Content

[Project package classes 2](#_Toc53076079)

[Lifeform 2](#_Toc53076080)

[As of: 02.10.2020 2](#_Toc53076081)

[Monster 5](#_Toc53076082)

[As of: 02.10.2020 5](#_Toc53076083)

[Player 6](#_Toc53076084)

[As of: 02.10.2020 8](#_Toc53076085)

[Project package game 9](#_Toc53076086)

[Battle 9](#_Toc53076087)

[As of: 08.10.2020 9](#_Toc53076088)

[As of: 05.10.2020 12](#_Toc53076089)

[As of: 04.10.2020: 14](#_Toc53076090)

[CreateChar 15](#_Toc53076091)

[As of: 03.10.2020 15](#_Toc53076092)

[Game 16](#_Toc53076093)

[As of: 07.10.2020 16](#_Toc53076094)

[As of: 04.10.2020 18](#_Toc53076095)

[As of: 03.10.2020 20](#_Toc53076096)

[Main 21](#_Toc53076097)

[As of: 06.10.2020 21](#_Toc53076098)

[As of: 04.10.2020 21](#_Toc53076099)

[As of: 03.10.2020 22](#_Toc53076100)

[As of: 02.10.2020 24](#_Toc53076101)

[Project package GUI 26](#_Toc53076102)

[GameWindow 26](#_Toc53076103)

[As of: 08.10.2020 26](#_Toc53076104)

[As of: 07.10.2020 29](#_Toc53076105)

[StartWindow 31](#_Toc53076106)

[As of: 03.10.2020 31](#_Toc53076107)

[As of: 02.10.2020 33](#_Toc53076108)

# Project package classes

## Lifeform

### As of: 02.10.2020

**package** classes;

**public** **class** Lifeform // Superclass

{

**private** **boolean** death = **false**;

**private** **boolean** tired = **false**;

**private** **int** maxLife;

**private** **int** life;

**private** **int** maxStamina;

**private** **int** stamina;

**private** **int** level;

**private** **int** experience;

**private** **int** damage;

// Constructors

**public** **void** setDeath(**boolean** death)

{

**this**.death = death;

}

**public** **boolean** getDeath()

{

**return** death;

}

**public** **void** setTired(**boolean** tired)

{

**this**.tired = tired;

}

**public** **boolean** getTired()

{

**return** tired;

}

**public** **void** setMaxLife(**int** maxLife)

{

**this**.maxLife = maxLife;

}

**public** **int** getMaxLife()

{

**return** maxLife;

}

**public** **void** setLife(**int** life)

{

**this**.life = life;

}

**public** **int** getLife()

{

**return** life;

}

**public** **int** getMaxStamina() {

**return** maxStamina;

}

**public** **void** setMaxStamina(**int** maxStamina) {

**this**.maxStamina = maxStamina;

}

**public** **int** getStamina() {

**return** stamina;

}

**public** **void** setStamina(**int** stamina) {

**this**.stamina = stamina;

}

**public** **int** getLevel() {

**return** level;

}

**public** **void** setLevel(**int** level) {

**this**.level = level;

}

**public** **int** getExperience() {

**return** experience;

}

**public** **void** setExperience(**int** experience) {

**this**.experience = experience;

}

**public** **int** getDamage() {

**return** damage;

}

**public** **void** setDamage(**int** damage) {

**this**.damage = damage;

}

// Methods

**public** **int** giveDamage()

{

**if**(stamina <= 0)

{

rest();

**return** 0;

} **else**

{

stamina -= 10;

**return** damage;

}

}

**public** **boolean** receiveDamage(**int** damage)

{

life -= damage;

tired = **true**;

**if**(life <= 0)

{

death = **true**;

life = 0;

stamina = 0;

}

**return** death;

}

**public** **boolean** rest()

{

**if**(life <= maxLife || stamina >= maxStamina)

{

life = maxLife; stamina = maxStamina;

tired = **false**;

System.***out***.println("Du hast dich ausgeruht und bist wieder fit. \n");

} **else**

{

System.***out***.println("Du bist ausgeruht");

}

**return** tired;

}

**public** **void** receiveExperience(**int** exp)

{

**this**.experience += exp;

}

}

## Monster

### As of: 02.10.2020

**package** classes;

**public** **class** Monster **extends** Lifeform // Subclass from Lifeform

{

**private** String type;

**public** **void** setType(String type)

{

**this**.type = type;

}

**public** String getType()

{

**return** type;

}

**public** **void** profile()

{

System.***out***.println("Type: " +type +" Level: " +getLevel());

