Chapter 3 – Decisions

A file designed to help with your assignment. The following includes the questions and the corresponding sample solution. It makes sense to study this file first and then attempt to write your own solution for the assignment.

1. **Roulette problem – see the file Roulette.java for a sample solution.**

On a roulette wheel, the pockets are numbered and colored. The following requirements will be stored in one Java file.
Pocket 0 is green
For Pockets 1 through 10, the odd-numbered pockets are red and the even-numbered pockets are black
For pockets 11 through 18, the odd-numbered pockets are black and the even-numbered pockets are red
For pockets 19 through 28, the odd-numbered pockets are red and the even-numbered pockets are black
For pockets 29 through 36, the odd-numbered pockets are black and the even-numbered pockets are red

Write a Java program that asks the user to enter a pocket number and displays whether the color is green, red or black. The program should display an error message if the user enters a number that is outside the range of 0 through 36.

> For example, if the user enters 13, the displayed message indicates the color is black. The number 30 indicates the color is red. The number 100 is out of range. You can use the modulus (%) operator to decide if a number is even or odd.

2. **Rock/paper/Scissors – see the file Rockpaperscissors.java for a sample solution.**

Perform exercise 3.17 at the end of chapter 3 of your textbook. This exercise is based on the “rock, paper, scissors” game. This gives you the opportunity to practice with the Random class. In addition to the requirements in the textbook, issue an error message if the user does not enter a valid number (0, 1, 2). We have not done loops yet, so, you can just end the program if the entry is invalid