

# 프로그래밍 언어응용 문제해결 시나리오

AI기반 스마트제조공정 빅데이터 분석 시스템개발1차  
[토끼게임][RabbitGame]

2022.08.03. 이국두

# 목차

1. 토끼 게임[RabbitGame]에 대한 설명
2. 토끼 게임의 플로우차트(Flow Chart)
3. 토끼 게임의 Play영상
4. 코드 작성의 종류(배열),(doWhile),(tryCatch),(if)
5. 코드 작성의 종류 (ArrayList),(상속),(람다식)
6. 코드 작성의 종류 (doWhile),(switch),(for반복문)
7. 코드 작성의 종류 (While),(결과)

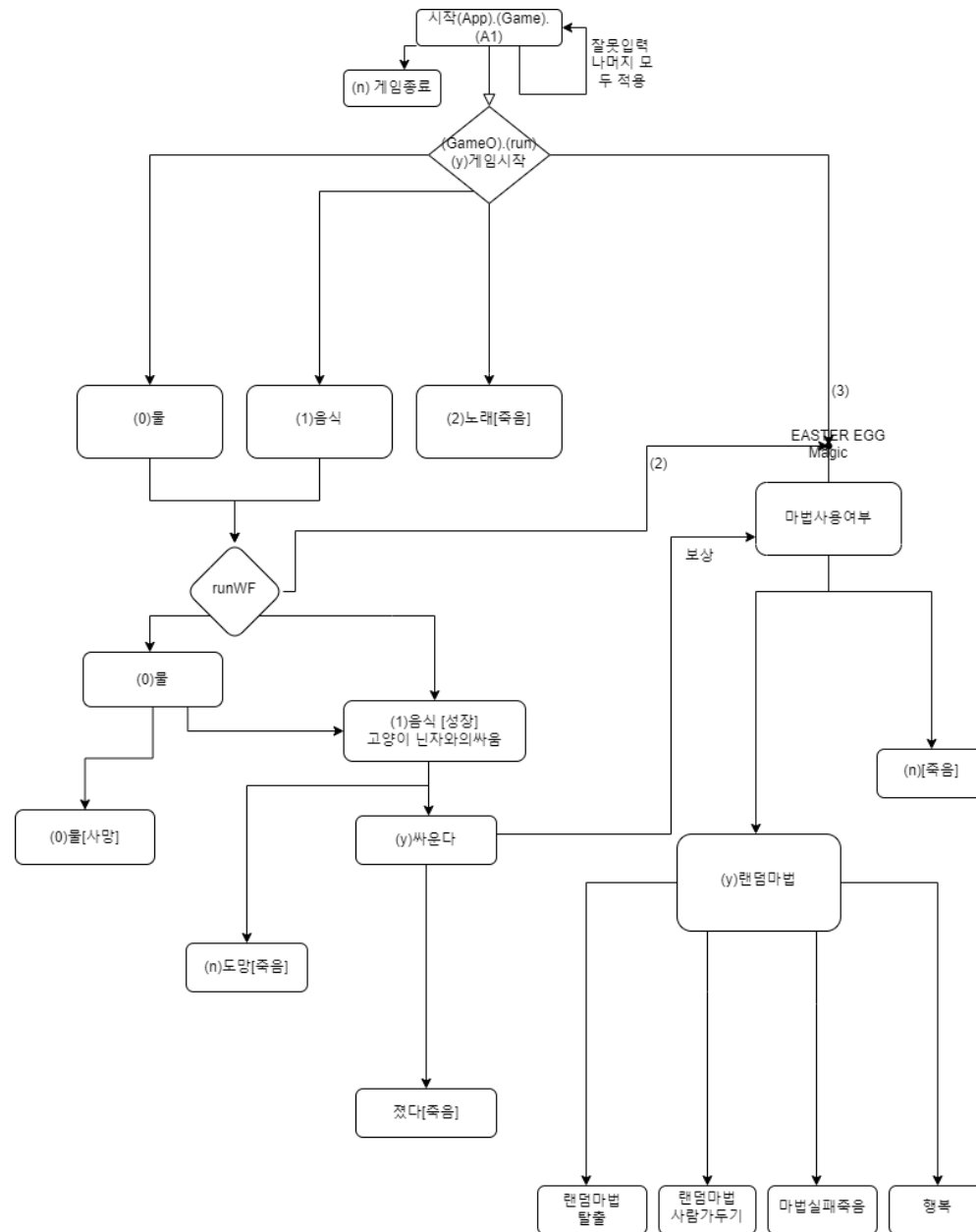
# 1. 토끼 게임[RabbitGame]에 대한 설명

토끼를 키우는 게임이다.

토끼를 키우면서 발생하는 이벤트들로 게임의 유저가 당황하거나 즐거워 할 수 있다.

토끼는 매우 잘 죽을 수 있다. 열심히 키워서 마음에 드는 엔딩을 찾아보자!

## 2. 토끼 게임의 플로우차트(Flow Chart)



The image is a screenshot of an IDE, likely IntelliJ IDEA, showing a Java source file named 'App.java'. The code defines a public class 'App' with a main method. Inside the main method, a 'Game' object is instantiated and its 'A1()' method is called. The console output shows a separator line of equals signs, followed by the text '====강종====', another separator line, and then the prompt '++++++ 당신의 이름은 무엇입니까? ++++++'. The IDE's interface includes a top menu bar, a toolbar, a left sidebar with a file explorer, and a bottom panel with tabs for 'Problems', 'Console', 'Javadoc', 'Declaration', 'Debug', and 'Coverage'. The 'Console' tab is active, displaying the program's output.

