

รายงาน

Mini Project : Catch Eggs

เสนอ

อาจารย์ สถิตย์ ประสมพันธ์

จัดทำโดย

นางสาว ธัญญาทิพย์ สอนวงศา

รหัสนักศึกษา 6204062630050

Section 2 RC

รายงานนี้เป็นส่วนหนึ่งของรายวิชา Object-Oriented Programming

ภาคเรียนที่ 1 ปีการศึกษา 2564

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

บทที่ 1 บทนำ

* ที่มาและความสำคัญ :

Catch Eggs เป็นเกมที่พัฒนาโดยใช้ความรู้ภาษา Java ออกแบบและเขียนโปรแกรมเกม ทั้งแนวคิด OOP สามารถเล่นได้เพื่อความสนุกสนานเพลิดเพลิน

* ประเภทของโปรเจ็ค :

โปรแกรมเกม

* ประโยชน์ :

1. เพื่อความสนุกสนาน
2. ฝึกสมาธิ
3. ฝึกใช้ความอดทน
4. ฝึกความแม่นยำ

* ขอบเขตของโปรเจ็ค :

ใช้ความรู้ภาษา Java ออกแบบและเขียนโปรแกรมเกม ทั้งแนวคิด OOP การทำงานของ GUI Graphics Timer

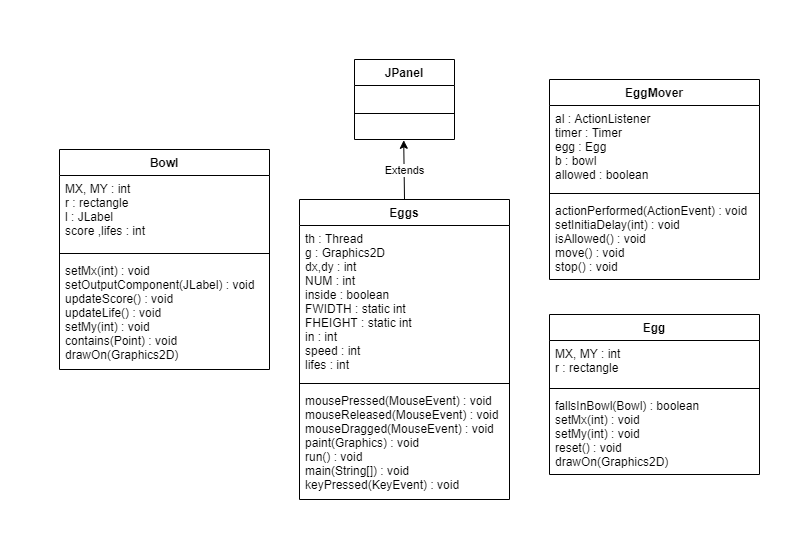
บทที่ 2 ส่วนการพัฒนา

* เนื่อเรื่อง :

เป็นเกมที่ให้ผู้เล่นบังคับทิศทางของถ้วยเพื่อเก็บไข่ที่กำลังเลื่อนลงมา

* วิธีเล่น :

ใช้เม้าท์คลิ๊กที่ถ้วย เพื่อบังคับทิศทางของถ้วย โดยการเก็บไข่ได้ 1 ฟอง จะได้คะแนน(Score) ครั้งละ 10 ซึ่งเริ่มเกมผู้เล่นจะได้รับจำนวนชีวิต(Life) = 30 หากผู้เล่นทำไข่ตกถึงพื้น จะถูกลดจำนวนชีวิตลงครั้งละ 1 เมื่อจำนวนชีวิตเป็น 0 เท่ากับจบเกม