System.***out***.printf("Life: %d Stamina: %d \n", getLife(), getStamina());

System.***out***.printf("Damage: %d Experience: %d \n", getDamage(), getExperience());

}

}

## Player

As of: 04.10.2020

**package** classes;

**public** **class** Player **extends** Lifeform //Subclass of Lifeform

{

**private** String name;

**private** **int** levelCap;

**public** **void** setName(String name)

{

**this**.name = name;

}

**public** String getName()

{

**return** name;

}

**public** **void** setLevelCap(**int** levelCap)

{

**this**.levelCap = levelCap;

}

**public** **int** getLevelCap()

{

**return** levelCap;

}

/\*

\* profile() is a method which will show the Information of the Player

\*/

**public** **void** profile()

{

**if**(getExperience() >= getLevelCap())

{

levelUp();

}

**else** **if**(getDeath() == **true**)

{

System.***out***.println("Status: Dead");

}

System.***out***.println("Name: " +name +" Level: " +getLevel());

System.***out***.printf("Life: %d(%d) Stamina: %d(%d) \n", getLife(), getMaxLife(), getStamina(), getMaxStamina());

System.***out***.printf("Damage: %d Experience: %d(%d) \n", getDamage(), getExperience(), getLevelCap());

}

//Method for leveling up

**public** **void** levelUp()

{

**do**

{

setLevel(getLevel() + 1);

setExperience(getExperience() - getLevelCap());

setLevelCap((**int**)(getLevelCap() \* 1.2));

setMaxLife((**int**)(getMaxLife() \* 1.2));

setMaxStamina((**int**)(getMaxStamina() \* 1.2));

setDamage((**int**)(getDamage() \* 1.2));

System.***out***.println("Glückwunsch, du bist im Level gestiegen\n");

}**while**(getExperience() >= getLevelCap());

}

}

### As of: 02.10.2020

**package** classes;

**public** **class** Player **extends** Lifeform //Subclass of Lifeform

{

**private** String name;

**public** **void** setName(String name)

{

**this**.name = name;

}

**public** String getName()

{

**return** name;

}

/\*

\* profile() is a method which will show the Information of the Player

\*/

**public** **void** profile()

{

**if**(getDeath() == **true**) System.***out***.println("Status: Dead");

System.***out***.println("Name: " +name +" Level: " +getLevel());

System.***out***.printf("Life: %d Stamina: %d \n", getLife(), getStamina());

System.***out***.printf("Damage: %d Experience: %d \n", getDamage(), getExperience());

}

}

# Project package game

## Battle

### As of: 08.10.2020

**package** game;

**import** javax.swing.JOptionPane;

**public** **class** Battle

{

**private** String text;

**public** String battle(classes.Player myPlayer)

{

classes.Monster monster01 = **new** classes.Monster();

monster01.setType("Boar");

monster01.setLevel(1);

monster01.setLife(50);

monster01.setStamina(30);

monster01.setDamage(5);

monster01.setExperience(10);

//monster01.profile();

System.***out***.println("\n");

/\*

\* fight between Player and Monster with Playeroptions

\*/

**boolean** getAway = **false**;

**boolean** goOn = **true**;

String selection = "";

**do**

{

selection = JOptionPane.*showInputDialog*("Wähle: \nAngriff: 1, Ausruhen: 2, Fliehen: 3");

**switch**(selection)

{

**case** "1": text = attack(myPlayer, monster01);

**return** text;

//break;

**case** "2": myPlayer.rest();

myPlayer.receiveDamage(monster01.getDamage());

System.***out***.println("Du wurdest vom gegnerischen Angriff getroffen. \nRestleben: " +myPlayer.getLife());

text = "Du wurdest vom gegnerischen Angriff getroffen. \nRestleben: " +myPlayer.getLife();

**return** text;

//break;

**case** "3": getAway = escape();

**if**(getAway == **true**)

{

goOn = **false**;

System.***out***.println("Flucht erfolgreich");

} **else**

{

myPlayer.receiveDamage(monster01.getDamage());

System.***out***.println("nicht erfolgreich \nDu wurdest vom gegnerischen Angriff getroffen. \nRestleben: " +myPlayer.getLife());

}

**break**;

**default**: System.***out***.println("Bitte nur zwischen Angegebenem wählen");

}

//Test if one is death

**if**(myPlayer.getDeath() == **true**)

{

goOn = **false**;

} **else** **if**(monster01.getDeath() == **true**)