## 4. 코드 작성의 종류(배열),(doWhile),(tryCatch),(if)

```
public class Game0 {
    GameObject[] objects = { new Water(), new Food(), new Sing(), new Magic() };
    MagicS[] MS = { new CageEscape(), new UserEscape(), new DoNotEscape(), new DoNotEscape2() };
    GameObject2[] objects2 = { new Water2(), new Food2(), new Magic2() };

    boolean a = false;

    public void run() {
        System.out.println("♡나를 잘 키워줘♡");
        System.out.println("+++++++");
        System.out.println("+      +");
        System.out.println("+      ♡      +");
        System.out.println("+++++++");

        System.out.print("물# 음식# 노래# [0,1,2] 입력해주세요");
        System.out.println();

        do {
            String line = scanner.nextLine(); // 문자열로 받는다.
            try {
                int rum = Integer.parseInt(line); // 정수로 바꿔준다.

                if (!(rum == 0 || rum == 1 || rum == 2 || rum == 3)) {
                    System.out.println("[0,1,2]중 선택하세요");

                    run();
                } else if (rum == 3) { // 숨은 효과
                    objects[rum].getN1().forEach(s -> System.out.println(s));
                    runMagic();
                } else if (rum == 2) {
                    objects[rum].getN1().forEach(s -> System.out.println(s));
                    System.exit(0);
                } else if (rum == 0) {
                    objects[rum].getN1().forEach(s -> System.out.println(s));
                    System.out.println();
                    System.out.println("토끼가 신나서 폴짝 뛰었어요!");
                    System.out.println("토끼가 물과 음식을 내놓으라 합니다.");
                    System.out.print("물# 음식# [0,1] 선택 해주세요");
                    System.out.println();
                    runWF();
                } else {
                    objects[rum].getN1().forEach(s -> System.out.println(s));
                    System.out.println();
                    System.out.println("토끼가 신나서 폴짝 뛰었어요!");
                    System.out.println("토끼가 물과 음식을 내놓으라 합니다.");
                    System.out.print("물# 음식# [0,1] 선택 해주세요");
                    System.out.println();
                    runWF();
                }
            } catch (NumberFormatException e) {
                System.out.println("잘못선택하셨습니다.");
                System.out.print("게임을 다시 시작합니다.");
                System.out.println();
                run();
                continue;
            }
        } while (!a);

        scanner.close();
    }
}
```

## 5. 코드 작성의 종류 (ArrayList),(상속),(람다식)

```
import java.util.ArrayList;
```

```
public class GameObject {  
    protected int id;  
    protected String name0;  
    List<String> n1 = new ArrayList<String>();  
  
    ..  
  
    objects[rum].getN1().forEach(s -> System.out.println(s));
```

```
public class Food extends GameObject {  
    public Food() {  
        setId(1);  
        setName0("맛있는 풀");  
  
        n1.add("맛있는 풀을 선택하셨습니다.");  
        n1.add("맛있는 풀을 토끼가 맛있게 먹습니다.");  
        n1.add("+++++++");  
        n1.add("+      히히  +");  
        n1.add("+      🐰🐰  +");  
        n1.add("+++++++");  
        n1.add("HP가 생성됩니다.");  
    }  
}
```

## 6. 코드 작성의 종류 (doWhile),(switch),(for반복문)

```
do {
    String line4 = scanner4.nextLine();
    switch (line4) {
        case "y":
            System.out.println("공격한다.");
            // 격투 시작
            battle(cat, rabbit);

            break;
        case "n":
            System.out.println("토끼와 함께 도망칩니다!!ㄹㄹㄹ");
            System.out.println("토끼가 도망치다가 발목부상을 입었습니다! 상태이상에 빠집니다.");
            System.out.println("토끼 사망!");
            System.out.println("+++++++");
            System.out.println("+          +");
            System.out.println("+      🐰      +");
            System.out.println("+++++++");
            System.out.println("게임을 종료 합니다.");
            System.exit(0);
            break;
        default:
            System.out.println("잘못 선택하셨습니다.");
            System.out.println("몬스터와 싸우시겠습니까?");
            System.out.println("[y/n]다시 선택해주세요");

            break;
    }
} while (!a);
```

```
int hp = 1;
for (int i = 0; i < hp; i++) {
    if (hp == 31) {
        break;
    }
    System.out.println();
    hp += 5;
    System.out.println("Hp = " + (hp - 1));
}
```



## 7. 코드 작성의 종류 (While),(결과)

```
public void battle(Monster b, Monster a) {
    /* 2. 전투를 시작한다. */
    boolean firstAttack = true; // firstAttack이 참이면 a가 펀치를 날린다.
    while ((a.getHp() > 0) && (b.getHp() > 0)) {
        if (firstAttack) {
            a.punch(b);
            System.out.printf("[%s]의 펀치%n", a.getName());
            System.out.printf("%s: %d/30%n", b.getName(), b.getHp());
            System.out.println();

        } else {
            b.punch(a);
            System.out.printf("[%s]의 펀치%n", b.getName());
            System.out.printf("%s: %d/30%n", a.getName(), a.getHp());

        }
        firstAttack = !firstAttack;
    }
    if (a.getHp() > 0) {
        System.out.println(a.getName() + "가 이겼다!!");
        System.out.println("축하합니다.");
        System.out.println("보상을 지급합니다.");
        System.out.println("🎁🎁🎁🎁🎁");
        objects[3].getN1().forEach(s -> System.out.println(s));
        runMagic();
    } else {
        System.out.println(b.getName() + "가 이겼다!!");
        System.out.println("토끼가 비상식량을 모두 빼앗겼습니다.");
        System.out.println("토끼가 추겨! 사태이사에 빠집니다.");
    }
}
```

감사합니다.