Part 1: This code gets the number of students in the class, the highest and lowest marks and the average mark formatted to two decimal places.

```
data =
[90,30,13,67,85,87,50,45,51,72,64,69,59,17,22,23,44,25,16,67,85,87,50,45,51,72,59,14,50,55,32,23,24,25,37
,28,39,30,33,35,40,34,41,43,94,95,98,99,44,45,47,48,49,53,61,63,69,75,77,60,83]

print(f"Number of marks: {len(data)}")
print(f"Max mark: {max(data)}")
print(f"Min mark: {min(data)}")
print(f"Max mark: {round(sum(data) / len(data), 2)}")
```

Part 2: This code returns a dictionary with the mark ranges, the number of marks within that range and a asterisks counter to tally them up.

```
def get_asterisks(num):
    return('*' * num)

def get_occourances_in_range(start, fin):
    occourances = []
    for item in data:
        if item > start and item < fin:
            occourances.append(item)

    print()
    return [len(occourances), get_asterisks(len(occourances))]


table = {
    "0-10": get_occourances_in_range(0, 10),
    "10-20": get_occourances_in_range(11, 20),
    "20-30": get_occourances_in_range(21, 30),
    "30-40": get_occourances_in_range(31, 40),
    "40-50": get_occourances_in_range(41, 50),
    "50-60": get_occourances_in_range(51, 60),
    "60-70": get_occourances_in_range(61, 70),
}
print(table)
```

Part 3: This code displays the calculated pass mark for the class where at least 60% of the class has passed the exam (to the nearest 10).

```
def students_over(num):
    passed = []
    for item in data:
        if item > num:
            passed.append(item)

    return len(passed)

got_pass_mark = False
i = 0

while got_pass_mark == False:
    passed_percentage = ((students_over(i) / len(data)) * 100)
    rounded_percentage = round(passed_percentage, -1)

    if (rounded_percentage  == 60):
        got_pass_mark = True
        pass_mark = i
        print(f"Pass mark: {pass_mark}")
    i += 1
```

I found it easy to get the min, max of the arrays data and to calculate the averages. I found the calculation of the pass mark to be the most difficult bit as it required some more complex calculations and working with multiple loops to get the desired result. I found it quite easy to complete this activity as we did a similar task in the classroom sessions.