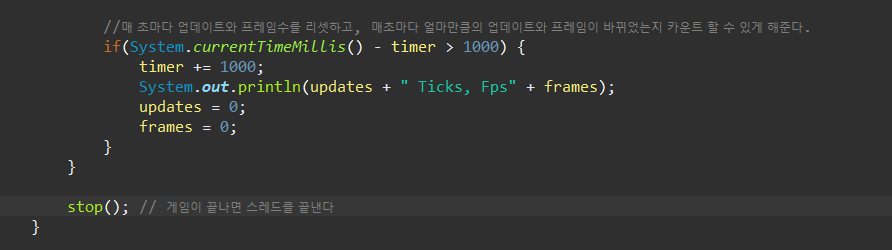
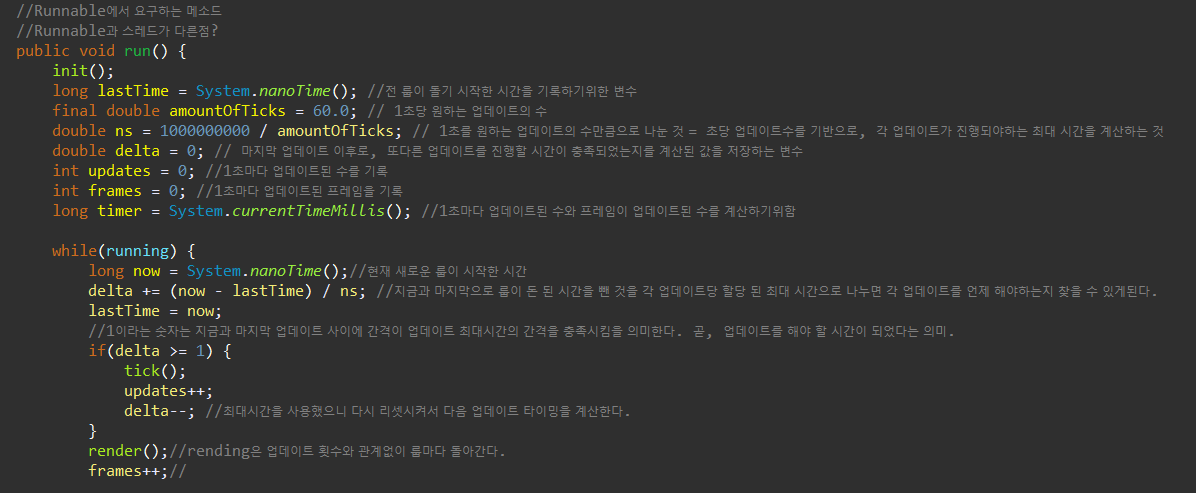
사용된 알고리즘

게임룹 (요약은 스크린샷에 메모됌)



왜 TICK의 수는 59-61을 유지하는데 Fps는 숫자가 이렇게 큰가?

바로(if delta >= 1)때문임. 이 조건문 때문에, tick은 매번 룹이 돌 때마다 업데이트되는 것이아니다. 그러나 fps는 룹이 도는 만큼 증가한다. 즉 fps는 위에 스크린샷의 코드로는 룹 카운터라고해도 무방하다.

Synchronized

<https://www.geeksforgeeks.org/synchronized-in-java/>

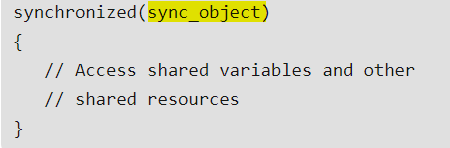
Synchronized in Java

[Multi-threaded](http://quiz.geeksforgeeks.org/multithreading-in-java/)programs may often come to a situation where multiple threads try to access the same resources and finally produce erroneous and unforeseen results.

So it needs to be made sure by some synchronization method that only one thread can access the resource at a given point of time.

