

Python Developer

Candidate-ID	0012500073
Candidate Name	Manikanta Meka
Email	mvsmanikantameka@gmail.com
Mobile	9704309632
Source	Careers

Screening Test

Completed on : 11-Nov-2025 05:39 PM

#	Question	Response	Marks
1	Do you have professional experience between 0 to 2 years?	Yes	1
2	Would you be open to relocating to Hyderabad?	Yes	1
3	Do you have experience in performance optimization and debugging?	Yes	1
4	Explain about yourself why you are the right candidate for this Job	video response	1
5	Are you, willing to work on-premises (there is no work from home)	Yes	1
6	Are you, available immediately to join us, if we hire you	Yes	1
7	if we hire you, will you be prepared to stay and work with us at least for two years	Yes	1
8	Are you open to traveling for work when necessary, either domestically or internationally?	Yes	1
9	Are you familiar with Agile methodologies, including Scrum or Kanban?	Yes	1
10	Have you recently completed your education and training, and there no is major Gap (more then one year after the education)	Yes	1

Overall Score : 10/10 (Promoted)

Coding Test

Completed on : 11-Nov-2025 06:23 PM

1. Write a program to return words to any given number ?(Ex. for input 150 output has to be One hundred fifty only)
Marks Scored : 5

```
class NumberToWordsConverter:
    def __init__(self):
        # Define words for different ranges
        self.units = ["", "One", "Two", "Three", "Four", "Five", "Six", "Seven", "Eight", "Nine"]
        self.teens = ["Ten", "Eleven", "Twelve", "Thirteen", "Fourteen", "Fifteen", "Sixteen", "Seventeen", "Eighteen", "Nineteen"]
        self.tens = ["", "", "Twenty", "Thirty", "Forty", "Fifty", "Sixty", "Seventy", "Eighty", "Ninety"]

    def convert_to_words(self, number):
        if number == 0:
            return "Zero"
        elif number < 10:
            return self.units[number]
        elif 10 <= number < 20:
            return self.teens[number - 10]
        elif 20 <= number < 100:
            return self.tens[number // 10] + (" " if number % 10 == 0 else " " + self.units[number % 10])
        elif 100 <= number < 1000:
            hundreds = self.units[number // 100] + " Hundred"
            remainder = number % 100
            if remainder == 0:
                return hundreds
            else:
                return hundreds + " " + self.convert_to_words(remainder)
```

2. Exercise : An incomplete shopping cart program is provided for you to complete it, expected implementation is adding items, removing items and calculating the total value. Methods to be Implanted. 1. add_item(self, item_name, qty, price) function. 2. remove_item(self, item_name). 3. calculate_total(self).
Marks Scored : 5

```
class ShoppingCart:
    def __init__(self):
        # Initialize an empty dictionary to store items, quantities, and prices
        self.cart = {}

    def add_item(self, item_name, qty, price):
        # Add or update item in the cart with quantity and price
        if item_name in self.cart:
            self.cart[item_name]['quantity'] += qty
            self.cart[item_name]['price'] = price # Update price in case it changes
        else:
            self.cart[item_name] = {'quantity': qty, 'price': price}
        print(f"Added {qty} of {item_name} at ${price} each to the cart.")
```

```
def remove_item(self, item_name):
    # Remove item from cart if it exists
    if item_name in self.cart:
        del self.cart[item_name]
        print(f'Removed {item_name} from the cart.')
    else:
        print(f'{item_name} not found in the cart.')

def calculate_total(self):
    # Calculate and return the total cost of items in the cart
    total = sum(item['quantity'] * item['price'] for item in self.cart.values())
    print(f'Total cost: ${total:.2f}')
    return total

# Example usage:
cart = ShoppingCart()

cart.calculate_total()
```

3. Write a program to identify and return the smallest and largest number from given series of numbers, Note: Don't to use library functions min and max and sort and sorted in the code, it is expected that you write your own algorithm to identify and return the smallest and largest number, Please return the values in format (smallest,largest), Given array : [56,4,48,2,75,35,45,63,500,8,20,452,324] Example : (2 , 500)
Marks Scored : 5

```
class NumberAnalyzer:
    def __init__(self, numbers):
        self.numbers = numbers

    def find_smallest_and_largest(self):
        smallest = self.numbers[0]
        largest = self.numbers[0]
        for num in self.numbers:
            if num < smallest:
                smallest = num
            if num > largest:
                largest = num

        return (smallest, largest)

# Given array
array = [56, 4, 48, 2, 75, 35, 45, 63, 500, 8, 20, 452, 324]

# Create an instance of NumberAnalyzer
analyzer = NumberAnalyzer(array)

# Find smallest and largest
result = analyzer.find_smallest_and_largest()
print(result)
```

Overall Score : 15/15 (Promoted)

Interview

Interviewed on : 11-Nov-2025 08:30 PM

Interviewd by : Srinivas Gadepalli

Interview Summary

Code Optimization	1.0
Virtual Environments	2.5
Dictionaries	3.0
Lists	4.0
Recursion	3.0
Mysql Connector	5.0
Average interview rating	3.08

Question wise rating:

Code Optimization

What is code optimization in best practices?	1
Average rating	1.0

Virtual Environments

Can you explain how virtual environments works in Python?	2
How does virtual environments relate to other core Python concepts?	3
Average rating	2.5

Dictionaries

What is dictionaries in data structures?	3
Average rating	3.0

Lists

What is lists in data structures?	4
Average rating	4.0

Recursion

What is recursion in functions?	4
Can you explain how recursion works in Python?	2
Average rating	3.0

Mysql Connector

What is mysql connector in database?	5
Average rating	5.0

Interviewer Remarks:

No Remarks

Interviewer's Feedback:

Interviewer	Decision	Notes
Srinivas Gadepalli	Hire	High Potential Good hire