**UML class diagram**

* อธิบาย class diagram

Eggs – ใช้รันเกม ควบคุมการสร้างหน้าเกม

EggMover – ทำให้ไข่เคลื่อนที่

Egg – ให้ไข่มีคะแนนเมื่อลงไปในถ้วย

Bowl – ทำให้ถ้วยนับคะแนนเมื่อเก็บไข่ได้

* รูปแบบการพัฒนาโปรแกรม

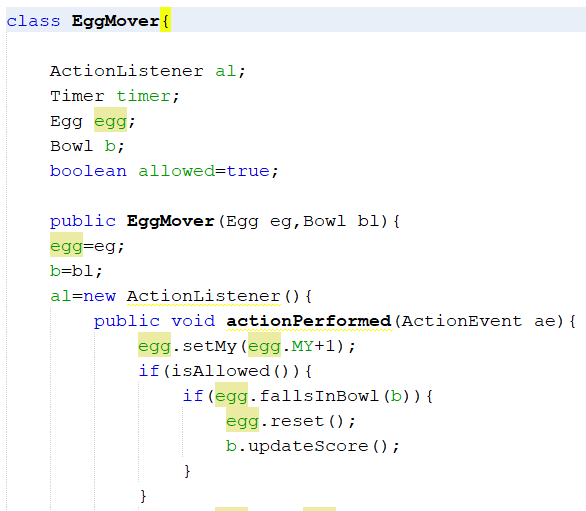
พัฒนาแบบ Application โดยทำงานบน JFrame

* โครงสร้างของโปรแกรมตามแนวคิด OOP

Contructor

Mathod ที่มีชื่อเดียวกับชื่อคลาส ใช้งานครั้งเดียวตอนสร้างวัตถุ เพื่อกำหนดค่าเริ่มต้นและการทำงาน

ตัวอย่าง :



Encapsulation

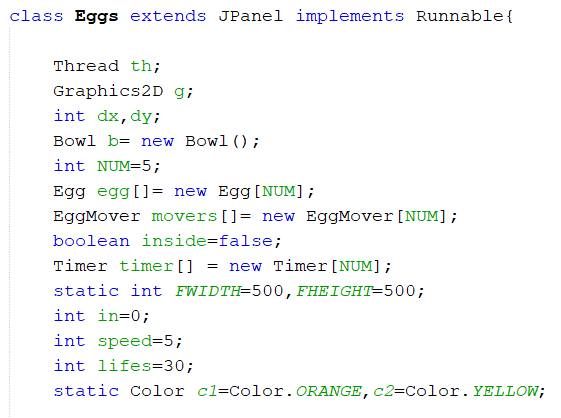
การห่อหุ้มตัวแปรด้วย keyword private (หรือ protected เพื่อให้ตัวแปรนั้นปรากฎในคลาสที่สืบทอดมา) เพื่อป้อนกันการแก้และเรียกใช้งานตัวแปรในวัตถุนั้น จะใช้งานตัวแปรในวัตถุนั้นต้องทำผ่าน Method เท่านั้น

ตัวอย่าง : ไม่มีการใช้ Encapsulation

Inheritance

การสืบทอดคลาส ให้คลาสที่รับการสืบทอด (คลาสลูก) ได้ตัวแปรและการทำงานจากคลาสที่สืบทอด(คลาสแม่) และยังเพิ่มการทำงานเฉพาะให้คลาสลูกได้ด้วย

ตัวอย่าง :

****

Polymorphism

การที่ตัวแปรที่มี type ของคลาสแม่สามารถเรียกตัวแปรของคลาสลูกได้ โดยดูจาก type ของวัตถุที่รับมา

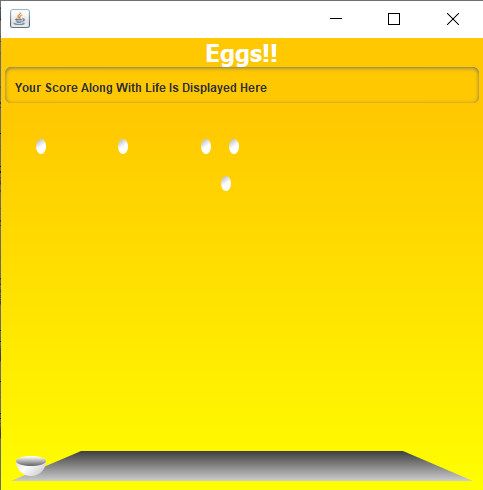
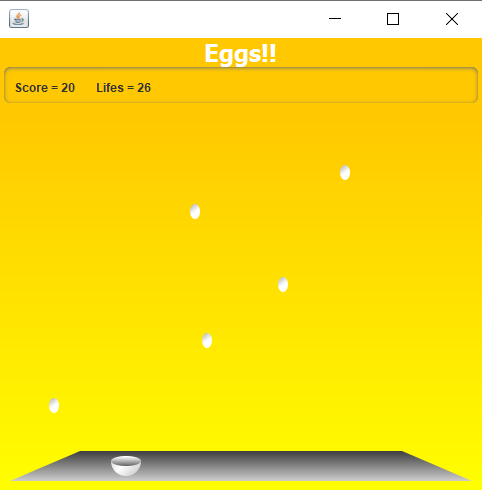
ตัวอย่าง : ไม่มีการใช้ Polymorphism

