HW1 Basics and Control Statements (100 pts.):

The Safe Keeping Parking Garage determines the price it charges for parking based on the day of the week and the time spent parked. To avoid bleeding its customers dry, the garage has maximum fees it will charge. Here are the rules the company uses to calculate its prices:

Day of Week	Rate for 15 Minutes or Portion Thereof	Maximum Fee Charged
Monday - Friday	1.25	15.00
Saturday & Sunday	0.50	15.00

To help you understand the business process, the garage manager created the following examples to illustrate some parking situations:

Day of Week	Vehicle Arrives at	Vehicle Leaves at	Arrival Minutes	Depart Minutes	Duration in Minutes	15 Minute Intervals	Rate	Charge Based on Rate	Max Charge	Actual Charge
M - F	745	1000	465	600	135	9	1.25	11.25	15	11.25
M - F	700	1200	420	720	300	20	1.25	25	15	15
M - F	1130	1600	690	960	270	18	1.25	22.5	15	15
M - F	1305	1500	785	900	115	8	1.25	10	15	10
M - F	1420	1500	860	900	40	3	1.25	3.75	15	3.75
M - F	750	1715	470	1035	565	38	1.25	47.5	15	15
M - F	1730	1900	1050	1140	90	6	1.25	7.5	15	7.5
M - F	1755	1905	1075	1145	70	5	1.25	6.25	15	6.25
S & S	1400	1920	840	1160	320	22	0.5	11	15	11
S & S	940	1425	580	865	285	19	0.5	9.5	15	9.5

Notice that the arrival in minutes is the number of hours in the arrival time multiplied by 60 plus the number of minutes. For instance, in the first row the "Arrival Minutes" is calculated as 7 hours times 60 giving 420 plus 45 minutes resulting in a total of 465 minutes. Likewise, the "Depart Minutes" is calculated as 10 hours times 60 giving 600 minutes plus 0 minutes resulting in total of 600 minutes. How might you use integer division and modulus operations to separate hours and minutes from an int?

The table above represents time in a 24 hour format using integers. This makes calculating duration much easier. As anyone who has used a parking garage knows, the number of 15 minute intervals counts even a minute over as a new interval. For instance, a duration of 60 minutes produces 4 15 minute intervals, and a duration of 61 minutes produces 5 intervals. In the sixth row above, a duration of 565 minutes is 38 15 minute intervals – 37 full intervals plus 10 minutes of the next one.

Since negative durations cannot exist, display an error message if the user enters an arrival time that exceeds the departure time.

Develop a set of test plan results that shows that your program produces correct results.

Write a Java program to calculate the ampunt charged for parking in the garage over a specified interval on any given day. The program must implements the requirements described above. It will use an object of either the Scanner class or the BufferedReader class to obtain input.

The program should request information from the user in the following format:

Please enter the day of the week : mon

Please enter the vehicle's arrival time : 745

Please enter its departure time : 1000

The program will use printf with the appropriate format specifiers to display the output which should look like this:

Day of week: Mon

Parking duration in minutes: 135, rate: $1.25

Amount charged: $11.25

Be sure to use the correct data type for the variables that will store the user input.

Write an efficient algorithm: try to obtain the correct results using the fewest possible number of steps.

You will also develop a test plan that shows what values you tested and whether the program correctly calculated the cost of parking.

You will upload your test file and your java file compressed together in a zip or 7z file.

Be sure to use the assignment naming convention for your compressed file, for instance, DAngelo HW1.zip.