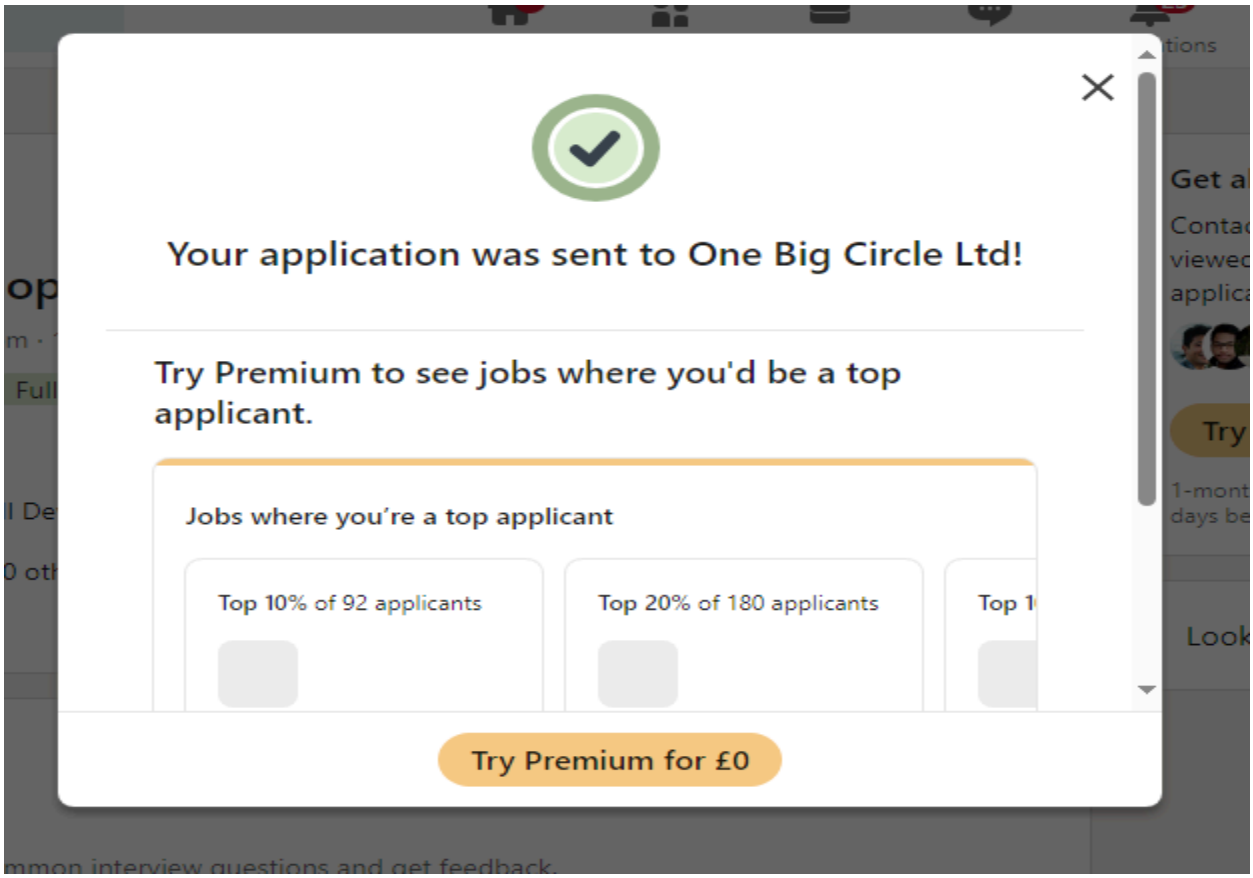
1)


SoBold - the High Performance WordPress Agency
Junior Front End WordPress Developer
Greater London, England, United Kingdom · 5 months ago · Over 100 applicants
£23K/yr - £27.5K/yr · Hybrid · Full-time
11-50 employees
Skills: Responsive Web Design, Front-End Development, +7 more
See how you compare to over 100 other applicants. Try Premium for £0
Am I a good fit for this job?
How can I best position myself for this job?
Tell me more abo
Continue
Save
You last modified this application 25 seconds ago.

