Class WarWorld

java.lang.Object
greenfoot.World
WarWorld

```
public class WarWorld
extends greenfoot.World
```

WarWorld Class

The year is 500 BCE. The Great Dragon War ended 3000 years ago, bringing peace to the Four Kingdoms. But now, corruption has brought that balance to shambles. The Four Kingdoms are now in full out war. Come together and witness this epic showdown!

Description:

WarWorld is where the simulation takes place. It includes four seperate corners, each home to a kingdom. Each kingdom starts off with a Chateau, two Treasuries and two Garrisons. Throughout the simulation, treasuries produce money and Garrisons produce troops. The money produced by the treasuries goes towards creating troops, and upgrading buildings. An upgraded treasury produces more money per cycle, and an upgraded garrison produces stronger troops. Their are two types of troops, the Dragon and the Serpentine. The Serpentine targets establishments, but may hit any of the troops in its way. It attacks through a hard-hitting cannonball, but takes significant time to reload. The Dragon, on the other hand, targets whatever is the closest, both estblishments and troops. It attacks through quickly fired, fireballs. Although the fireballs do little damage, the firerate allows for the Dragon to have a higher damage per act (DPA). During the simulation, the kingdoms will be divided into two alliances, blue or gold. Once an alliance is wiped out, the remaining kingdoms will be divided up into the same two alliance until there is a single kingdom standing, the winner. While the simulation takes place, there will be a timer displaying the time in minutes and seconds that has passed, along with scoreboards for each of the kingdoms, displaying their kingdom name, alliance, money, current troops, and current establishments.

NOTE: For more information about the simulation, consult the important elements section below.

Important elements:

- Four seperate soundtracks that play throughout the simulation for each of the stages. Consult the credits section below for their names and credits
- A starting screen that greets players with powerful music and attractive graphics
- A story screen explaining to the players the reasoning behind the conflict, with some more powerful music, allowing players to submerge themselves mentally to the simulation
- Control screen that allows players to choose the starting alliances and starting money for each kingdom
- Powerful music that plays during the simulation, keeping players engaged
- Exciting victory music that plays after the simulation, along with a victory screen that displays which kingdom won and the time it took

    - Serpentine targets only establishments, but may hit troops in its way, shoots cannonballs, moves slower, does less damage per act
    - Dragon shoots fireballs that do much less damage, but due to the high rate of fire, does more DPA
    - Serpentine shoots hard hitting fireballs that do much more damange, but due to the low rate of fire, does less DPA
    - Dragon also does limited splash damage, whereas serpentine is single target
- Three types of establishments, Chateau, Treasury, and Garrison
    - Chateau is the command centre of the kingdom and determines when to queue troops or upgrade
    - Treasury creates money to upgrade establishments and create troops
    - Garrison creates troops
- The Treasury and Garrison establishments are able to level up
    - Treasuries level up to produce more money each cycle (every 2 seconds)
    - Garrisons level up to produce troops with more HP and damage
- Two types of ammunition, Fireball and Cannonball
    - The cannonball does a significant amount of damange and is single target
    - A single fireball does little damange and does splash damage
- After buildings are destroyed, there is a **very cool explosion animation** that plays, made up of **25 different frames**
    - The animation was created by taking a transparent gif of an explosion and seperating it into seperate frames
    - These frames are played through, changing every 4 acts, giving the effect of a super clean animation
- Sound effects for the destruction of a building, cannonball firing, fireball firing, establishment upgrades, serpentines being destroyed, and dragon's losing all their health (We were going to add more, but it sounded too hectic)
- Individual scoreboards for each of the kingdoms, can be toggled on or off using the "Toggle Scoreboards" button at the top of the simulation
- A time keeper to keep track of the minutes and seconds that have passed since the simulation started
- A health bar for each of the troops and establishments to display their health
- A timer for the garrisons to display how much time is remaining before the next troops is created
- A disappearing skull that is displayed once a troop dies

Credits:

- Ashes on the Fire Soundtrack (plays during the WarWorld): Kohta Yamamoto
- Attack on Titan Soundtrack (plays during the StoryScreen): Hiroyuki Sawano
- Bird in a Cage Soundtrack (plays during the EndScreen): Hiroyuki Sawano
- Wings of Freedom Soundtrack (plays during the StartScreen): Hiroyuki Sawano
- Apple Seed Soundtrack (plays during the ControlScreen): Hiroyuki Sawano
- Clash of Clans Sound Effects: Supercell Oy
- Explosion sound effect: Renato Ottlakan
- Cannon image: imarvintpa
- Castle image: Serge Billault
- Dragon image: DraconomiconArt
- Parchment image: Wikimedia Commons
- Parchment assets: GameDev Market

- StatBar class and zeroAdder() method: Mr. Cohen
- SimpleTimer class: Neil Brown

Next Steps (Possible Improvements):

- Add more troop subclasses
- Add more establishment subclasses
- Add more user controls on the ControlScreen for the user to experiment with

Version:

April 2021

Author:

