Appendix - Watch

```
package src;

import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
import java.time.Clock;
import java.time.Instant;
import java.time.ZoneId;


public class Watch  {

	public long timePassed;

	private float ticks;

	private long previous;

	private int passedTicks;

	private float excessTicks;

	private int gameStart;

	private boolean isPaused;

	public Watch(float ticks){

		gameStart=0;
		setTicks(ticks);
		reset();


	}

	public void update(){


		long curr = getCurrentTime();
		float change = (float)(curr - previous)+excessTicks;

		if(!isPaused){
			this.passedTicks += (int)Math.floor(change / ticks);
			this.excessTicks = change%ticks;
		}
```

```
        this.previous = curr;
}

public void reset(){
        this.passedTicks =0;
        this.excessTicks = 0.0f;
        this.previous = getCurrentTime();
        timePassed = 0;
}

public void setTicks(float tickss){
        this.ticks = (1.0f/tickss)*1000;
}

public void setPaused(boolean paused){
        this.isPaused = paused;
}

public boolean isPaused(){
        return isPaused;
}
public boolean hasPassedTicks(){
        if(passedTicks > 0){
                this.passedTicks--;
                return true;
        }
        return false;
}

public boolean peekElapsedTick() {
        return (passedTicks > 0);
}

private static final long getCurrentTime(){
        return (System.nanoTime()/ 1000000L);
}

public int getSinceStart(){
        return gameStart;
}

public void addSinceStart(){
        gameStart++;
}
```

```
}
```