

```
public class a_Quzi17 {  
    //17. Math.random() 응용 문제  
    //컴퓨터와 주사위 게임을 해보자!  
    //주사위를 2번 굴려서 합산 눈금 숫자가 큰 사람이 이기는 것으로 한다.  
    public static void main(String[] args) {  
        System.out.println("컴퓨터가 주사위를 굴립니다.");  
  
        int comdice1 = (int) (Math.random() * 6 + 1);  
        int comdice2 = (int) (Math.random() * 6 + 1);  
  
        System.out.println("사용자가 주사위를 굴립니다.");  
  
        int userdice1 = (int) (Math.random() * 6 + 1);  
        int userdice2 = (int) (Math.random() * 6 + 1);  
  
        int comdices = comdice1 + comdice2;  
        int userdices = userdice1 + userdice2;  
  
        if (comdices > userdices) {  
            System.out.printf("컴퓨터 승! |컴퓨터(%d) 유저(%d)\n", comdices, userdices);  
        } else if (comdices < userdices) {  
            System.out.printf("유저 승! 컴퓨터(%d) 유저(%d) \n", comdices, userdices);  
        } else {  
            System.out.println("무승부");  
        }  
    }  
}
```

```

1 ▶ public class b_Quiz17_2 {
2     //17. Math.random() 응용 문제
3     //컴퓨터와 주사위 게임을 해보자!
4     //주사위를 2번 굴려서 합산 눈금 숫자가 큰 사람이 이기는 것으로 한다.
5     // for문을 활용한 문제풀이
6 ▶ public static void main(String[] args) {
7     int com_total = 0, user_total = 0;
8
9     //com_total = 0으로 설정, user_total = 0으로 설정
10    // 0으로 설정하는 이유는 다이스 수식에 영향을 미치기 때문에, 1로 설정하면 다이스 수식의 +1을 제외해도 될듯
11
12    for (int i = 0 ; i < 2; i++) {
13        // int i = 0으로 설정, i가 2미만일때 동작, i = i + 1
14        // 여기서 i는 주사위를 굴리는 횟수, 0, 1일때 굴러가므로 주사위가 2번 굴러감.
15
16        System.out.println("컴퓨터와 사용자가 주사위를 굴립니다.");
17
18        com_total += (int)(Math.random() * 6);
19        user_total += (int)(Math.random() * 6);
20
21        // com_total = com_total + 무작위 숫자(1~6)
22        // user_total = user_total + 무작위 숫자(1~6)
23        //
24        System.out.printf("현재 상황: %d(컴퓨터) vs %d(사용자)\n", com_total, user_total);
25    }
26
27    if (com_total > user_total) {
28        System.out.printf("컴퓨터 승! 점수는 %d(컴퓨터) vs %d(사용자)\n", com_total, user_total);
29    } else if (user_total > com_total) {
30        System.out.printf("사용자 승! 점수는 %d(컴퓨터) vs %d(사용자)\n", com_total, user_total);
31    } else {
32        System.out.printf("무승부! 점수는 %d(컴퓨터) vs %d(사용자)\n", com_total, user_total);
33    }
34 }
35 }
36

```

```
1  ▶ public class c_final {
2  ▶  public static void main(String[] args) {
3      int sum = 0;
4      // 뒤이을 수식에 영향을 미치지 않기 위해 0으로 값 초기설정
5
6      final int Start = 1;
7      final int End = 10;
8      //start =1 , End = 10으로 값 설정
9
10
11     for(int i = Start; i <= End; i++) {
12         sum += i;
13         System.out.println("sum = " + sum);
14
15         // i = 1이고 i가 10이하일때 동작, i는 계속하여 1씩 증가됨
16         //sum 은 i만큼 계속 증감됨. 즉 1씩 계속 증가됨
17         //1~10까지 계속 더해지는것이 출력됨 (1+2, 3+3, 6+4 ....)
18
19     }
20
21     System.out.println("최종 합산 값 = " + sum);
22     // 앞의 수식의 최종값이 출력됨
23     // 위의 수식값은 보이지 않고 최종값만 출력하게 할 수 있을까?
24 }
25 }
26
```

```

1 ▶ public class d_for를while로 {
2 ▶   public static void main(String[] args) {
3       int com_total = 0, user_total = 0;
4       int i = 0;
5
6       // 뒤이을 수식에 영향을 주지 않기위해 0으로 설정
7
8       while(i < 2) {
9           //i가 2이상일 때까지 계속하여 출력한다. 즉, 0과 1이 만족되어 2번 출력됨. 주사위가 2번 굴려진다는 의미
10          System.out.println("주사위를 굴립니다");
11          com_total += (int) (Math.random()*6+1);
12          user_total += (int) (Math.random()*6+1);
13
14          System.out.printf("현재 상황 : %d(컴퓨터) vs %d(사용자)\n", com_total, user_total);
15          i++;
16          // i++이 없다면 0으로 고정되어 있어, 끊임없이 출력됨
17      }
18      if (com_total > user_total) {
19          System.out.printf("컴퓨터 승! 점수는 각각 %d(컴퓨터) %d(사용자)\n", com_total, user_total);
20      } else if (com_total < user_total) {
21          System.out.printf("유저 승! 점수는 각각 %d(컴퓨터) %d(사용자)\n", com_total, user_total);
22      } else {
23          System.out.printf("무승부! 점수는 %d\n", com_total);
24      }
25  }
26 }
27