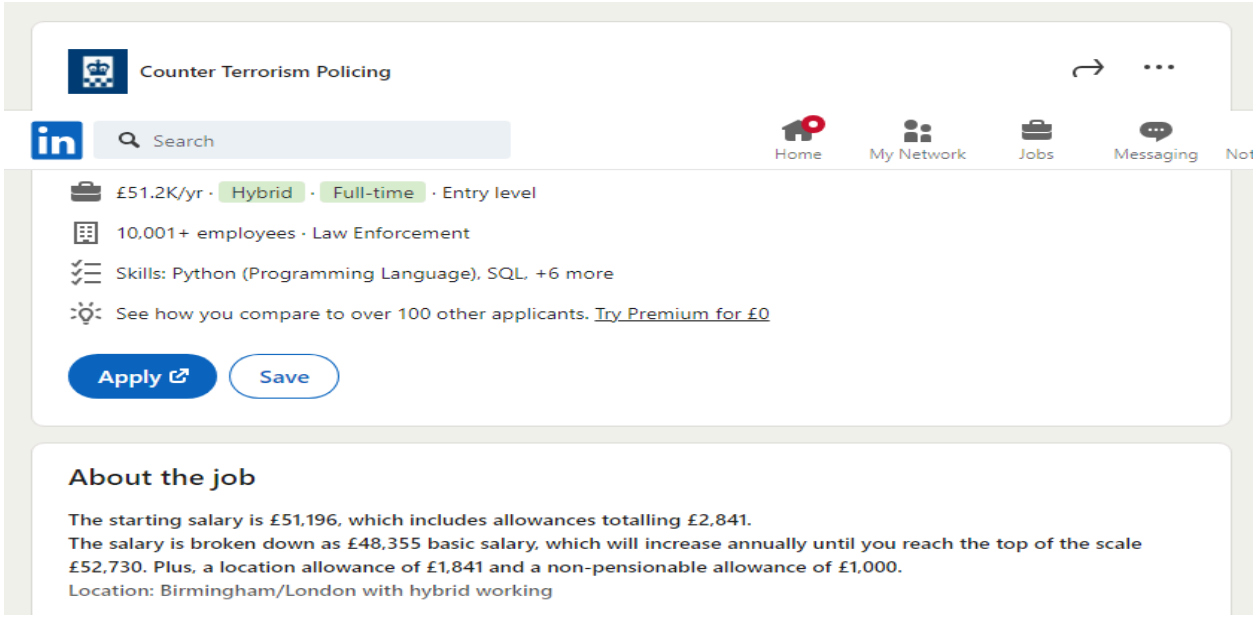

Your application was sent to SoBold - the High Performance WordPress Agency!
Try Premium to see jobs where you'd be a top applicant.
Jobs where you're a top applicant
Top 10% of 92 applicants
Top 20% of 180 applicants
Try Premium for £0

2)


One Big Circle Ltd
Junior Software Developer
Submitted resume
City Of Bristol, England, United Kingdom · 1 month ago · Over 100 applicants
£22.5K/yr - £25K/yr · On-site · Full-time
11-50 employees
Skills: Software Development, Skill Development, +8 more
See how you compare to over 100 other applicants. Try Premium for £0
Find me similar jobs
How should I prepare for an interview?
Tell me more about One Big
Job activity
Application submitted
3 minutes ago


Your application was sent to One Big Circle Ltd!
Try Premium to see jobs where you'd be a top applicant.
Jobs where you're a top applicant
Top 10% of 92 applicants
Top 20% of 180 applicants
Try Premium for £0

3)

