Write a Java Program to solve each one of the problem. Please name your project as **Q#.Java**.
Upload the homework to @homeworkbot with the tag #java-class5

1) **Body mass index.** The body mass index (BMI) is the ratio of the weight of a person (in kilograms) to the square of the height (in meters). Write a program that takes two command-line arguments, weight and height, computes the BMI, and prints out the corresponding BMI category:
   Starvation: less than 15
   Anorexic: less than 17.5
   Underweight: less than 18.5
   Ideal: greater than or equal to 18.5 but less than 25
   Overweight: greater than or equal to 25 but less than 30
   Obese: greater than or equal to 30 but less than 40
   Morbidly Obese: greater than or equal to 40

2) **Reynolds number.** The *Reynolds number* is the ratio if inertial forces to viscous forces and is an important quantity in fluid dynamics. Write a program that takes in 4 command-line arguments, the diameter d, the velocity v, the density rho, and the viscosity mu, and prints out the Reynold's number d * v * rho / mu (assuming all arguments are in SI units). If the Reynold's number is less than 2000, print laminar flow, if it's between 2000 and 4000, print transient flow, and if it's more than 4000, print turbulent flow

3) In the country of Rahmania, the cost of mailing a letter is 40 sinas for letters up to 30 grams. Between 30 g and 50 g, it is 55 sinas. Between 50 g and 100 g, it is 70 sinas. Over 100 g, it costs 70 sinas plus an additional 25 sinas for each additional 50 g (or part thereof). For example, 101 grams would cost $70 + 25 = 95$ sinas. 149 g would also cost 95 sinas, because both of these packages only use the first 50 g over 100 g. Write a program that prompts the user for a mass and then gives the cost of mailing a letter having that mass.

4) One way of giving a direction is to simply use one of the letters N, E, S, or W, indicating the nearest compass point. Another way is to give a bearing as a number from 0 to 359 (since 360 would be back to 0 again). A bearing of 0 corresponds to N, a bearing of 90 corresponds to E, and so on. Write a program that prompts the user for a bearing from 0 to 359 and prints the corresponding letter of he compass point *nearest* to the bearing. For example, input of 73 is between N and E, but it is closer to E, so the program should output E. Bearings exactly halfway between two compass points should produce either N or S (never E or W).

Write a Java Program to solve each one of the problem. Please name your project as **Q#.Java**.
Upload the homework to @homeworkbot with the tag #java-class5

```
         N
     360°/000°



  W                    E
 270°                 090°



         S
       180°
```
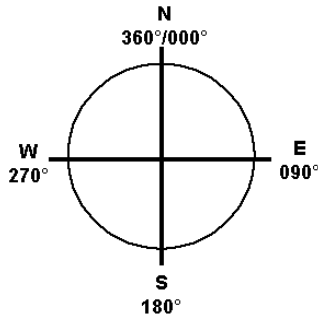
5) (**Spicy Challenge)** Number-to-English. Write a program to read in a command line integer between 0 and 1000 and print out the English equivalent. Here is an exhaustive list of words that your program should use: zero, one, two, three, four, five, six, seven, eight, nine, ten, eleven, twelve, thirteen, fourteen, fifteen, sixteen, seventeen, eighteen, nineteen, twenty, thirty, forty, fifty, sixty, seventy, eighty, ninety, hundred, thousand. Don't use hundred, when you can use thousand, e.g., use one thousand five hundred instead of fifteen hundred