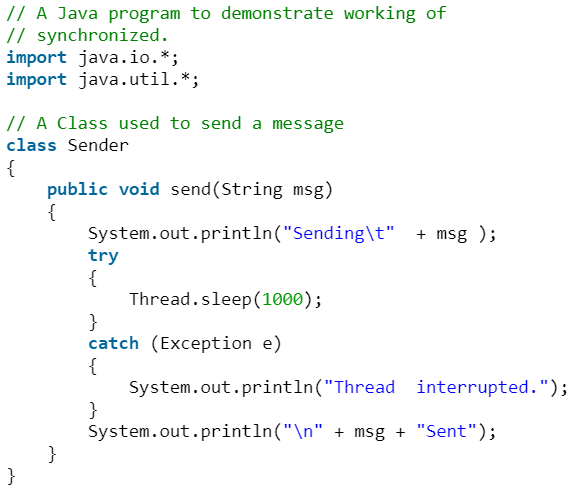
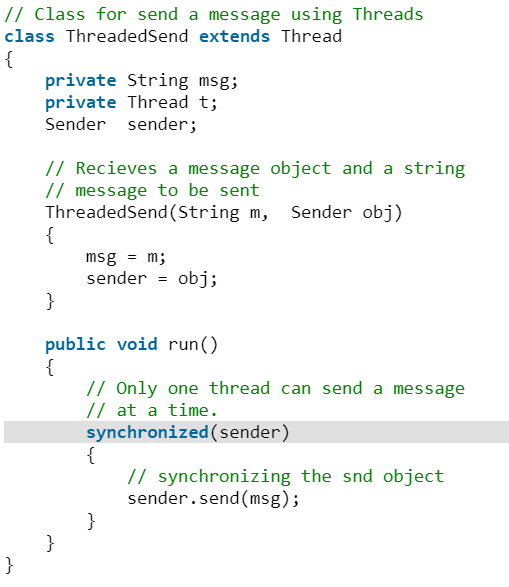
Java provides a way of creating threads and synchronizing their task by using synchronized blocks.

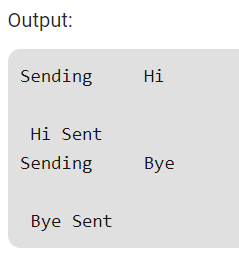
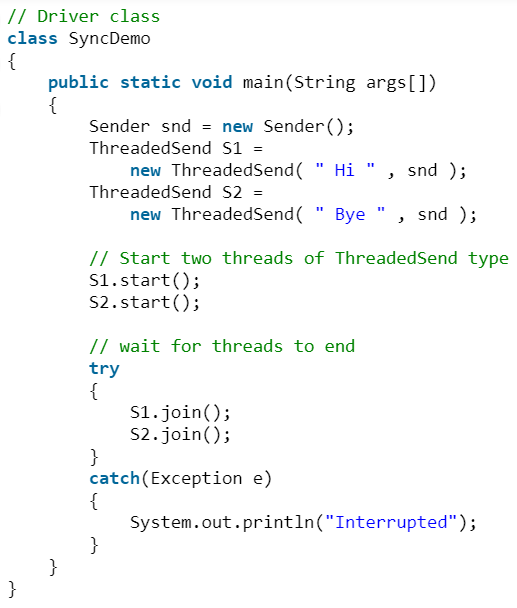
A synchronized block in Java is synchronized on some object. All synchronized blocks synchronized on the same object can only have one thread executing inside them at a time.



This synchronization is implemented in Java with a concept called monitors. Only one thread can own a monitor at a given time. When a thread acquires a lock, it is said to have entered the monitor. All other threads attempting to enter the locked monitor will be suspended until the first thread exits the monitor.

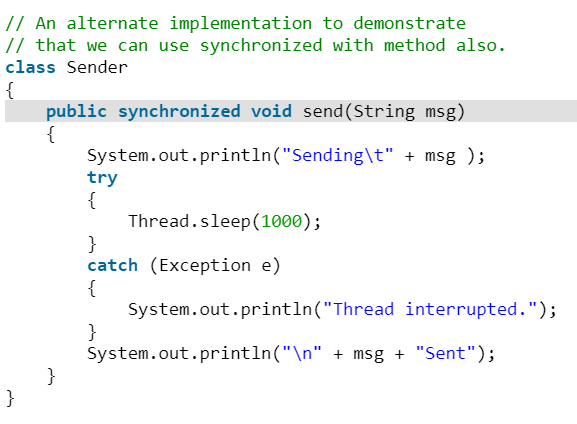
Following is an example of multi threading with synchronized.

