

Program Code:

🔗 CSC_180_LAB_07.py > ...

```
1  from timeit import default_timer as timer
2
3  # Store interview data
4  interview_data = []
5
6  # Set thresholds - my criteria based on how fast I think they should answer
7  THRESHOLDS = {
8      "name": 30,          # >30s: Likely stepped away
9      "job": 30,          # >30s: Hesitation (embarrassed or lying)
10     "years": 5,         # >30s: Possibly lying
11     "reference": 45,    # >45s: Possibly lying
12     "skills": 150,      # >150s: Thinking too hard, probably exaggerating
13     "qualities": 150,   # >150s: Not very unique
14     "weaknesses": 150,  # >150s: Fabricating weaknesses
15     "achievement": 150, # >150s: Fabricating or exaggerating
16     "why": 120,         # >120s: Not sure if they want the job
17     "what": 150         # >150s: Nothing separates them
18 }
19
20 # Function to analyze response time and generate assumptions
21 def analyze_response(questionList, response_time):
22     if questionList in THRESHOLDS:
23         threshold = THRESHOLDS[questionList]
24         if response_time >= threshold:
25             # Assumptions based on question
26             assumptions = {
27                 "name": "Likely stepped away.",
28                 "job": "Possible hesitation—either embarrassed or unsure about the job.",
29                 "years": "Possible dishonesty about work experience.",
30                 "reference": "Might be fabricating how they heard about the job.",
31                 "skills": "Thinking too hard—possibly exaggerating skills.",
32                 "qualities": "Took too long—might not be very unique.",
33                 "weaknesses": "Possible fabrication of weaknesses.",
34                 "achievement": "Possible exaggeration of accomplishment.",
35                 "why": "Uncertain about wanting the job.",
36                 "what": "Nothing truly separates them from others."
37             }
38             return assumptions.get(questionList, None) # Returns assumption, if applicable
39     return None # Returns nothing if no assumption should be made
40
41 # Function to ask a question, track response time, and analyze individual response
42 def askQuestion(questionList, question_text, whatToPrint):
43     start_time = timer()
44     print(question_text)
45     user_response = input()
46     end_time = timer()
47
48     response_time = end_time - start_time
49     assumption = analyze_response(questionList, response_time)
50
```

```
51 # Store the response in the interview_data list
52 interview_data.append({
53     "question": question_text,
54     "response": user_response,
55     "response_time": response_time,
56     "assumption": assumption
57 })
58
59 print(f"whatToPrint: {user_response}")
60 print(f"Response Time: {response_time:.0f} seconds")
61
62 # Print analysis only if response time exceeds the threshold
63 if assumption:
64     print(f"Analysis: {assumption}\n")
65
66 # List of interview questions with their criteria
67 questions = [
68     {"key": "name", "text": "Please enter your name:", "print": "Welcome"},
69     {"key": "job", "text": "Please enter your current occupation:", "print": "Current Job"},
70     {"key": "years", "text": "How many years have you been at that job?", "print": "Years"},
71     {"key": "reference", "text": "How did you hear about this job?", "print": "How interviewee heard of job"},
72     {"key": "skills", "text": "What skills do you have?", "print": "Skills:"},
73     {"key": "qualities", "text": "List some qualities about yourself?", "print": "Qualities of interviewee"},
74     {"key": "weaknesses", "text": "What are your weaknesses?", "print": "Interviewee's Weaknesses"},
75     {"key": "achievement", "text": "What is your greatest accomplishment?", "print": "Greatest Accomplishment"},
76     {"key": "why", "text": "Why do you want this job?", "print": "Why interviewee wants the job"},
77     {"key": "what", "text": "What separates you? Why should I hire you?", "print": "Why the interviewee feels they should be hired"}
78 ]
79
80 # Ask all questions
81 for q in questions:
82     askQuestion(q["key"], q["text"], q["print"])
83
84 # Function to calculate overall statistics
85 def analyze_overall_data():
86     response_times = [entry["response_time"] for entry in interview_data]
87     if response_times:
88         avg_time = sum(response_times) / len(response_times)
89         max_time = max(response_times)
90         min_time = min(response_times)
91         print("\n### Overall Analysis ###")
92         print(f"Average Response Time: {avg_time:.2f} seconds")
93         print(f"Fastest Response Time: {min_time:.2f} seconds")
94         print(f"Slowest Response Time: {max_time:.2f} seconds")
95
96     # Print only assumptions that exist (skip responses below threshold)
97     assumptions_to_print = [entry for entry in interview_data if entry["assumption"]]
98
99     if assumptions_to_print:
100         print("\n### Individual Assumptions ###")
101         for entry in assumptions_to_print:
102             print(f"Question: {entry['question']}")
103             print(f"Response Time: {entry['response_time']:.2f} seconds")
104             print(f"Assumption: {entry['assumption']}\n")
```

```
104         print(f"Assumption: {entry['assumption']}\n")
105
106 # Run overall analysis after interview
107 analyze_overall_data()
108
```

Program Output Examples (some of the thresholds were changed just to save some time in displaying output):

```
Please enter your name:
mario
Welcome: mario
Response Time: 15 seconds
Please enter your current occupation:
software engineer
Current Job: software engineer
Response Time: 7 seconds
How many years have you been at that job?
6
Years: 6
Response Time: 18 seconds
Analysis: Possible dishonesty about work experience.

How did you hear about this job?
friend
How interviewee heard of job: friend
Response Time: 8 seconds
What skills do you have?
6
Skills:: 6
Response Time: 447 seconds
Analysis: Thinking too hard—possibly exaggerating skills.
```

```
List some qualities about yourself?
nice guy
Qualities of interviewee: nice guy
Response Time: 26 seconds
What are your weaknesses?
none
Interviewee's Weaknesses: none
Response Time: 2 seconds
What is your greatest accomplishment?
super bowl
Greatest Accomplishment: super bowl
Response Time: 4 seconds
Why do you want this job?
i want the money
Why interviewee wants the job: i want the money
Response Time: 6 seconds
What separates you? Why should I hire you?
im always prepared
Why the interviewee feels they should be hired: im always prepared
Response Time: 7 seconds
```

Overall Analysis

```
Average Response Time: 53.91 seconds
Fastest Response Time: 2.10 seconds
Slowest Response Time: 446.69 seconds
```

Individual Assumptions

```
Question: How many years have you been at that job?
Response Time: 18.16 seconds
Assumption: Possible dishonesty about work experience.
```

```
Question: What skills do you have?
Response Time: 446.69 seconds
Assumption: Thinking too hard—possibly exaggerating skills.
```

Artificial Intelligence Reflection:

In this project, I was able to take information from the user during an interview, save it to lists that python could go through and analyze in real time, and after the interview, and make predictions based on thresholds and criteria that I created based on how long the user took to respond to each question. This gives the employer (me in this case) some information, based on data, to know whether or not the potential employee was a professional candidate. We gave the program data (our criteria) to analyze the user input and thus make predictions.