{

System.***out***.printf("Du hast gewonnen und erhältst %d Erfahrungspunkte \n\n" ,monster01.getExperience());

goOn = **false**;

}

}**while**(goOn == **true**);

// Player still alive and dind't escape, then he gets experience

**if**(myPlayer.getDeath()==**false** && getAway ==**false**) myPlayer.receiveExperience(monster01.getExperience());

// change getAway to default!

getAway = **false**;

goOn = **true**;

//return text;

**return** text;

}

**public** String attack(classes.Player myPlayer, classes.Monster monster01)

{

monster01.receiveDamage(myPlayer.giveDamage());

**if**(monster01.getDeath() == **false**) myPlayer.receiveDamage(monster01.giveDamage());

System.***out***.println("Restleben Player: " +myPlayer.getLife() +"\nRestausdauer Player: " +myPlayer.getStamina()

+"\nRestleben Monster: " +monster01.getLife() +"\n");

String text = "Restleben Player: " +myPlayer.getLife() +"\nRestausdauer Player: " +myPlayer.getStamina()

+"\nRestleben Monster: " +monster01.getLife() +"\n";

**return** text;

}

**public** **boolean** escape()

{

**boolean** getAway = **false**;

// You have a chance of 66% to escape the battle

**int** chance = (**int**)Math.*round*((Math.*random*()\*99)+1);

**if**(chance >= 33) getAway = **true**;

**return** getAway;

}

}

### As of: 05.10.2020

**package** game;

**import** javax.swing.JOptionPane;

**public** **class** Battle

{

**public** **void** battle(classes.Player myPlayer)

{

classes.Monster monster01 = **new** classes.Monster();

monster01.setType("Boar");

monster01.setLevel(1);

monster01.setLife(50);

monster01.setStamina(30);

monster01.setDamage(5);

monster01.setExperience(10);

//monster01.profile();

System.***out***.println("\n");

/\*

\* fight between Player and Monster with Playeroptions

\*/

**boolean** getAway = **false**;

**boolean** goOn = **true**;

String selection = "";

**do**

{

selection = JOptionPane.*showInputDialog*("Wähle: \nAngriff: 1, Ausruhen: 2, Fliehen: 3");

**switch**(selection)

{

**case** "1": *attack*(myPlayer, monster01);

**break**;

**case** "2": myPlayer.rest();

myPlayer.receiveDamage(monster01.getDamage());

System.***out***.println("Du wurdest vom gegnerischen Angriff getroffen. \nRestleben: " +myPlayer.getLife());

**break**;

**case** "3": getAway = escape();

**if**(getAway == **true**)

{

goOn = **false**;

System.***out***.println("Flucht erfolgreich");

} **else**

{

myPlayer.receiveDamage(monster01.getDamage());

System.***out***.println("nicht erfolgreich \nDu wurdest vom gegnerischen Angriff getroffen. \nRestleben: " +myPlayer.getLife());

}

**break**;

**default**: System.***out***.println("Bitte nur zwischen Angegebenem wählen");

}

//Test if one is death

**if**(myPlayer.getDeath() == **true**)

{

goOn = **false**;

} **else** **if**(monster01.getDeath() == **true**) goOn = **false**;

}**while**(goOn == **true**);

// Player still alive and dind't escape, then he gets experience

**if**(myPlayer.getDeath()==**false** && getAway ==**false**) myPlayer.receiveExperience(monster01.getExperience());

// change getAway to default!

getAway = **false**;

goOn = **true**;

}

**public** **static** **void** attack(classes.Player myPlayer, classes.Monster monster01)

{

monster01.receiveDamage(myPlayer.giveDamage());

**if**(monster01.getDeath() == **false**) myPlayer.receiveDamage(monster01.giveDamage());

System.***out***.println("Restleben Player: " +myPlayer.getLife() +"\nRestausdauer Player: " +myPlayer.getStamina()

+"\nRestleben Monster: " +monster01.getLife() +"\n");

}

**public** **boolean** escape()

{

**boolean** getAway = **false**;

// You have a chance of 66% to escape the battle

**int** chance = (**int**)Math.*round*((Math.*random*()\*99)+1);

**if**(chance >= 33) getAway = **true**;

**return** getAway;

}

}

### As of: 04.10.2020:

**package** game;

**public** **class** Battle

{

**public** **void** battle(classes.Player myPlayer)

{

classes.Monster monster01 = **new** classes.Monster();

monster01.setType("Boar");

monster01.setLevel(1);

monster01.setLife(50);

monster01.setStamina(30);

monster01.setDamage(5);

monster01.setExperience(10);

