Problem Statement

1) Calculate the BMI (Body Mass Index) using Formula 1, BMI Category and Health risk from Table 1 of the person and add them as 3 new columns

2) Count the total number of overweight people using ranges in the column BMI Category of Table 1, check this is consistent programmatically and add any other observations in the documentation

3) Create build, tests to make sure the code is working as expected and this can be added to an automation build / test / deployment pipeline

BMI Formula = BMI(kg/m) = mass(kg) / height(m)

For example, if you are 175cm = (1.75m) in height and 75kg in weight, you can calculate your BMI as follows: 75kg / (1.75m2) = 24.49kg/m2

BMI Category=['Underweight', 'Normal weight', 'Overweight', 'Moderately obese', 'Severely obese', 'Very severely obese']

BMI Range (kg/m2)

0=18.4 and below

1=18.5 - 24.9

2=25 - 29.9

3=30 - 34.9

4=35 - 39.9

5=40 and above

Health risk: Malnutrition risk, Low risk, Enhanced risk, Medium risk, High risk, Very high risk

Project github link with readme: https://github.com/devendraprasad1984/baincotest

Folder structure / project Dependencies

Run result

npm run app

```
devendras-MacBook-Pro:bain_co dpadmin$ npm run app

> bainnco_assessment@1.0.0 app
> node app.js

printing BMI calculation results for 216 patients
found 24 Overweight patients
found 72 Moderately obese patients
printing only top 5
ROW-1 - He is "Moderately obese" having bmi at "32.83kg/m2" and has "Medium health risk". [height=171cm, weight=96kg]
ROW-2 - He is "Moderately obese" having bmi at "32.79kg/m2" and has "Medium health risk". [height=161cm, weight=85kg]
ROW-3 - He is "Normal weight" having bmi at "23.77kg/m2" and has "Low health risk". [height=180cm, weight=77kg]
ROW-4 - She is "Normal weight" having bmi at "22.5kg/m2" and has "Low health risk". [height=166cm, weight=62kg]
ROW-5 - She is "Moderately obese" having bmi at "31.11kg/m2" and has "Medium health risk". [height=150cm, weight=70kg]
devendras-MacBook-Pro:bain_co dpadmin$
```

Test Result

npm run test

```
 PASS  __TESTS__/checks.test.js
  ✓ check people with overweight counter (3 ms)
  ✓ check people with high risk

 PASS  __TESTS__/print.test.js
  ✓ print test -- hello (21 ms)
  ✓ printRecords with data (3 ms)
  ✓ printRecords with no data
  ✓ printRecords with no data TOHAVEBEENCALLEDWITH (1 ms)

 PASS  __TESTS__/test1.test.js
  ✓ quick test

Test Suites: 3 passed, 3 total
Tests:       7 passed, 7 total
Snapshots:   0 total
Time:        1.173 s
Ran all test suites.
```

npm run testc

```
PASS __TESTS__/checks.test.js
  ✓ check people with overweight counter (2 ms)
  ✓ check people with high risk

PASS __TESTS__/print.test.js
  ✓ print test -- hello (25 ms)
  ✓ printRecords with data (3 ms)
  ✓ printRecords with no data
  ✓ printRecords with no data TOHAVEBEENCALLEDWITH (1 ms)

PASS __TESTS__/test1.test.js
  ✓ quick test
```

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Line #s
All files	97.22	85	91.67	98.53	
__TESTS__	100	100	100	100	
sampledata.js	100	100	100	100	
app	100	100	100	100	
update.js	100	100	100	100	
data	100	100	100	100	
bmi_cat.js	100	100	100	100	
bmi_health_risks.js	100	100	100	100	
bmi_range.js	100	100	100	100	
utils	96.72	83.33	90	98.28	
calculations.js	94.59	83.33	87.5	97.14	55
consts.js	100	100	100	100	
print.js	100	83.33	100	100	12,23