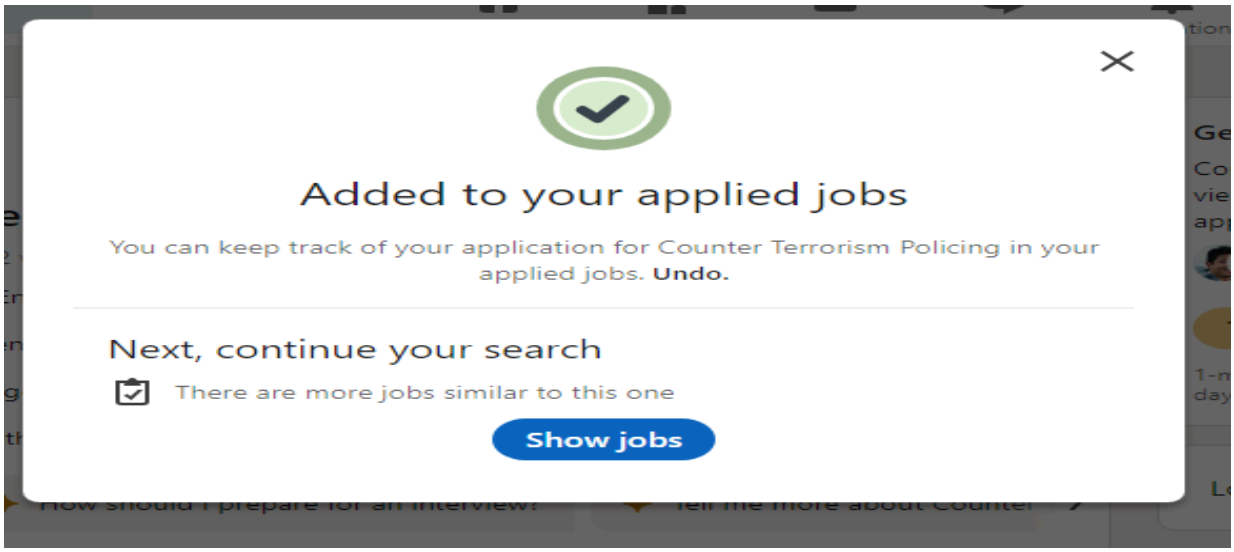


Return to search results

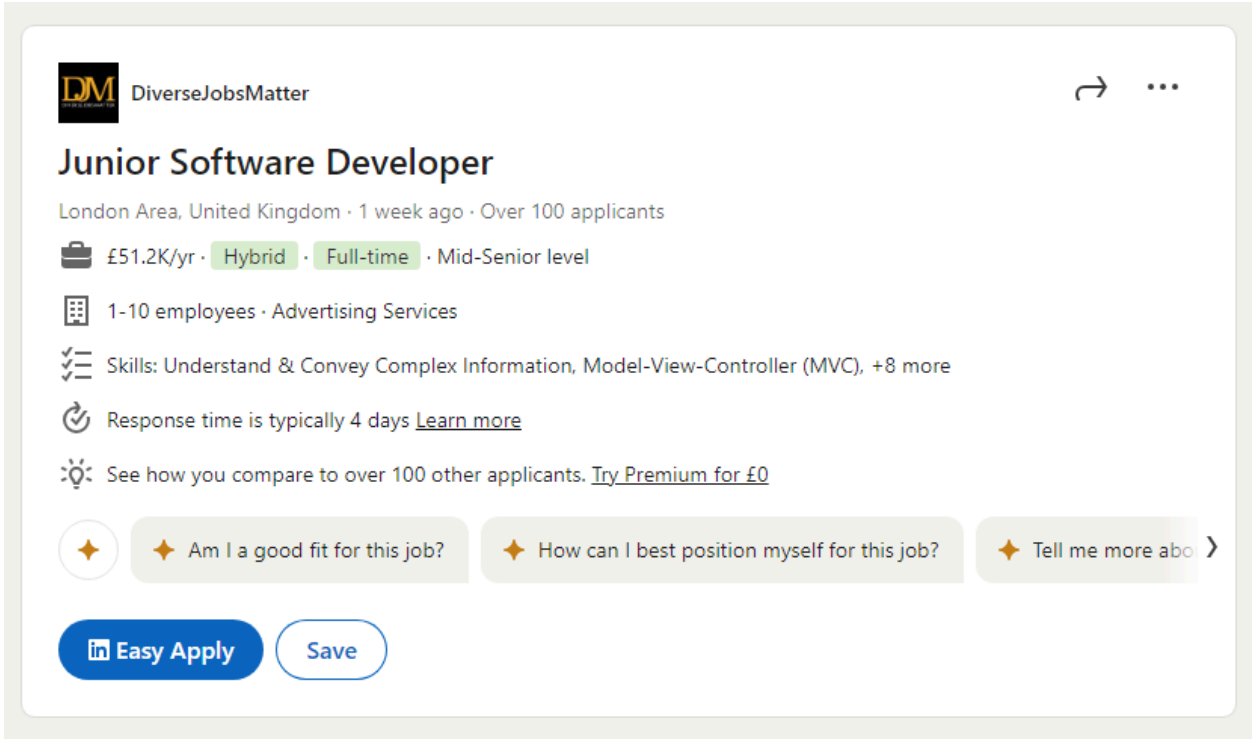
Junior Software Developer - Counter Terrorism Policing HQ