Andrew Qiao, Ryo Minakami, Brenden Chan, James Li

Field Summary

Fields

Modifier and Type	Field	Description
static double	ATTACK_RANGE	
static int	CANNONBALL_DAMAGE	
static int	CANNONBALL_SPEED	
static int[][]	CHATEAU_COORDINATES	
static int	CHATEAU_HP	
static int	DRAGON_COST	
static int	DRAGON_HP	
static int	DRAGON_RELOAD	
static int	DRAGON_SPEED	
static int	DRAGON_TIME	
static int	FIREBALL_DAMAGE	
static double	FIREBALL_RANGE	
static int	FIREBALL_SPEED	
static int[][]	GARRISON_COORDINATES	
static int	GARRISON_HP	
static int	GARRISON_LEVEL_2_COST	

static int	SERPENTINE_HP
static int	SERPENTINE_RELOAD
static int	SERPENTINE_SPEED
static int	SERPENTINE_TIME
static int[][]	TREASURY_COORDINATES
static int	TREASURY_HP
static int	TREASURY_LEVEL_2_COST
static int	TREASURY_LEVEL_3_COST

Constructor Summary

Constructors

Constructor	Description
WarWorld(int belzergAlliance, int leidenAlliance, int marleyAlliance, int eldiaAlliance, int belzergMoney, int leidenMoney, int marleyMoney, int eldiaMoney)	Constructor for WarWorld, where the simulation takes place, initializes each of the chateaus, as well as the objects that they contain.

Method Summary

All Methods Static Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
void	act()	Called every act, checks if any kingdoms have fallen and if so, changes the current alliances accordingly, also updates scoreboards and keeps track of if the user clicked the hide scoreboard button.
static ArrayList<Chateau>	getChateaus()	Returns the ArrayList of chateaus.

Methods inherited from class greenfoot.World

Methods inherited from class java.lang.Object

clone, equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Field Detail

SERPENTINE_COST

```
public static final int SERPENTINE_COST
```

See Also:

Constant Field Values

SERPENTINE_TIME

```
public static final int SERPENTINE_TIME
```

See Also:

Constant Field Values

SERPENTINE_HP

```
public static final int SERPENTINE_HP
```

See Also:

Constant Field Values

SERPENTINE_SPEED

```
public static final int SERPENTINE_SPEED
```

See Also:

Constant Field Values

```
public static final int SERPENTINE_RELOAD
```

See Also:

Constant Field Values

DRAGON_COST

```
public static final int DRAGON_COST
```

See Also:

Constant Field Values

DRAGON_TIME

```
public static final int DRAGON_TIME
```

See Also:

Constant Field Values

DRAGON_HP

```
public static final int DRAGON_HP
```

See Also:

Constant Field Values

DRAGON_SPEED

```
public static final int DRAGON_SPEED
```

See Also:

Constant Field Values

Constant Field Values

CANNONBALL_SPEED

```
public static final int CANNONBALL_SPEED
```

See Also:

Constant Field Values

CANNONBALL_DAMAGE

```
public static final int CANNONBALL_DAMAGE
```

See Also:

Constant Field Values

FIREBALL_SPEED

```
public static final int FIREBALL_SPEED
```

See Also:

Constant Field Values

FIREBALL_DAMAGE

```
public static final int FIREBALL_DAMAGE
```

See Also:

Constant Field Values

TREASURY_LEVEL_2_COST

TREASURY_LEVEL_3_COST

```
public static final int TREASURY_LEVEL_3_COST
```

See Also:

Constant Field Values

GARRISON_LEVEL_2_COST

```
public static final int GARRISON_LEVEL_2_COST
```

See Also:

Constant Field Values

GARRISON_LEVEL_3_COST

```
public static final int GARRISON_LEVEL_3_COST
```

See Also:

Constant Field Values

TREASURY_HP

```
public static final int TREASURY_HP
```

See Also:

Constant Field Values

GARRISON_HP

```
public static final int GARRISON_HP
```

See Also:

CHATEAU_HP

```
public static final int CHATEAU_HP
```

See Also:

Constant Field Values

CHATEAU_COORDINATES

```
public static final int[][] CHATEAU_COORDINATES
```

GARRISON_COORDINATES

```
public static final int[][] GARRISON_COORDINATES
```

TREASURY_COORDINATES

```
public static final int[][] TREASURY_COORDINATES
```

SCOREBOARD_COORDINATES

```
public static final int[][] SCOREBOARD_COORDINATES
```

ATTACK_RANGE

```
public static final double ATTACK_RANGE
```

See Also:

Constant Field Values

Constant Field Values

Constructor Detail

WarWorld

```
public WarWorld(int belzergAlliance,
                int leidenAlliance,
                int marleyAlliance,
                int eldiaAlliance,
                int belzergMoney,
                int leidenMoney,
                int marleyMoney,
                int eldiaMoney)
```

Constructor for WarWorld, where the simulation takes place, initializes each of the chateaus, as well as the objects that they contain. Also takes in parameters for the starting values of alliance and money for each kingdom, as decided by the user on the ControlScreen.

Parameters:

`belzergAlliance` - Starting alliance for Belzerg kingdom (0 - Gold, 1 - Blue)

`leidenAlliance` - Starting alliance for Leiden kingdom (0 - Gold, 1 - Blue)

`marleyAlliance` - Starting alliance for Marley kingdom (0 - Gold, 1 - Blue)

`eldiaAlliance` - Starting alliance for Eldia kingdom (0 - Gold, 1 - Blue)

`belzergMoney` - Starting money for Belzerg kingdom

`leidenMoney` - Starting money for Leiden kingdom

`marleyMoney` - Starting money for Marley kingdom

`eldiaMoney` - Starting money for Eldia kingdom

Method Detail

also updates scoreboards and keeps track of if the user clicked the hide scoreboard button.

Overrides:

`act` in class `greenfoot.World`

getChateaus

```
public static ArrayList<Chateau> getChateaus()
```

Returns the ArrayList of chateaus.

Returns:

ArrayList ArrayList of chateaus