

Hadoop Administrator

Candidate-ID	0042400088
Candidate Name	Raviteja Ramakurthi
Email	ramakurthiraviteja@gmail.com
Mobile	9849767288

Screening Test

Completed on : 11-Dec-2024 02:32 PM

#	Question	Response	Marks
1	Do you have a bachelors degree in computer science or in the related field	Yes	1
2	Do you have working experience in Hadoop on Linux for (1 to 3 years)	Yes	1
3	Did you you ever worked with multiple teams and used various collaboration tools and methods	Yes	1
4	Do you have Hadoop Administration skills	Yes	1
5	Do you have Linux Administration & troubleshooting skills	Yes	1
6	Do you have Shell Scripting skills	Yes	1
7	Do you think you can Handle and process large datasets using Hadoop ecosystem tools such as HDFS, Hive, and Hbase.	Yes	1
8	Do you have efficient coding skills in Python programming	Yes	1
9	Are you ready to travel outside of India on the project work	Yes	1
10	Do you have a valid Passport	Yes	1

Overall Score : 10/10 (Promoted)

Coding Test

Completed on : 11-Dec-2024 03:32 PM

1. Kattapa, as you all know was one of the greatest warriors of his time. The kingdom of Maahishmati had never lost a battle under him (as army-chief), and the reason for that was their really powerful army, also called as Mahasena. Kattapa was known to be a very superstitious person. He believed that a soldier is "lucky" if the soldier is holding an even number of weapons, and "unlucky" otherwise. He considered the army as "READY FOR BATTLE" if the count of "lucky" soldiers is strictly greater than the count of "unlucky" soldiers, and "NOT READY" otherwise. Given the number of weapons each soldier is holding, your task is to determine whether the army formed by all these soldiers is "READY FOR BATTLE" or "NOT READY".

Marks Scored : 2

```
class Army:
    def __init__(self, weapons):
        self.weapons = weapons

    def army_readiness(self):
        even_count=sum(1 for weapon in self.weapons if weapon %2==0)
        odd_count=len(self.weapons)-even_count
        if even_count > odd_count:
            return "READY FOR BATTLE"
        else:
            return "NOT READY"    # Write code here

army1 = Army([1]) #input
print(army1.army_readiness()) # Output: NOT READY

army2 = Army([2])#input
print(army2.army_readiness()) # Output: READY FOR BATTLE

army3 = Army([11, 12, 13, 14])#input
print(army3.army_readiness()) # Output: NOT READY

army4 = Army([2, 3, 4])#input
print(army4.army_readiness()) # Output: READY FOR BATTLE
```

2. A space explorer's ship crashed on Mars! They send a series of SOS messages to Earth for help. Letters in some of the SOS messages are altered by cosmic radiation during transmission. Given the signal received by Earth as a string, determine how many letters of the SOS message have been changed by radiation. Example The original message was SOSSOS. Two of the message's characters were changed in transit. Function Description Complete the marsExploration function in the editor below. marsExploration has the following parameter(s): • string s: the string as received on Earth Returns • int: the number of letters changed during transmission Input Format There is one line of input: a single string, .

Marks Scored : 4

```
class SignalTransmission:
    def countAlteredCharacters(self, s):
```

```
# Write your code here
original_sos="SOS"
altered_count=0
for i in range(0,len(s),3):
    chunk=s[i:i+3]
    for j in range(len(chunk)):
        if chunk[j]!=original_sos[j]:
            altered_count+=1
    return altered_count
signal=SignalTransmission()
received_message="SOTSOXSOSSOT"
print(signal.countAlteredCharacters(received_message))
```

Overall Score : 6/6 (Promoted)