Abstract

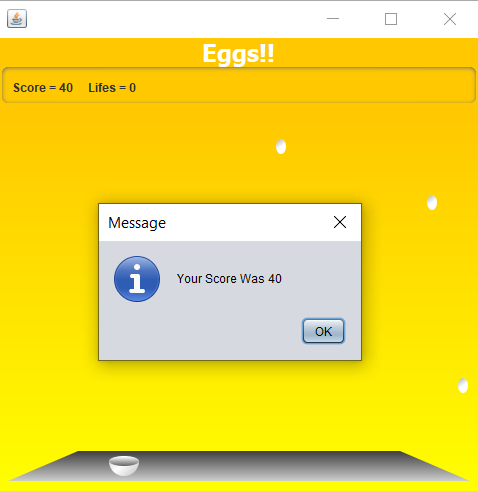
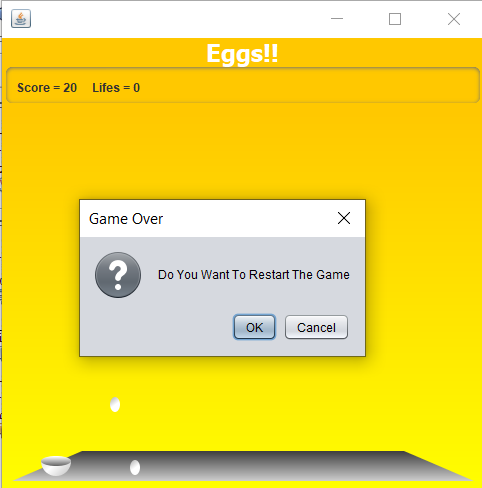
คลาสที่มี Method ที่ไม่มีการทำงาน คลาสที่สืบทอดมาต้องมี Method ที่ไม่มีการทำงานมาจากคลาสแม่

ตัวอย่าง : ไม่มีการใช้ Abstract

* หน้าโปรแกรมเกม

เมื่อเริ่มเกม จะปรากฏหน้าหลักของเกม กดลูกศร ซ้าย-ขวา เพื่อบังคับทิศทางของถ้วย ซึ่งการเก็บไข่ได้ 1 ฟอง จะได้ Score ครั้งละ 10 โดยเริ่มเกมผู้เล่นจะได้รับจำนวน Life = 30

หากผู้เล่นทำไข่ตกถึงพื้น จะถูกลดจำนวนชีวิตลง หน้าต่างเพื่อเริ่มเกมใหม่หรือออกจากเกม

ครั้งละ 1 เมื่อจำนวนชีวิตเป็น 0 เท่ากับจบเกม

จะมีหน้าต่างแสดงคะแนนที่ผู้เล่นทำได้ปรากฎ

บทที่ 3 สรุป

* ปัญหาที่พบระหว่างการพัฒนา :

1). ปรับความเร็วการเคลื่อนที่ไม่ได้

2). การจัดวาง node ต่างๆ ไม่ได้ตามต้องการ

* จุดเด่นของโปรแกรม :

1). รูปร่างรูปทรงต่างๆในเกมใช้ Graphics ของ java วาดทั้งหมด

2). เป็นเกมที่ฝึกความแม่นยำและความอดทนได้ดี