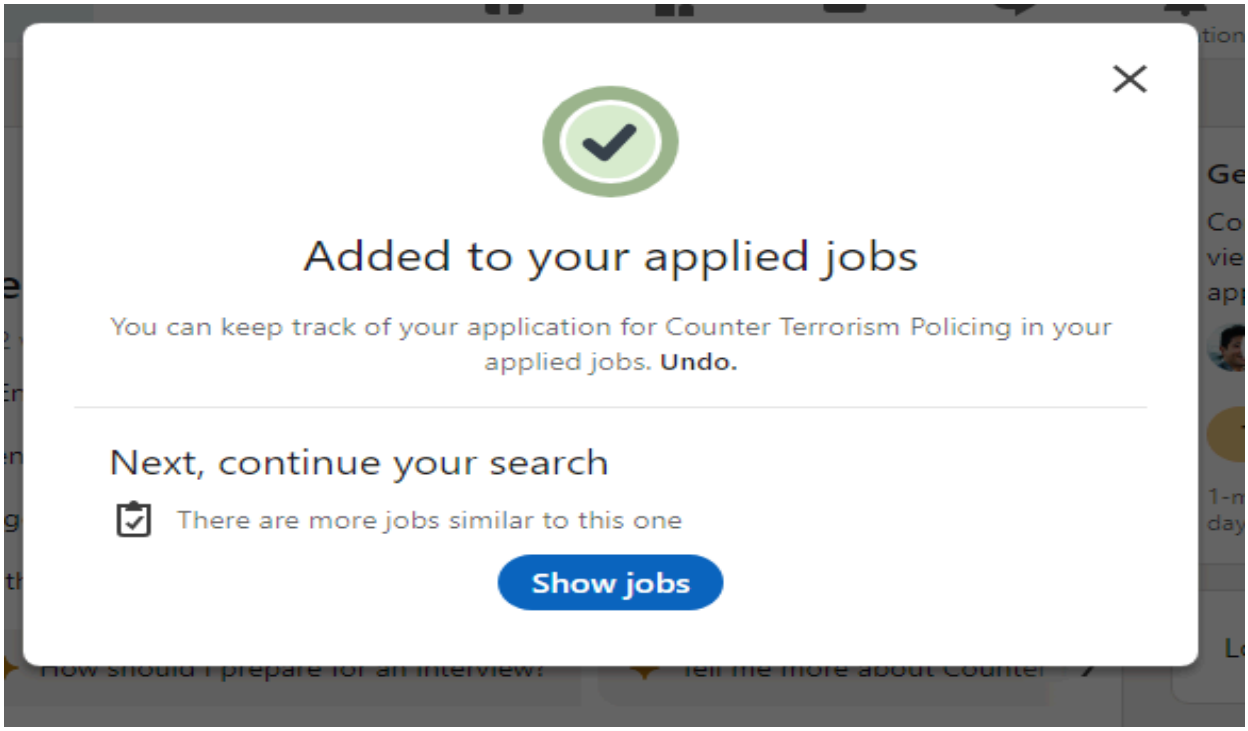
Apply

Vacancy Reference Number	17304
Number of Vacancies	4
(B) OCU	CTPHQ Counter Terrorism Policing Headquarters

