Consider a 2D list storing student's name along with their marks. Use list comprehension to create another list comprising names of students with marks greater than 85.

```
def comprehendlist(arr):
  return [x for x in arr if x[1] > "85"]

def startingPoint():
  rows = int(input("Enter number of rows: "))
  arr=[]
  for i in range(rows):
      name = input("Enter names : ")
      marks = input("Enter marks : ")
      arr.append([name,marks])
  print(arr)
  print(comprehendlist(arr))

if __name__ == "__main__":
    startingPoint()
```

```
Enter number of rows: 3
Enter names : bella
Enter marks : 86
Enter names : edward
Enter marks : 89
Enter names : matteo
Enter marks : 65
[['bella', '86'], ['edward', '89'], ['matteo', '65']]
[['bella', '86'], ['edward', '89']]
```

WAP that takes a list of marks as an input from the user and creates a dictionary storing marks and the corresponding frequency as key-value pairs.

```
def frequency(marksList):
   freq = {}
   for mark in marksList:
       if mark in freq:
           freq[mark] += 1
       else:
           freq[mark] = 1
   return freq


def startingPoint():
    marks = []
    num = int(input("Enter size of list : "))
    for _ in range(0, num):
        mark = int(input("Enter the marks : "))
```

```
        marks.append(mark)
    print(frequency(marks))


if __name__ == "__main__":
    startingPoint()
```

```
 Enter size of list : 6
 Enter the marks : 56
 Enter the marks : 58
 Enter the marks : 51
 Enter the marks : 53
 Enter the marks : 49
 Enter the marks : 50
 {56: 1, 58: 1, 51: 1, 53: 1, 49: 1, 50: 1}
```

WAP that takes a list of names as an input from the user and creates a dictionary storing word-length as key-value pair for each word given in the list.

```
def createDict(lst):
    res_dct = {}
    print(num)
    for i in range(0, num):
        res_dct[lst[i]] = len(lst[i])
    return res_dct


def startingPoint():
  names = []
  num = int(input("Enter total number of names: "))
  for i in range(0, num):
      name = input("Enter the name: ")
      names.append(name)
  print(names)
  print(createDict(names))

if __name__ == "__main__":
    startingPoint()
```

```
 Enter total number of names: 5
 Enter the name: swati
 Enter the name: mansi
 Enter the name: bella
 Enter the name: veronica
 Enter the name: natalia
 ['swati', 'mansi', 'bella', 'veronica', 'natalia']
 5
 {'swati': 5, 'mansi': 5, 'bella': 5, 'veronica': 8, 'natalia': 7}
```