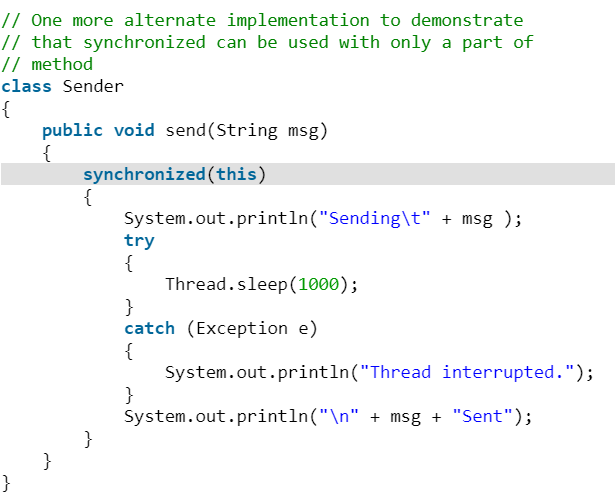


The output is same every-time we run the program.

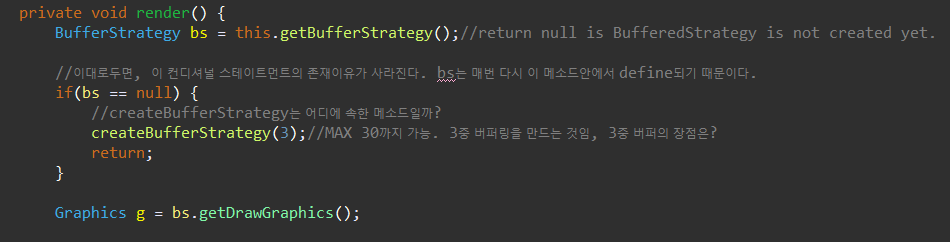
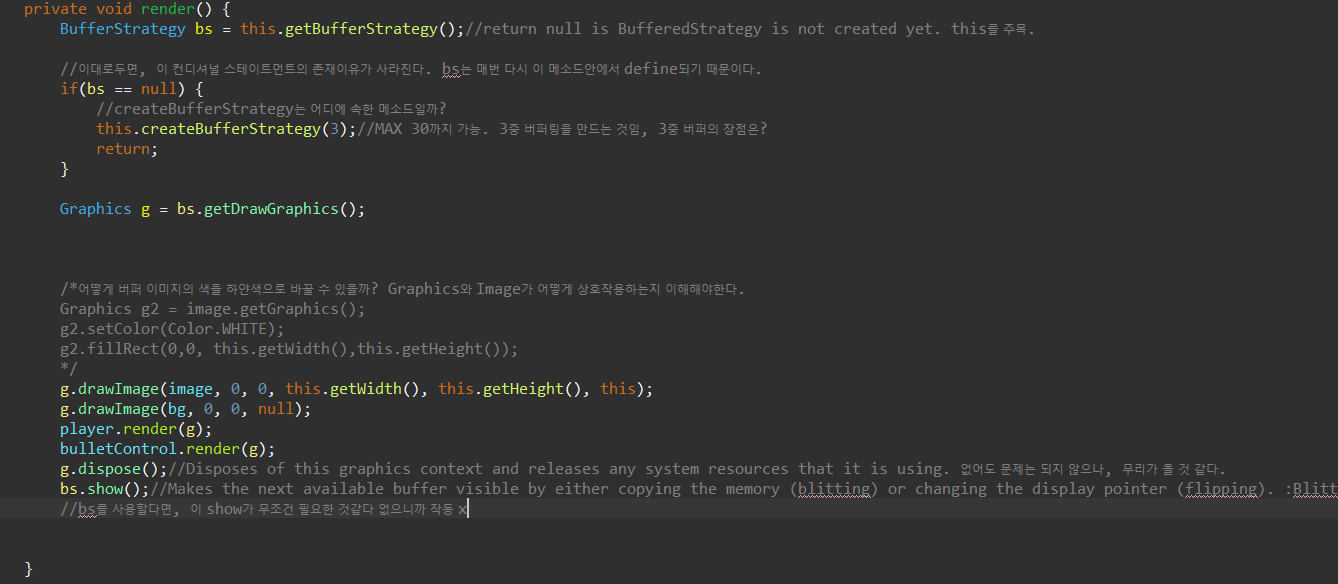
In the above example, we chose to synchronize the Sender object inside the run() method of the ThreadedSend class. Alternately, we could define the whole send() block as synchronized and it would produce the same result. Then we don’t have to synchronize the Message object inside the run() method in ThreadedSend class.