//monster01.profile();

System.***out***.println("\n");

// fight between Player and Monster

**while**(monster01.getDeath() == **false** || myPlayer.getDeath())

{

*attack*(myPlayer, monster01);

}

// Player still alive, then he gets experience

**if**(myPlayer.getDeath()==**false**) myPlayer.receiveExperience(monster01.getExperience());

}

**public** **static** **void** attack(classes.Player myPlayer, classes.Monster monster01)

{

monster01.receiveDamage(myPlayer.giveDamage());

**if**(monster01.getDeath() == **false**) myPlayer.receiveDamage(monster01.giveDamage());

System.***out***.println("Restleben Player: " +myPlayer.getLife() +"\nRestausdauer Player: " +myPlayer.getStamina()

+"\nRestleben Monster: " +monster01.getLife() +"\n");

}

}

## CreateChar

As of: 04.10.2020

**package** game;

**public** **class** CreateChar

{

**public** **void** createChar(String name)

{

classes.Player myPlayer = **new** classes.Player();

myPlayer.setName(name);

myPlayer.setLife(100);

myPlayer.setMaxLife(myPlayer.getLife());

myPlayer.setStamina(100);

myPlayer.setMaxStamina(myPlayer.getStamina());

myPlayer.setLevel(1);

myPlayer.setLevelCap(50);

myPlayer.setExperience(0);

myPlayer.setDamage(20);

//myPlayer.profile();

Game game = **new** Game();

game.game(myPlayer);

}

}

### As of: 03.10.2020

**package** game;

**public** **class** CreateChar

{

**public** **void** createChar(String name)

{

classes.Player myPlayer = **new** classes.Player();

myPlayer.setName(name);

myPlayer.setLife(100);

myPlayer.setMaxLife(myPlayer.getLife());

myPlayer.setStamina(100);

myPlayer.setMaxStamina(myPlayer.getStamina());

myPlayer.setLevel(1);

myPlayer.setExperience(0);

myPlayer.setDamage(20);

//myPlayer.profile();

Game game = **new** Game();

game.game(myPlayer);

}

}

## Game

### As of: 07.10.2020

**package** game;

**import** javax.swing.JOptionPane;

**import** javax.swing.SwingUtilities;

**import** java.lang.Math;

**public** **class** Game

{

**public** **void** game(classes.Player myPlayer)

{

//boolean does the game continue

**boolean** loose = **false**;

/\*

\* Start GameWindow

\*/

GUI.GameWindow gameWindow = **new** GUI.GameWindow();

gameWindow.setVisible(**true**);

SwingUtilities.*invokeLater*(**new** Runnable()

{

@Override

**public** **void** run() {

**new** GUI.StartWindow();

}

});

//Test Übergabewert

myPlayer.profile();

String selection = "";

//The game, player can play as long as he isn't level 3 or higher or doesn't die

**do**

{

selection = JOptionPane.*showInputDialog*("Wähle: \nErforschen: 1, Ausruhen: 2, Profil: 3");

**switch**(selection)

{

**case** "1": System.***out***.println("Erforschen");

**int** dice = (**int**)Math.*round*((Math.*random*()\*5)+1);

**if**(dice <= 3)

{

System.***out***.println("Monsterangriff");

Battle battle = **new** Battle();

battle.battle(myPlayer);

} **else**

{

System.***out***.println("Kein Kontakt");

}

**break**;

**case** "2": System.***out***.println("Ausruhen");

myPlayer.rest();

**break**;

**case** "3": System.***out***.println("Profil");

myPlayer.profile();

**break**;

**default**: System.***out***.println("Bitte nur zwischen Angegebenem wählen");

}

**if**(myPlayer.getDeath() == **true**)

{

loose = **true**;

} **else** **if**(myPlayer.getLevel() >= 3) loose = **true**;

} **while**(loose == **false**);

System.***out***.println("Game Ends");

**if**(myPlayer.getDeath() == **false**)

{

System.***out***.println("You won!");

} **else** System.***out***.println("You lost!");

}

}

### As of: 04.10.2020

package game;

import javax.swing.JOptionPane;

import java.lang.Math;

public class Game

{

public void game(classes.Player myPlayer)

{

//boolean does the game continue

boolean loose = false;

//Test Übergabewert

myPlayer.profile();

String selection = "";

//The game, player can play as long as he isn't level 3 or higher or doesn't die

do

{

selection = JOptionPane.showInputDialog("Wähle: \nErforschen: 1, Ausruhen: 2, Profil: 3");

switch(selection)