```

```
1 ▶ public class e_BIT연산자_AND {
2 ▶   public static void main(String[] args) {
3     int num1 = 10, num2 = 8;
4     System.out.printf("%d and %d = %d\n", num1, num2, num1 & num2);
5     // 10&8을 2진법으로 표현하면 8과 동일
6
7     num2 = 138;
8     // 여기서부터 num2는 위의 8이 아닌 138을 의미하는것?, 똑같은 명칭임에도 변수의 값을 계속 바꿀수있는것 같음.
9
10    System.out.printf("%d And %d = %d\n", num1, num2, num1 & num2);
11
12  }
13 }
14
```

```

1  ▶ public class f_쉬프트연산자 {
2  ▶     public static void main(String[] args) {
3      int num1 = 2, num2 = 5, num3 = 10;
4
5
6
7      System.out.printf("%d << %d = %d\n", num1, num2, num1 << num2);
8      System.out.printf("%d << %d = %d\n", num2, num2, num2 << num2);
9      System.out.printf("%d << %d = %d\n", num3, num2, num3 << num2);
10
11     System.out.printf("%d << %d = %d\n", num1, num1, num1 << num1);
12     System.out.printf("%d << %d = %d\n", num2, num1, num2 << num1);
13     System.out.printf("%d << %d = %d\n", num3, num1, num3 << num1);
14
15     System.out.printf("%d << %d = %d\n", num1, num3, num1 << num3);
16     System.out.printf("%d << %d = %d\n", num2, num3, num2 << num3);
17     System.out.printf("%d << %d = %d\n", num3, num3, num3 << num3);
18
19
20     System.out.printf("%d >> %d = %f\n", num2, num1, (float)(num2 >> num1));
21
22     // 쉬프트 연산은 정수형끼리밖에 안된다는 점 주의 - 이로인하여 float없이는 계산이 안됨
23 }
24 }

```

```

1 ▶ public class g_쉬프트연산자심화문제 {
2 ▶     public static void main(String[] args) {
3         // 0 ~ 9까지의 숫자가 중복되지 않게 나오게 만들어보자! (배열 없이)
4         final int BIN = 1;
5         //BIN 2진법의 binary의 약어.
6         //final은 값을 고정시킨다는 것. 함부로 값을 변경시킬시 큰 문제가 발생할 수 있음. 주의해야함.
7         //BIN이 1부터 시작하는 것은 논리적 오류 발생
8
9         int testBit = 0;
10        int randNum;
11        // randNum의 값을 지정하지 않아도 되는 것은 주사위의 값으로 표현될 것이기 때문에?
12        // 실제로 영향을 안주는 0 값으로 바꾸어도 기능상 별 상관이 없음.
13
14        for(int i= 0; i < 10; i++) {
15            randNum = (int) (Math.random() * 10);
16            // 0~9까지; i는 10미만으로 설정; 증감식 i++ 입력
17            // int형 숫자로 0~9까지 랜덤출력을 위한 Math.random * 10.
18            // 즉 i가 0부터 9(Math.random * 10)까지 수를 무작위로 10번(i<10 : i=0부터니 9까지 10번)을 출력하게 됨.
19            // 즉 주사위를 굴리는 횟수, 출력하는 횟수와 관계?
20
21            while ((testBit & (BIN << randNum)) != 0) {
22                randNum = (int) (Math.random() * 10);
23
24                // testBit는 현재 0이므로 0*2의 n승(0~9)=0이 나옴.
25                // 중복이 있는지 검사하는 과정
26                // 중복문제 해결 0 or 2^0, 0 or 2^1, 0 or 2^2.....으로 이어지면서 ^0 ~ ^9
27            }
28
29            System.out.printf("randNum = %d\n", randNum);
30
31            testBit |= (BIN << randNum);
32            // 2^0 ~ 2^9의 합 = 1023
33
34        }
35        System.out.println(testBit);
36        //1023 출력
37    }
}

```

```
1 public class h_인터럽트이벤트 {
2     public static void main(String[] args) throws InterruptedException {
3         for (int i = 0; ; i++) {
4             if ( i % 2 == 0 ) {
5                 System.out.println("짝수");
6             } else {
7                 System.out.println("홀수");
8             }
9
10            Thread.sleep( millis: 500);
11        }
12    }
13 }
```

14 // 여기서 Thread는 인터럽트, 이벤트이며 가장 중요하고, 최우선시되어야 할 작업으로, 무조건 0.5초 대기하게되어있음



# 퀴즈

```
10
11 final int bin = 1;
12
13 int a = 0, b = 0;
14 int input1;
15 int input2;
16
17 for (int i = 0; i < 6; i++) {
18     input1 = (int) (Math.random() * 6 + 5);
19
20     while ((a & (bin << input1)) != 0) {
21         input1 = (int) (Math.random() * 6 + 5);
22     }
23
24     System.out.printf("5부터10 : %d\n", input1);
25     a |= (bin << input1);
26
27 }
28
29
30 for (int i = 0; i < 4; i++) {
31     input2 = (int) (Math.random() * 4 + 7);
32
33     while ((b & (bin << input2)) != 0) {
34         input2 = (int) (Math.random() * 4 + 7);
35     }
36
37
38     System.out.printf("7부터10 : %d\n", input2);
39     b |= (bin << input2);
40
41 }
42
43 }
44
45
```

## 결과값

"C:\Program Files\Java\jdk-15.0.2\bin\java.exe"

5부터10 : 5

5부터10 : 8

5부터10 : 6

5부터10 : 7

5부터10 : 9

5부터10 : 10

7부터10 : 9

7부터10 : 7

7부터10 : 10

7부터10 : 8

Process finished with exit code 0