

한상우 자바 5일차 복습노트

(21. 5. 7. ~ '21. 12.8.)

이상훈선생님.



출처

<https://github.com/bombom23/LectureContents/blob/main/Java/SangwooHan/bucksp/src/buksp0502.java>

```
// 5 / 2^2 = 1.25
// 결론: 오른쪽 쉬프트는 2^n으로 나누되 소수점을 버려야 한다.
System.out.printf("%d >> %d = %f\n", num2, num1, (float)(num2 >> num1));

// 이유:
// 0101 ----> 5
// 0001 ----> 1
//이부분이 이해가 안되었었는데,
// 만약 14<<4 라고 했을때
// 14의 2진법은 1110 이고 우항이 2^4이다
// 이때 *N칸만큼 칸수가 증가 하기에,
// 1110 맨앞 전체가 4칸이라, 4칸모두 감소 0
// 만약 14<<3 이라면,
// 1110 에서 3칸을빼는것이기에,
// 0001이될것이다.

// 종합적 결론:
// 쉬프트 연산은 2^n을 곱하거나 나눈다.
// 안타깝게도 쉬프트 연산은 정수형끼리밖에 안된다.
// 최근에 나온 휴대용 전용 ARM 프로세서에서는 소수점에 대한 쉬프트 연산을 지원하기도 한다.
```

)

출처

<https://github.com/bombom23/LectureContents/blob/main/Java/SangwooHan/bucksp/src/buksp0501.java>

```
public class buksp0501 {
    public static void main(String[] args) {
        int num1 = 10, num2 = 8;
        //추가적인 질문 없습니다.
        // & 이 비트별 AND
        // 관계 연산자에서는 && 형태로 존재하였음
        // 10 ==> 1010
        // 8 ==> 1000 AND
        // -----
        // 8 ==> 1000
        System.out.printf("%d AND %d = %d\n", num1, num2, num1 & num2);

        num2 = 138;
        // 138 ==> 10001010
        // 10 ==> 1010 AND
        // -----
        // 10 ==> 00001010
        System.out.printf("%d AND %d = %d\n", num1, num2, num1 & num2);
    }
}
```

□ 출처

<https://github.com/bombom23/LectureContents/blob/main/Java/SangwooHan/bucksp/src/buksp0503.java>

```
1 public class buksp0503 {
2     public static void main(String[] args) {
3         int num1 = 10, num2 = 5;
4         // 추가적인 질문 없습니다.
5         // | 이 비트연산자 OR
6         // 관계 연산자에서는 || 형태로 존재하였음
7         // 10 ==> 1010
8         // 5 ==> 0101 OR
9         // -----
10        // 15 ==> 1111
11        System.out.printf("%d OR %d = %d\n", num1, num2, num1 | num2);
12
13        num2 = 130;
14
15        // 10 ==> 00001010
16        // 130 ==> 10001000 OR
17        // -----
18        // 138 ==> 10001010
19        System.out.printf("%d OR %d = %d\n", num1, num2, num1 | num2);
20
21        // OR 연산은 합집합 개념, AND 연산은 교집합 개념
22    }
23 }
```