{

case "1": System.out.println("Erforschen");

int dice = (int)Math.round((Math.random()\*5)+1);

if(dice <= 3)

{

System.out.println("Monsterangriff");

Battle battle = new Battle();

battle.battle(myPlayer);

} else

{

System.out.println("Kein Kontakt");

}

break;

case "2": System.out.println("Ausruhen");

myPlayer.rest();

break;

case "3": System.out.println("Profil");

myPlayer.profile();

break;

default: System.out.println("Bitte nur zwischen Angegebenem wählen");

}

if(myPlayer.getDeath() == true)

{

loose = true;

} else if(myPlayer.getLevel() >= 3) loose = true;

} while(loose == false);

System.out.println("Game Ends");

if(myPlayer.getDeath() == false)

{

System.out.println("You won!");

} else System.out.println("You lost!");

}

}

### As of: 03.10.2020

**package** game;

**import** javax.swing.JOptionPane;

**import** java.lang.Math;

**public** **class** Game

{

**public** **void** game(classes.Player myPlayer)

{

//Test Übergabewert

myPlayer.profile();

String selection = "";

//Test with for 5 times, later as do-while

**for**(**int** i = 0; i < 5; i++)

{

selection = JOptionPane.*showInputDialog*("Wähle: \nErforschen: 1, Ausruhen: 2, Profil: 3");

**switch**(selection)

{

**case** "1": System.***out***.println("Erforschen");

**int** dice = (**int**)Math.*round*((Math.*random*()\*5)+1);

**if**(dice <= 3)

{

System.***out***.println("Monsterangriff");

} **else**

{

System.***out***.println("Kein Kontakt");

}

**break**;

**case** "2": System.***out***.println("Ausruhen");

myPlayer.rest();

**break**;

**case** "3": System.***out***.println("Profil");

myPlayer.profile();

**break**;

**default**: System.***out***.println("Bitte nur zwischen Angegebenem wählen");

}

}

}

}

## Main

### As of: 06.10.2020

**package** game;

**import** javax.swing.SwingUtilities;

**public** **class** Main

{

**public** **static** **void** main(String[] args)

{

//Start the game with a GUI, this will lead to the other classes to play

GUI.StartWindow startWindow = **new** GUI.StartWindow();

startWindow.setVisible(**true**);

SwingUtilities.*invokeLater*(**new** Runnable() {

**public** **void** run() {

**new** GUI.StartWindow();

}

});

}

}

### As of: 04.10.2020

**package** game;

**public** **class** Main

{

**public** **static** **void** main(String[] args)

{

//Start the game with a GUI, this will lead to the other classes to play

GUI.StartWindow startWindow = **new** GUI.StartWindow();

startWindow.setVisible(**true**);

}

}

### As of: 03.10.2020

**package** game;

**public** **class** Main

{

**public** **static** **void** main(String[] args)

{

/\*

\* Create the Player from the Player Class as an Object as myPlayer

\* Give the Player the first player status

\*/

classes.Player myPlayer = **new** classes.Player();

myPlayer.setName("Schad");

myPlayer.setLife(100);

myPlayer.setMaxLife(myPlayer.getLife());

myPlayer.setStamina(100);

myPlayer.setMaxStamina(myPlayer.getStamina());

myPlayer.setLevel(1);

myPlayer.setExperience(0);

myPlayer.setDamage(20);

myPlayer.profile();

System.***out***.println("\n");

/\*

\* Testing monster massattack

\*/

**for**(**int** i = 0; i < 6; i++)

{

*monsterkontakt*(myPlayer);

**if**(myPlayer.getLife() < 50 || myPlayer.getStamina() < 20) myPlayer.rest();

}

/\*

\* Testing receiveDamage

\* and resting

\*/

myPlayer.receiveDamage(120);

System.***out***.println("\n");

myPlayer.profile();

myPlayer.rest();

System.***out***.println("\n");

//myPlayer.profile();

//Test my GUI

GUI.StartWindow startWindow = **new** GUI.StartWindow();

startWindow.setVisible(**true**);

/\*

//Test CreateChar

CreateChar myChar = new CreateChar();

myChar.createChar("Test");

\*/

}

/\*

\* Testing if i can get the monster status

\* Testing transfer of my class Player

\*/

**public** **static** **void** monsterkontakt(classes.Player myPlayer)

{

classes.Monster monster01 = **new** classes.Monster();

monster01.setType("Boar");

monster01.setLevel(1);

monster01.setLife(50);

monster01.setStamina(30);

monster01.setDamage(5);

monster01.setExperience(10);

