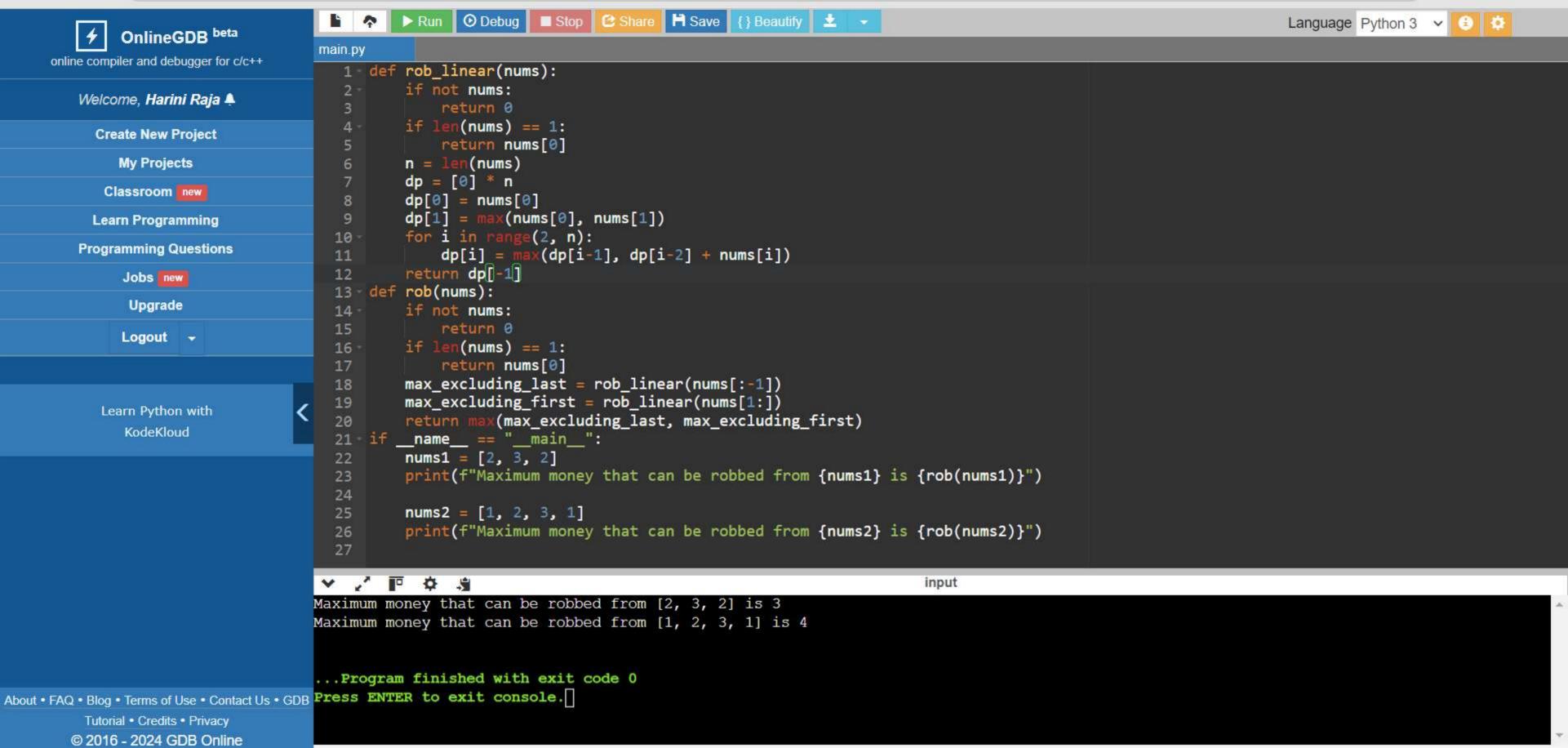


© 2016 - 2024 GDB Online





Classroom new

Learn Programming **Programming Questions**

Jobs new

Upgrade

Logout -

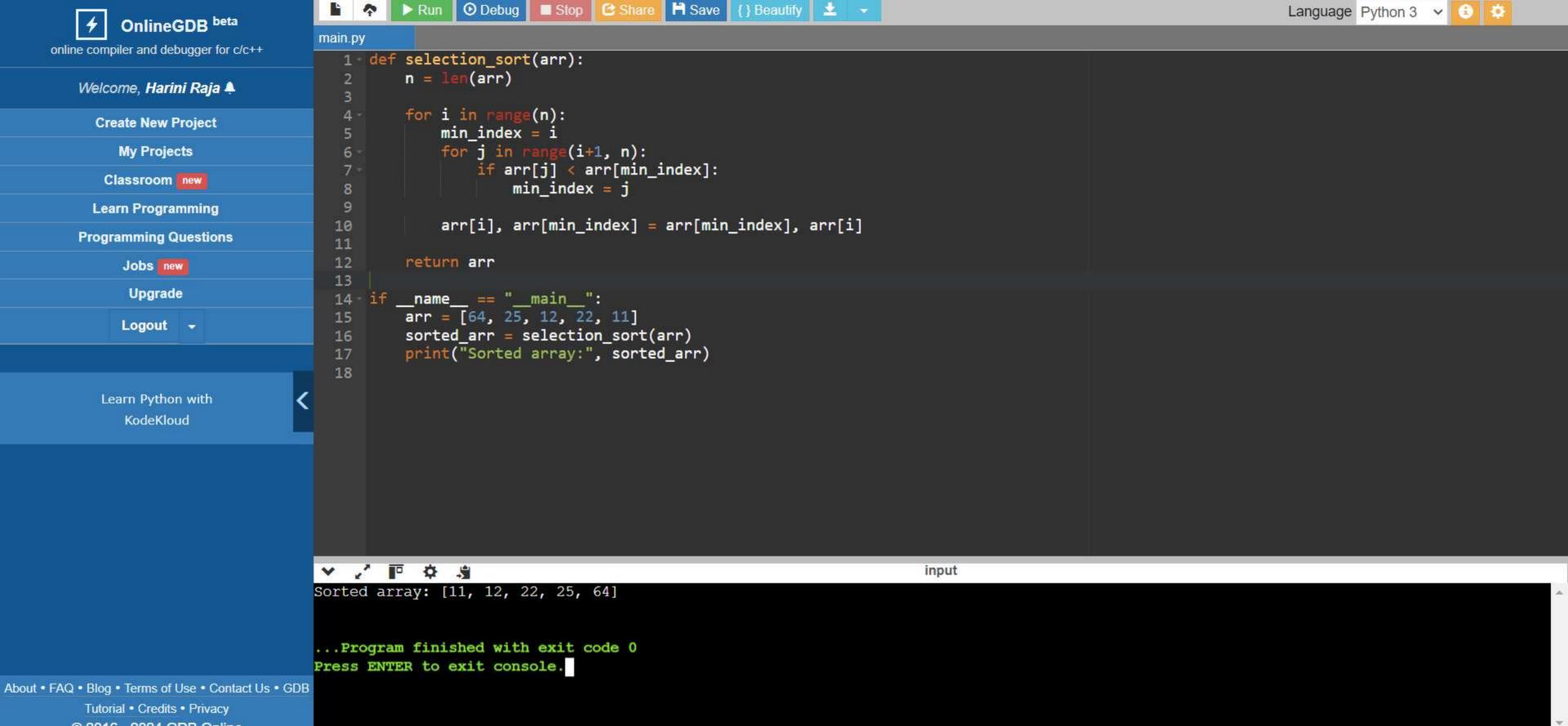
Learn Python with KodeKloud

```
main.py
  1 import heapq
     def dijkstra(graph, source):
          n = len(graph)
          distances = [float('inf')] * n
          distances[source] = 0
          priority_queue = [(0, source)]
         while priority queue:
              current distance, current vertex = heapq.heappop(priority queue)
  8
              if current distance > distances[current vertex]:
  9 -
                  continue
 10
              for neighbor, weight in enumerate(graph[current_vertex]):
 11
                  if weight != float('inf'):
 12
                       distance = current distance + weight
 13
                      if distance < distances[neighbor]:</pre>
 14
                           distances[neighbor] = distance
 15
                                   eappush(priority queue, (distance, neighbor))
 16
          return distances
 17
     graph = [
          [0, 10, 3, float('inf'), float('inf')],
 19
                t('inf'), 0, 1, 2, float('inf')],
 20
               t('inf'), 4, 0, 8, 2],
 21
          [float('inf'), float('inf'), float('inf'), 0, 7],
[float('inf'), float('inf'), float('inf'), 9, 0]
 22
 23
 24
 25 source = 0
 26 output = dijkstra(graph, source)
 27 print(output)
♥ ( □ ( ) ♥
                                                                        input
```

Language Python 3 V

... Program finished with exit code 0 Press ENTER to exit console.

Run O Debug Stop O Share Save { Beautify Y





Learn Programming

Programming Questions

Jobs new

Upgrade

Logout -

Learn Python with KodeKloud

```
Language Python 3 🗸 💽 😳
main.py
  1 def find_element(arr, target):
        for i, num in enumerate(arr):
            if num == target:
               return i
        return -1
     if __name__ == "__main__":
        arr = [2, 3, 4, 5, 6, 1, 0, 9]
        target = 8
  10
 11
        index = find_element(arr, target)
 12
 13
        if index != -1:
            print(f"Element {target} found at index {index}.")
 14
 15
        else:
            print(f"Element {target} not found in the array.")
 16
 17
input
Element 8 not found in the array.
```

...Program finished with exit code 0 Press ENTER to exit console.