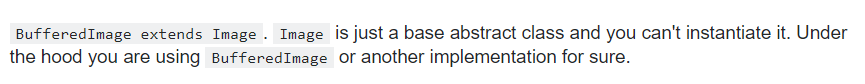
We do not always have to synchronize a whole method. Sometimes it is preferable to synchronize only part of a method. Java synchronized blocks inside methods makes this possible.



삼중버퍼링



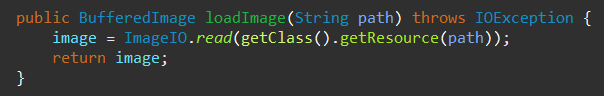
BufferedImage vs Image



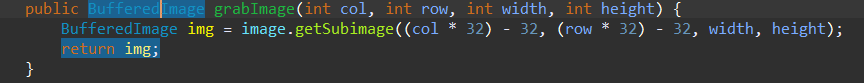
DynamicBeat에서는 Image class사용했는데?

내가 잘 모르는 클래스 & 메소드

BufferedImageLoader



SpriteSheet



Game



Canvas vs JFrame vs Jpanel

In java canvas is area used to draw something by java graphics. For ex. drawing an image or rectangle.

* Frame is used as JFrame(swing), a top level container which can contain canvas, panels, pane(DesktopPane, ScrollPane) etc..
* Panel or JPanel is a subcontainer used to contain textboxes, buttons, canvas etc.
* Jframe can contain multiple panels, but panel can't contain JFrame.

Textboxes, buttons can directly be added to Jframe but it decreases flexibility, Suppose we want to hide a set of buttons from ui, then we need to hide them one by one from JFrame. If those text boxes are added to panel then we just need to hide that panel only. There are so many cases about using panel in jframe.

Game class가 Runnable을 implement했다. 그렇다면 Thread object는 어디에서 만들어져야하고 왜 그곳에서 만들어져야하는가?

Thread는 게임이 실행될 때 필요한 것이다. Thread는 각각의 클래스가 자기 고유의 스레드를 갖고 돌게해준다. 즉, 이 스레드는 게임을 시작할 때에 들어가면 되는 것이다. 주의할 것은 run에서 만드는 것이 아니라, start 메소드에서 만들어져야한다는 것이다. Start 메소드는 run을 부르는 메소드이며, 런은 게임의 전체적인 흐름을 관리하지만, start는 게임의 시작을 의미하기에 조금 의미가 다르다.

Init 메소드는 필요한가?

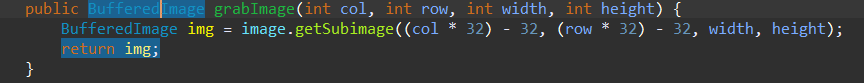


개인적인 생각으로는 constructor로 대체되어도 무방하다

로더와 SpriteSheet의 연계성?

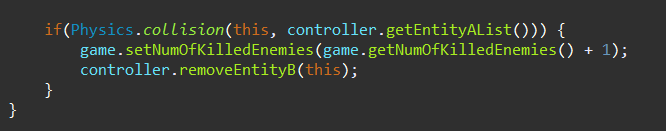
로더는 말그대로 reusability를 조금더 용이하게 만들어준다.

