

## 1. 좌표에 X표하기: 입력한 2차원 좌표의 위치에 X를 표시

```

import java.util.*;

public class Ex {
    public static void main(String[] args) {
        final int SIZE = 10;
        int x = 0, y = 0;

        char[][] board = new char[SIZE][SIZE];
        byte[][] shipBoard = {
            //0  1  2  3  4  5  6  7  8  9    // 0
            {0, 0, 0, 0, 0, 0, 1, 0, 0}, // 1
            {1, 1, 1, 1, 0, 0, 1, 0, 0}, // 2
            {0, 0, 0, 0, 0, 0, 1, 0, 0}, // 3
            {0, 0, 0, 0, 0, 0, 1, 0, 0}, // 4
            {0, 0, 0, 0, 0, 0, 0, 0, 0}, // 5
            {1, 1, 0, 1, 0, 0, 0, 0, 0}, // 6
            {0, 0, 0, 1, 0, 0, 0, 0, 0}, // 7
            {0, 0, 0, 1, 0, 0, 0, 0, 0}, // 8
            {0, 0, 0, 0, 0, 1, 1, 1, 0}, // 9
        };

        // 배열 board는 좌표를 쉽게 입력하기 위한 행번호와 열번호가 필요하다
        // 그래서 배열 board가 배열 shipBoard보다 행과 열의 길이가 1씩 큰 것이다
        // board 1행에 행번호를, 1열에 열 번호를 저장
        for (int i = 1; i < SIZE; i++) {
            board[0][i] = board[i][0] = (char) (i + '0'); // 숫자를 문자로 변환
        }

        // x, y 좌표 입력 받기 (종료는 00)
        Scanner scan = new Scanner(System.in);

        while (true) {
            System.out.printf("좌표를 입력하세요 (종료는 00) : ");
            String input = scan.nextLine();

            if (input.length() == 2) { // 두 글자를 입력한 경우
                x = input.charAt(0) - '0'; // 첫번째 문자를 숫자로 변환해 x에 저장
                y = input.charAt(1) - '0'; // 두번째 문자를 숫자로 변환해 y에 저장
            }

            if (x == 0 && y == 0) {
                break;
            }

            if (input.length() != 2 || x <= 0 || y <= 0 || x >= SIZE || y >= SIZE) {
                System.out.println("잘못된 입력입니다. 다시 입력해 주세요.");
                continue;
            }
        }
    }
}

```

```

// 0은 바다, 1은 배가 있는 곳.
// shipBoard[x-1][y-1]의 값이 1이면, '0'를, 아니면 'X'를 board[x][y]에 저장한다
board[x][y] = shipBoard[x - 1][y - 1] == 1 ? '0' : 'X';

// 배열 board의 내용을 화면에 출력한다.
// print 메서드로 1차원 char배열의 참조변수를 출력하면, 배열의 모든 요소를 한 줄로
// char배열인 경우에만 가능하다.
for (int i = 0; i < SIZE; i++) {
    System.out.println(board[i]);
}

}

}

```

2. 빙고: 빙고판을 만들고 입력받은 숫자를 빙고판에서 지운다

```

import java.util.*;

public class Ex {
    public static void main(String[] args) {
        final int SIZE = 5;
        int x = 0, y = 0, num = 0;

        int[][] bingo = new int[SIZE][SIZE];
        Scanner scan = new Scanner(System.in);

        // 배열의 모든 요소를 1부터 SIZE*SIZE까지의 숫자로 초기화
        for (int i = 0; i < bingo.length; i++) {
            for (int j = 0; j < bingo.length; j++) {
                bingo[i][j] = i * SIZE + j + 1;
            }
        }

        // 배열에 저장된 값 뒤섞기(shuffle)
        // bingo[i][j]와 임의로 선택된 값 bingo[x][y] 바꾸기.
        for (int i = 0; i < SIZE; i++) {
            for (int j = 0; j < SIZE; j++) {
                x = (int) (Math.random() * SIZE);
                y = (int) (Math.random() * SIZE);

                int temp = bingo[i][j];
                bingo[i][j] = bingo[x][y];
                bingo[x][y] = temp;
            }
        }

        do {
            for (int i = 0; i < SIZE; i++) {
                for (int j = 0; j < SIZE; j++) {
                    System.out.printf("%2d ", bingo[i][j]);
                }
                System.out.println();
            }

            System.out.printf("1~%d의 숫자를 입력하세요 (종료:0) : ", SIZE * SIZE);
            num = scan.nextInt();

            // 입력받은 숫자와 같은 숫자가 저장된 요소를 찾아서 0을 저장
            outer:
            // 이중 반복문이므로 빠져나올 때를 위해 이름 붙여주기
            for (int i = 0; i < SIZE; i++) {
                for (int j = 0; j < SIZE; j++) {
                    if (bingo[i][j] == num) {
                        bingo[i][j] = 0;
                        break outer; // 이중 반복문 벗어나기.
                    }
                }
            }
        } while (num != 0);
    }
}