//monster01.profile();

System.***out***.println("\n");

// fight between Player and Monster

**while**(monster01.getDeath() == **false** || myPlayer.getDeath())

{

*attack*(myPlayer, monster01);

}

// Player still alive, then he gets experience

**if**(myPlayer.getDeath()==**false**) myPlayer.receiveExperience(monster01.getExperience());

}

**public** **static** **void** attack(classes.Player myPlayer, classes.Monster monster01)

{

monster01.receiveDamage(myPlayer.giveDamage());

**if**(monster01.getDeath() == **false**) myPlayer.receiveDamage(monster01.giveDamage());

System.***out***.println("Restleben Player: " +myPlayer.getLife() +"\nRestausdauer Player: " +myPlayer.getStamina()

+"\nRestleben Monster: " +monster01.getLife() +"\n");

}

}

### As of: 02.10.2020

**package** game;

**public** **class** Main

{

**public** **static** **void** main(String[] args)

{

/\*

\* Create the Player from the Player Class as an Object as myPlayer

\* Give the Player the first player status

\*/

classes.Player myPlayer = **new** classes.Player();

myPlayer.setName("Schad");

myPlayer.setLife(100);

myPlayer.setMaxLife(myPlayer.getLife());

myPlayer.setStamina(100);

myPlayer.setMaxStamina(myPlayer.getStamina());

myPlayer.setLevel(1);

myPlayer.setExperience(0);

myPlayer.setDamage(20);

myPlayer.profile();

System.***out***.println("\n");

/\*

\* Testing monster massattack

\*/

**for**(**int** i = 0; i < 6; i++)

{

*monsterkontakt*(myPlayer);

**if**(myPlayer.getLife() < 50 || myPlayer.getStamina() < 20) myPlayer.rest();

}

/\*

\* Testing receiveDamage

\* and resting

\*/

myPlayer.receiveDamage(120);

System.***out***.println("\n");

myPlayer.profile();

myPlayer.rest();

System.***out***.println("\n");

//myPlayer.profile();

//Test my GUI

GUI.StartWindow startWindow = **new** GUI.StartWindow();

startWindow.setVisible(**true**);

}

/\*

\* Testing if i can get the monster status

\* Testing transfer of my class Player

\*/

**public** **static** **void** monsterkontakt(classes.Player myPlayer)

{

classes.Monster monster01 = **new** classes.Monster();

monster01.setType("Boar");

monster01.setLevel(1);

monster01.setLife(50);

monster01.setStamina(30);

monster01.setDamage(5);

monster01.setExperience(10);

//monster01.profile();

System.***out***.println("\n");

// fight between Player and Monster

**while**(monster01.getDeath() == **false** || myPlayer.getDeath())

{

*attack*(myPlayer, monster01);

}

// Player still alive, then he gets experience

**if**(myPlayer.getDeath()==**false**) myPlayer.receiveExperience(monster01.getExperience());

}

**public** **static** **void** attack(classes.Player myPlayer, classes.Monster monster01)

{

monster01.receiveDamage(myPlayer.giveDamage());

**if**(monster01.getDeath() == **false**) myPlayer.receiveDamage(monster01.giveDamage());

System.***out***.println("Restleben Player: " +myPlayer.getLife() +"\nRestausdauer Player: " +myPlayer.getStamina()

+"\nRestleben Monster: " +monster01.getLife() +"\n");

}

}

# Project package GUI

## GameWindow

### As of: 08.10.2020

**package** GUI;

**import** javax.swing.\*;

**import** java.awt.event.ActionListener;

**import** java.awt.CardLayout;

**import** java.awt.Color;

**import** java.awt.event.ActionEvent;

@SuppressWarnings("serial")

**public** **class** GameWindow **extends** JFrame //implements ActionListener

{

/\*

\* Implement all the Objects I need to create my User Interface

\*/

**private** JPanel framePanel;

**private** JPanel cardPanel;

**private** JPanel worldPanel;

**private** JPanel battlePanel;

**private** JLabel text;

**private** CardLayout wbCardLayout;

**private** JButton explore;

**private** JButton restWorld;

**private** JButton profil;

**private** JButton attack;

**private** JButton restBattle;

**private** JButton escape;

//private String text;

**private** **int** currentCard = 1;

**public** GameWindow(classes.Player myPlayer)

{

//Shut down program, while pressing the X in the window

**this**.setDefaultCloseOperation(WindowConstants.***EXIT\_ON\_CLOSE***);

//set settings for the Window (Frame)