□ 출처

<https://github.com/bombom23/LectureContents/blob/main/Java/SangwooHan/study/src/AndOrShift05.java>

```
1 public class AndOrShift05 {
2     // 0 ~ 9까지의 숫자가 중복되지 않게 나오게 만들어보자! (배열 없이)
3     public static void main(String[] args) throws InterruptedException {
4         final int BIN = 1;
5
6
7         int testBit = 0;
8         int randNum;
9
10
11        for (int i = 0; i < 10; i++) {
12            // 0 ~ 9 가 나올것
13            randNum = (int)(Math.random() * 10);
14            // 4가 나왔다고 하자.
15            // randNum 값 현재 정수 4
16            // while 이말한다 비트연산자 앤드로 testBit 하고 (BIN << randNum) 연산해서
17            // 0이면 나옴과해 근데 !=0이면, randNum을 다시뽑자, 언제까지? 0이 나올때까지.
18            // 현재 testBit 는 0 (BIN << randNum)은 2^4 (쉬프트연산되어)
19            // 0 & 2^4 == 0000000000 AND //
20            // 10000 //
21            // ----- // <- while 식
22            // 0 //
23            // 앤드연산 하여 0이 나오면, testBit에 중복이 없는걸로 건주.
24            // 만약있으면, testBit에 중복이 있는걸로 간주, 하여 중복이 안될때까지 계속 돌린다.
25            while ((testBit & (BIN << randNum)) != 0) {
26                System.out.println("저기요?" + randNum + "은 중복입니다 다시뽑아주시죠");
27                randNum = (int)(Math.random() * 10);
28            }
29
30            System.out.printf("randNum = %d\n", randNum);
31
32
33            testBit |= (BIN << randNum);
34            // 위에 randNum 의 값들을 저장하여, 이전에 나왔던 randNum이 안뜨게끔
35            // randNum의 2진법 공간들을 요하는 역할을 하는 핵심적인 로직이다.
36        }
```

```
1 import java.util.Scanner;
2
3 public class buksp0505 {
4     public static void main(String[] args) throws InterruptedException {
5         Scanner scan = new Scanner(System.in);
6         System.out.println(" 이번주로프 번호를 뽑으시겠습니까?? 1번 [예]2번[아니요]?");
7         int i = scan.nextInt();
8         final int BIT =1;
9         int randNum;
10        int testBit = 0;
11        if(i ==1)
12        {
13            for(int j = 1; j <10; j++)
14            {
15                Thread.sleep( millis 250);
16                randNum = ((int)(Math.random()*45+1));
17                while ((testBit & (BIT <<randNum))!=0)
18                {
19                    randNum = ((int)(Math.random()*45+1));
20                }
21                System.out.println("이번주 황운의로프번호는["+randNum+"]입니다..!");
22                testBit |= (BIT<<randNum);
23            }
24        }
25        else{
26            System.out.println("다음주연속 로프합시다..");
27        }
28    }
29 }
```

Run buksp0505

"C:\Program Files\Java\jdk-11.0.10\bin\java.exe" -javaagent
이번주로프 번호를 뽑으시겠습니까?? 1번 [예]2번[아니요]

이번주 황운의로프번호는[39]입니다..!
이번주 황운의로프번호는[39]입니다..!
이번주 황운의로프번호는[31]입니다..!
이번주 황운의로프번호는[1]입니다..!
이번주 황운의로프번호는[6]입니다..!
이번주 황운의로프번호는[19]입니다..!
이번주 황운의로프번호는[16]입니다..!
이번주 황운의로프번호는[43]입니다..!
이번주 황운의로프번호는[22]입니다..!
Process finished with exit code 0

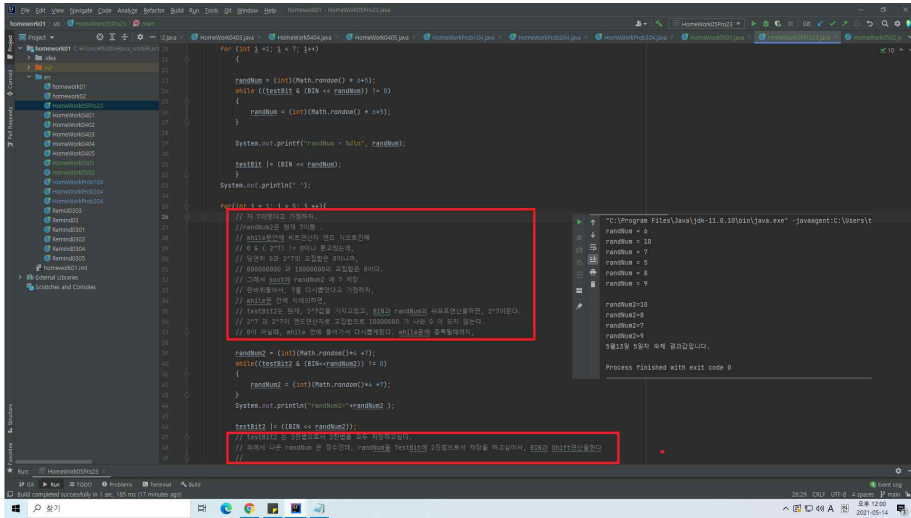
Run buksp0505

"C:\Program Files\Java\jdk-11.0.10\bin\java.exe" -javaagent
이번주로프 번호를 뽑으시겠습니까?? 1번 [예]2번[아니요]

다음주연속 로프합시다..
Process finished with exit code 0

출처

<https://github.com/bombom23/LectureContents/blob/main/Java/SangwooHan/bucksp/src/buksp0505.Java>



출치

<https://github.com/bombom2/LectureContents/blob/main/Java/SangwooHan/homework01/src/HomeWork05Pro23.java>