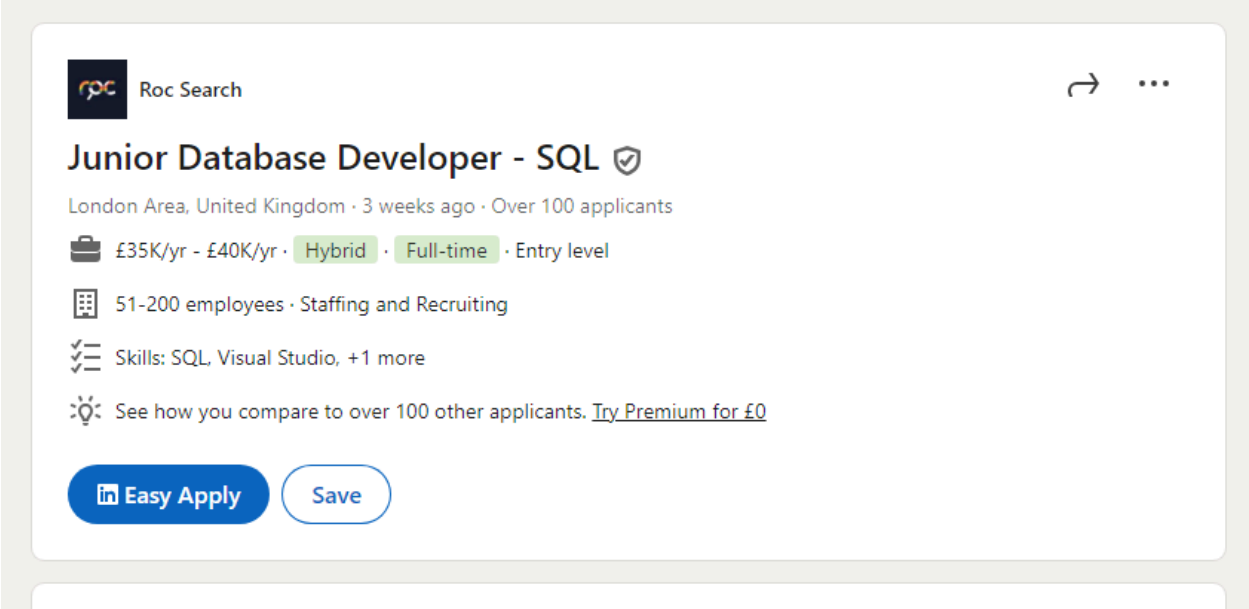


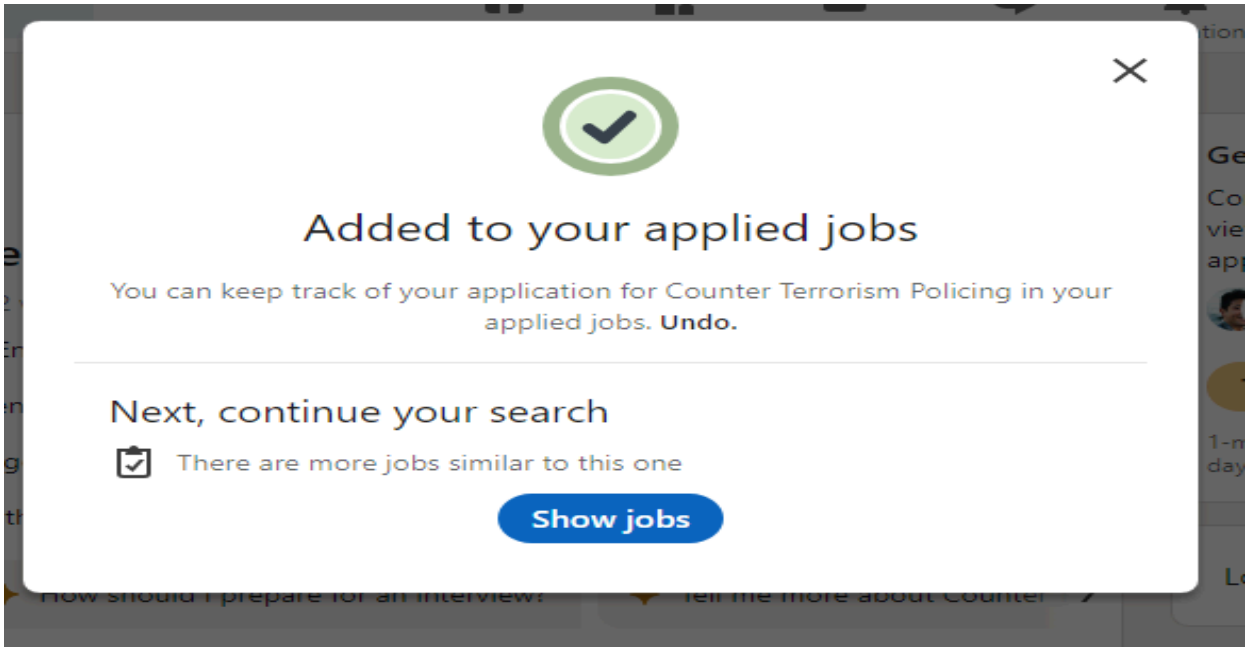
4)