```

### 3. 행렬의 곱셈: 두 행렬(matrix)을 곱한 결과를 출력

수학에서 두 개의 행렬(matrix) m1과 m2가 있을 때, 이 두 행렬을 곱한 결과인 행렬 m3는 아래와 같이 정의된다.

$$\begin{matrix} & m1 & & m2 & & m3 \\ \begin{pmatrix} A0 & A1 & A2 \\ B0 & B1 & B2 \end{pmatrix} & \times & \begin{pmatrix} a0 & a1 \\ b0 & b1 \\ c0 & c1 \end{pmatrix} & = & \begin{pmatrix} A0 \times a0 + A1 \times b0 + A2 \times c0 & A0 \times a1 + A1 \times b1 + A2 \times c1 \\ B0 \times a0 + B1 \times b0 + B2 \times c0 & B0 \times a1 + B1 \times b1 + B2 \times c1 \end{pmatrix} \end{matrix}$$

두 행렬의 곱셈이 가능하려면, m1의 열의 길이와 m2의 행의 길이가 같아야 한다는 조건이 있다. 위의 경우에는 m1이 2행 3열이고, m2가 3행 2열이므로 곱셈이 가능하다.

그리고 곱셈연산의 결과인 행렬 m3의 행의 길이는 m1의 행의 길이와 같고, 열의 길이는 m2의 열의 길이와 같다. 2행 3열인 행렬과 3행 2열인 행렬을 곱하면 결과는 2행 2열의 행렬이 되는 것이다.

```

import java.util.Arrays;

public class Ex {
    public static void main(String[] args) {
        int[][] m1 = {
            {1, 2, 3},
            {4, 5, 6}
        };

        int[][] m2 = {
            {1, 2},
            {3, 4},
            {5, 6}
        };

        // 두 행렬의 곱셈이 가능하려면 m1의 열의 길이와 m2의 행의 길이가 같아야 함
        // 두 행렬의 곱한 결과는 m1의 행의 길이 x m2의 열의 길이
        final int ROW = m1.length;    // m1의 행의 길이 (m3의 행의 길이)
        final int COL = m2[0].length;  // m2의 열 길이 (m2의 열의 길이)

        int[][] m3 = new int[ROW][COL];

        // 행렬곱 m1 x m2의 결과를 m3에 저장
        final int M2_ROW = m2.length; // m2의 행 길이

        for (int i = 0; i < ROW; i++) {
            for (int j = 0; j < COL; j++) {
                for (int k = 0; k < M2_ROW; k++) {
                    m3[i][j] += m1[i][k] * m2[k][j];
                    // m3: m1의 행의 길이 (ROW)와 m2의 열의 길이 (COL)
                    // m1: m1의 행의 길이 (ROW), m1의 열의 길이 = m2의 행의 길이 (M2_ROW)
                    // m2: m2의 행의 길이 (M2_ROW), 2의 열의 길이 (COL)
                }
            }
        }

        // 행렬 m3 출력
        System.out.println(Arrays.deepToString(m3));

        for (int i = 0; i < ROW; i++) {
            for (int j = 0; j < COL; j++) {
                System.out.printf("%3d ", m3[i][j]);
            }
            System.out.println();
        }
    }
}

```

4. 단어 맞추기: 영어 단어를 보여주고, 뜻을 맞추는 게임

```

package ch5;

import java.util.*;

class MultiArrEx4{
    public static void main(String[] args) {
        String[][] words = {
            {"chair", "의자"},           // words[0][0], words[0][1]
            {"computer", "컴퓨터"},      // words[1][0], words[1][1]
            {"integer", "정수"}          // words[2][0], words[2][1]
        };

        Scanner scanner = new Scanner(System.in);

        for(int i=0; i<words.length; i++) {
            System.out.printf("Q%d. %s의 뜻은?", i+1, words[i][0]);

            String tmp = scanner.nextLine();

            if(tmp.equals(words[i][1])) {
                System.out.printf("정답입니다. %n%n");
            } else {
                System.out.printf("틀렸습니다. 정답은 %s입니다. %n%n", words[i][1]);
            }
        } // for
    } // main의 끝
}

```

```

import java.util.*;

public class Ex {
    public static void main(String[] args) {
        String[][] words = {
            {"컴퓨터", "Computer"},
            {"책", "Book"},
            {"시계", "Watch"}
        };

        Scanner scan = new Scanner(System.in);

        for (int i = 0; i < words.length; i++) {
            System.out.print(words[i][0] + "를 영어로 하면?: ");
            String guess = scan.nextLine();
            if (guess.equalsIgnoreCase(words[i][1])) {
                System.out.println("정답입니다!");
            } else {
                System.out.print("틀렸습니다. ");
                System.out.println(words[i][0] + "을 영어로 하면 " + words[i][1] + " 입니다.");
            }
        }
    }
}

```

