

```
# Function to merge two halves
```

```
def merge(left, right):
```

```
    sorted_list = []
```

```
    i = j = 0
```

```
    # Compare elements from both halves and merge them in sorted order
```

```
    while i < len(left) and j < len(right):
```

```
        if left[i]['delivery_time'] <= right[j]['delivery_time']:
```

```
            sorted_list.append(left[i])
```

```
            i += 1
```

```
        else:
```

```
            sorted_list.append(right[j])
```

```
            j += 1
```

```
    # Append remaining elements from both halves
```

```
    sorted_list.extend(left[i:])
```

```
    sorted_list.extend(right[j:])
```

```
    return sorted_list
```

```
# Merge Sort function
```

```
def merge_sort(orders):
```

```
    if len(orders) <= 1:
```

```
        return orders
```

```
    # Divide the list into two halves
```

```
    mid = len(orders) // 2
```

```
    left_half = merge_sort(orders[:mid])
```

```
    right_half = merge_sort(orders[mid:])
```

```
# Merge the sorted halves
```

```
return merge(left_half, right_half)
```

```
# Example usage
```

```
if __name__ == "__main__":
```

```
    # List of orders with delivery times
```

```
    orders = [
```

```
        {"order_id": 1, "delivery_time": 45},
```

```
        {"order_id": 2, "delivery_time": 30},
```

```
        {"order_id": 3, "delivery_time": 60},
```

```
        {"order_id": 4, "delivery_time": 15},
```

```
        {"order_id": 5, "delivery_time": 20}
```

```
    ]
```

```
    # Sort orders by delivery time
```

```
    sorted_orders = merge_sort(orders)
```

```
    # Print sorted orders
```

```
    print("Sorted Orders by Delivery Time:")
```

```
    for order in sorted_orders:
```

```
        print(order)
```