**this**.setTitle("Growth");

**this**.setSize(640, 400); //1280, 800

**this**.setLocationRelativeTo(**null**);

//set Settings for the framePanel

framePanel = **new** JPanel();

framePanel.setLayout(**new** java.awt.BorderLayout());

text = **new** JLabel("Start");

text.setBackground(Color.***white***);

text.setOpaque(**true**);

cardPanel = **new** JPanel();

cardPanel.setSize(1200, 180);

/\*

\* Settings for the CardLayout

\*/

wbCardLayout = **new** CardLayout();

cardPanel.setLayout(wbCardLayout);

worldPanel = **new** JPanel();

battlePanel = **new** JPanel();

//Add Buttons to worldPanel

explore = **new** JButton("Erkunden");

restWorld = **new** JButton("Ausruhen");

profil = **new** JButton("Profil");

worldPanel.add(explore);

worldPanel.add(restWorld);

worldPanel.add(profil);

//Add Buttons to battlePanel

attack = **new** JButton("Angriff");

restBattle = **new** JButton("Ausruhen");

escape = **new** JButton("Fliehen");

battlePanel.add(attack);

battlePanel.add(restBattle);

battlePanel.add(escape);

//Add the two Panels to the cardPanel

cardPanel.add(worldPanel, "1");

cardPanel.add(battlePanel, "2");

/\*

\* ActionListener for the Buttons

\*/

explore.addActionListener(**new** ActionListener()

{

@Override

**public** **void** actionPerformed(ActionEvent aeExplore)

{

game.Battle battle = **new** game.Battle();

text.setText(battle.battle(myPlayer));

System.***out***.println(text);

}

});

/\*

\* For testing Reasons: Adding a Button to switch cards

\*/

JPanel buttonPanel = **new** JPanel();

JButton button = **new** JButton("Karten wechseln");

buttonPanel.add(button);

button.addActionListener(**new** ActionListener()

{

@Override

**public** **void** actionPerformed(ActionEvent e)

{

**if**(currentCard == 1)

{

currentCard++;

} **else** **if**(currentCard == 2) currentCard = 1;

wbCardLayout.show(cardPanel, "" +(currentCard));

}

});

//Add components to framePanel

framePanel.add(buttonPanel, java.awt.BorderLayout.***PAGE\_START***);

framePanel.add(text, java.awt.BorderLayout.***CENTER***);

framePanel.add(cardPanel, java.awt.BorderLayout.***PAGE\_END***);

**this**.add(framePanel);

}

}

### As of: 07.10.2020

**package** GUI;

**import** javax.swing.\*;

**import** java.awt.event.ActionListener;

**import** java.awt.CardLayout;

**import** java.awt.event.ActionEvent;

**public** **class** GameWindow **extends** JFrame //implements ActionListener

{

/\*

\* Implement all the Objects I need to create my User Interface

\*/

**private** JPanel framePanel;

**private** JPanel textPanel;

**private** JPanel cardPanel;

**private** JPanel worldPanel;

**private** JPanel battlePanel;

**private** JTextPane textPane;

**private** CardLayout wbCardLayout;

**private** JButton explore;

**private** JButton restWorld;

**private** JButton profil;

**private** JButton attack;

**private** JButton restBattle;

**private** JButton escape;

**private** **int** currentCard = 1;

**public** GameWindow()

{

//Shut down program, while pressing the X in the window

**this**.setDefaultCloseOperation(WindowConstants.***EXIT\_ON\_CLOSE***);

//set settings for the Window (Frame)

**this**.setTitle("Growth");

**this**.setSize(1280, 800);

**this**.setLocationRelativeTo(**null**);

//set Settings for the framePanel

framePanel = **new** JPanel();

framePanel.setLayout(**new** java.awt.BorderLayout());

textPane = **new** JTextPane();

textPane.setSize(1200, 600);

cardPanel = **new** JPanel();

cardPanel.setSize(1200, 180);

/\*

\* Settings for the CardLayout

\*/

wbCardLayout = **new** CardLayout();

cardPanel.setLayout(wbCardLayout);

worldPanel = **new** JPanel();

battlePanel = **new** JPanel();

//Add Buttons to worldPanel

explore = **new** JButton("Erkunden");

restWorld = **new** JButton("Ausruhen");

profil = **new** JButton("Profil");

worldPanel.add(explore);

worldPanel.add(restWorld);

worldPanel.add(profil);