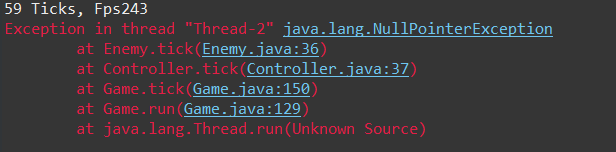
SpriteSheet은 하나의 이미지에서 부분적인 이미지를 갖고올수 있게 도와줌으로서 용량을 아낄 수 있다.



BufferStrategy

중요한 실수

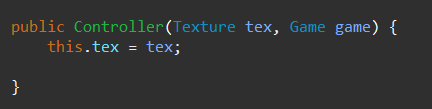


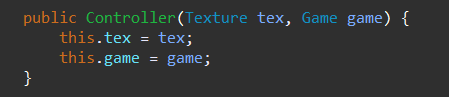


왜 에너미클래스 라인36에는 코드에 문제가 없었는데 NullPointerException이 나왔을까?

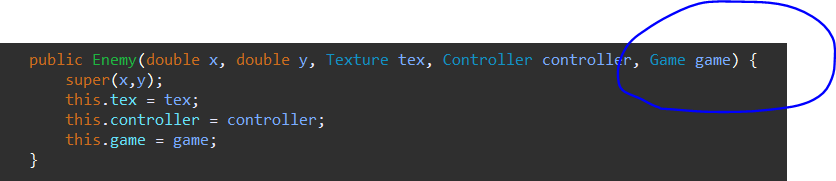
바로 연결Collection class에서 Enemy를 create할 때 Game class의 object를 넘기는데, Collection class에서 game object는 define만 되어있고, initialize되지 않았었다.

\*아무리, 버그가 어떠한 특정 라인을 가르키더라도, 그 클래스와 관련된 모든 것 그리고 관계를 살펴보아야한다.

<wrong>

<right>

두 클래스의 연결고리.



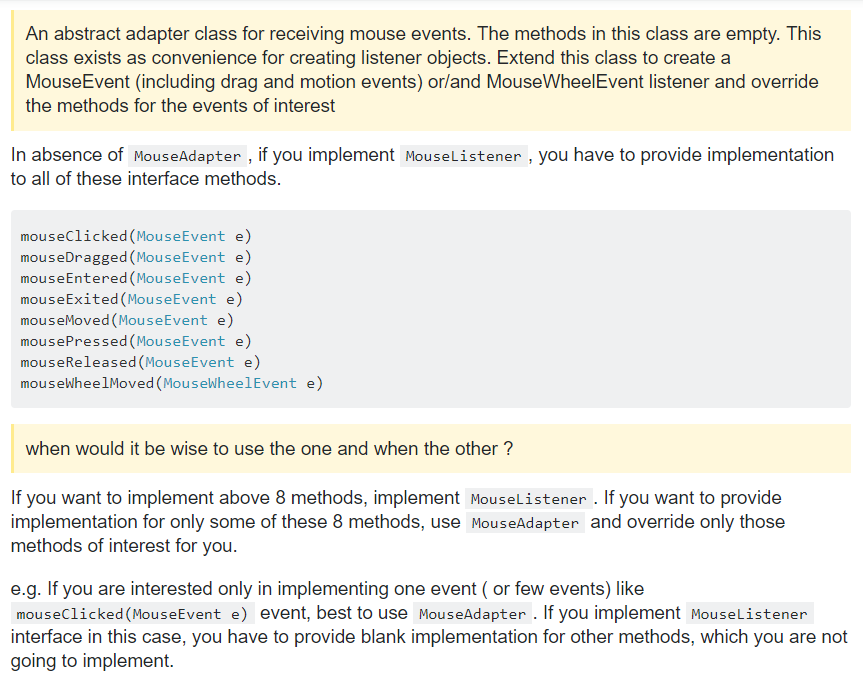
<Enemy.java>



<Controller.java>

What is difference between MouseAdapter and MouseListener

<https://stackoverflow.com/questions/44268617/difference-between-mouselistener-and-mouseadapter-in-java>



코드에서 이상한 부분