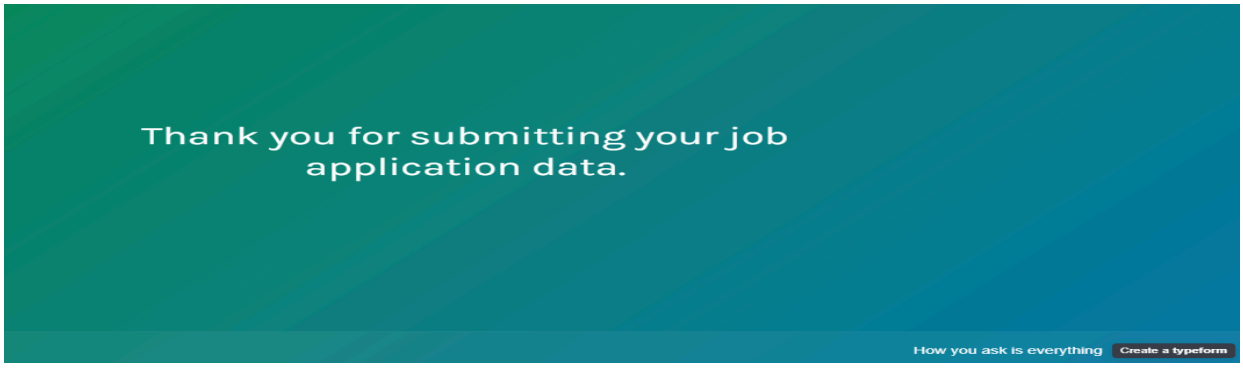

DiverseJobsMatter
Junior Software Developer
London Area, United Kingdom · 1 week ago · Over 100 applicants
£51.2K/yr · Hybrid · Full-time · Mid-Senior level
1-10 employees · Advertising Services
Skills: Understand & Convey Complex Information, Model-View-Controller (MVC), +8 more
Response time is typically 4 days Learn more
See how you compare to over 100 other applicants. Try Premium for £0
Am I a good fit for this job?
How can I best position myself for this job?
Tell me more abo
Easy Apply
Save


Added to your applied jobs
You can keep track of your application for Counter Terrorism Policing in your applied jobs. Undo.
Next, continue your search
There are more jobs similar to this one
Show jobs

5)


Roc Search
Junior Database Developer - SQL
London Area, United Kingdom · 3 weeks ago · Over 100 applicants
£35K/yr - £40K/yr · Hybrid · Full-time · Entry level
51-200 employees · Staffing and Recruiting
Skills: SQL, Visual Studio, +1 more
See how you compare to over 100 other applicants. Try Premium for £0
Easy Apply
Save


Added to your applied jobs
You can keep track of your application for Counter Terrorism Policing in your applied jobs. Undo.
Next, continue your search
There are more jobs similar to this one
Show jobs


Thank you for submitting your job application data.
How you ask is everything
Create a typeform

Practical Task 2

Record yourself answering these mock interview/pitch questions:
● Stating whether you are a job seeker or a self-employed student. Those who wish to remain at the same company but achieve an internal move to/promotion into a tech role should follow the instructions for job seekers.
● Tell me about yourself/your company/your product or service (elevator pitch).
● Why are you interested in [role/business opportunity]? (Tailor this response to the position/opportunity you've selected.)
● Please share some of your professional strengths/company areas of specialisation that you feel would be particularly beneficial for [role/business opportunity]? Also,
○ Job seekers: Discuss an area where you're looking to improve or a weakness you've been working on.
○ Self-employed people: Discuss an area of growth for yourself or your business where you are still trying to break into the market.
● Describe a technical project you worked on and how you overcame the challenges you encountered while completing it.
● Choose a technical concept and explain it to the imagined interviewer/potential client assuming they are not a technical person. (For example, explain algorithms in Software Engineering, responsive design in Web Development, or a statistical method in Data Science, depending on your field.)

Algorithms are fundamental to software engineering. They are step-by-step procedures or formulas for solving problems. Here's a detailed explanation of their role and importance in software engineering:

Definition
An algorithm is a set of well-defined instructions in sequence to solve a problem or perform a task. They are essential for developing software solutions that are efficient, effective, and scalable.