//Add Buttons to battlePanel

attack = **new** JButton("Angriff");

restBattle = **new** JButton("Ausruhen");

escape = **new** JButton("Fliehen");

battlePanel.add(attack);

battlePanel.add(restBattle);

battlePanel.add(escape);

//Add the two Panels to the cardPanel

cardPanel.add(worldPanel, "1");

cardPanel.add(battlePanel, "2");

/\*

\* For testing Reasons: Adding a Button to switch cards

\*/

JPanel buttonPanel = **new** JPanel();

JButton button = **new** JButton("Karten wechseln");

buttonPanel.add(button);

button.addActionListener(**new** ActionListener()

{

@Override

**public** **void** actionPerformed(ActionEvent e)

{

**if**(currentCard == 1)

{

currentCard++;

} **else** **if**(currentCard == 2) currentCard = 1;

wbCardLayout.show(cardPanel, "" +(currentCard));

}

});

//Add components to framePanel

framePanel.add(buttonPanel, java.awt.BorderLayout.***PAGE\_START***);

framePanel.add(textPane, java.awt.BorderLayout.***CENTER***);

framePanel.add(cardPanel, java.awt.BorderLayout.***PAGE\_END***);

**this**.add(framePanel);

}

}

## StartWindow

### As of: 03.10.2020

**package** GUI;

**import** javax.swing.\*;

**import** game.CreateChar;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.ActionListener;

**public** **class** StartWindow **extends** JFrame **implements** ActionListener

{

/\*

\* Implement the different Objects for the Window

\* cn = character name; tcn = text character name;

\*/

JTextField cn = **new** JTextField("");

JLabel title = **new** JLabel("Growth");

JLabel tcn = **new** JLabel("Character name: ");

JButton agree = **new** JButton("OK");

JPanel panel = **new** JPanel();

//This is for the question if a charakter name was set

**boolean** text = **false**;

**public** StartWindow()

{

//Shut down the program if you press the X on the top-right side

setDefaultCloseOperation(WindowConstants.***EXIT\_ON\_CLOSE***);

**this**.setTitle("Growth");

//this.setSize(400,500);

//set the Border Layout to the panel

panel.setLayout(**new** java.awt.BorderLayout());

//Button agree & TextField cn to listener

agree.addActionListener(**this**);

cn.addActionListener(**this**);

//Add all those components together

panel.add(title, java.awt.BorderLayout.***PAGE\_START***);

panel.add(agree, java.awt.BorderLayout.***PAGE\_END***);

panel.add(tcn, java.awt.BorderLayout.***LINE\_START***);

panel.add(cn, java.awt.BorderLayout.***CENTER***);

**this**.getContentPane().add(panel);

//Reduce the window to the smallest possible dimensions

pack();

}

**public** **void** actionPerformed(ActionEvent ae)

{

**if**(ae.getSource() == **this**.cn)

{

text = **true**;

//System.out.println("Test erfolgreich");

}**else** **if**(ae.getSource() == **this**.agree && text == **true**)

{

//System.out.println("Test 2 erfolgreich");

text = **false**;

setVisible(**false**);

CreateChar myChar = **new** CreateChar();

myChar.createChar(cn.getText());

}**else** **if**(ae.getSource() == **this**.agree)

{

JOptionPane.*showMessageDialog*(**null**, "Please enter a character name, press Enter \nthen the OK button");

}

}

}

### As of: 02.10.2020

package GUI;

import javax.swing.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class StartWindow extends JFrame implements ActionListener

{

/\*

\* Implement the different Objects for the Window

\* cn = character name; tcn = text character name;

\*/

JTextField cn;

JLabel title;

JLabel tcn;

JButton agree;

JPanel panel;

public StartWindow()

{

this.setTitle("Growth");

this.setSize(400,500);

panel = new JPanel();

//JLabel for Title

title = new JLabel("Growth");

title.setBounds(150,20,100,40);

//JLabel for lead (tcn)

tcn = new JLabel();

tcn.setBounds(30,100,100,40);

//JTextField for writing character name (cn)

cn = new JTextField();

cn.setBounds(130,100,100,40);

//Button to Set character name (agree)

agree = new JButton("OK");

//Button agree to listener

agree.addActionListener(this);

agree.setBounds(130,160,100,40);

//Add all those components together

panel.add(agree);

panel.add(title);

panel.add(tcn);

panel.add(cn);

this.add(panel);

}

public void actionPerformed(ActionEvent ae)

{

if(ae.getSource() == this.agree)

{

title.setText("Test erfolgreich");

}

}

}