Key Characteristics of Algorithms
1. **Input**: Algorithms may have zero or more inputs.
2. **Output**: They must produce at least one output.
3. **Definiteness**: Each step must be clear and unambiguous.
4. **Finiteness**: Algorithms should terminate after a finite number of steps.
5. **Effectiveness**: The steps must be simple enough to be performed, in principle, by a human using a pen and paper.

Types of Algorithms
1. **Sorting Algorithms**: Arrange elements in a particular order. Examples include QuickSort, MergeSort, and Bubble Sort.
2. **Search Algorithms**: Retrieve information stored within data structures. Examples include Binary Search and Linear Search.
3. **Graph Algorithms**: Used for solving graph problems like shortest path, connectivity, and spanning tree. Examples include Dijkstra's Algorithm and Kruskal's Algorithm.
4. **Dynamic Programming**: Solves problems by breaking them down into simpler subproblems. Examples include the Fibonacci sequence and the Knapsack problem.
5. **Greedy Algorithms**: Make the locally optimal choice at each stage with the hope of finding a global optimum. Examples include Prim's and Kruskal's algorithms for finding minimum spanning trees.
6. **Divide and Conquer**: Divide the problem into smaller subproblems, solve them independently, and then combine the results. Examples include MergeSort and QuickSort.

Importance in Software Engineering
1. **Efficiency**: Algorithms help in creating efficient solutions that save time and resources. Efficient algorithms can handle larger datasets and perform faster.
2. **Scalability**: Well-designed algorithms ensure that software can handle growing amounts of work or its capability to be enlarged.
3. **Problem-Solving**: Algorithms provide systematic approaches to problem-solving, which can be applied across various domains and applications.
4. **Optimization**: Many algorithms are designed to find the best solution among many, ensuring optimal performance in software applications.
5. **Foundation of Software Development**: Understanding algorithms is fundamental to learning advanced programming techniques and software development practices.

Examples in Real-World Applications
1. **Search Engines**: Use complex algorithms to search and rank web pages.
2. **Cryptography**: Relies on algorithms for securing communication.
3. **Data Analysis**: Algorithms process and analyze large datasets to extract meaningful insights.
4. **Artificial Intelligence**: Machine learning algorithms are at the core of AI applications.
5. **Operating Systems**: Scheduling algorithms determine the order in which processes run.

Developing Algorithms
When developing algorithms, software engineers follow these steps:
1. **Understand the Problem**: Clearly define the problem you are trying to solve.
2. **Plan the Algorithm**: Write a step-by-step procedure to solve the problem.
3. **Write Pseudocode**: Represent the algorithm in pseudocode to simplify the transition to actual code.
4. **Implement the Algorithm**: Translate the pseudocode into a programming language.
5. **Test the Algorithm**: Verify the algorithm with different inputs to ensure it works correctly.

6. **Analyze the Algorithm**: Evaluate its efficiency in terms of time and space complexity (Big O notation).

Algorithm Complexity
Algorithm complexity is analyzed to determine the efficiency of an algorithm. The two primary aspects are:
1. **Time Complexity**: Amount of time an algorithm takes to complete as a function of the length of the input.
2. **Space Complexity**: Amount of memory an algorithm uses in relation to the input size.

Understanding and mastering algorithms is crucial for any software engineer, as they are the backbone of creating effective and efficient software solutions.

Practical Task 3

Publicly accessible link, to my GitHub repository containing the code I have uploaded to my profile repository.

⇒ HelderGavazi/HelderGavazi (github.com)

Final Submission Checklist Ensure you hand in the following: ● In your Google answers doc: ○ Screenshots of your completed form and a screenshot showing proof that you have successfully submitted the form. ○ A publicly accessible link to your copy of the Job Interviews/Pitch Opportunities Tracker, updated to include the five new roles you applied for. This short video will guide you through creating a shareable link. ○ A link to your video from Practical Task 2 if you're hosting it online rather than uploading it. ○ A publicly accessible link, pasted into your Google answers doc, to your GitHub repository containing the code you uploaded to your developer profile repository. Once you've done the above, save your Google answers doc as a PDF (using menu options File -> Download -> PDF) and upload it to your Dropbox folder for this Task. ● If you haven't provided a link to an online version of your interview/pitch video in your Google doc, please upload a copy of the video to your Dropbox for this task.