

Skład projektu: Natalia Chwał ← Lider

Język: Java

Temat: Łąka

Opis:

Program służy do symulowania życia zwierząt na łące oraz występującej na niej roślinności. Tworzona jest plansza o określonej wielkości, a następnie losowo są dodawane obiekty na planszę. Możliwa jest opcja manipulowania wartościami pewnych zmiennych. Ostatecznie, program pozwala na obserwowanie życia na łące w czasie rzeczywistym i interakcje z planszą. Każdy obiekt posiada odmienne własności i wzorce zachowań. Zwierzęta mogą poruszać się (w sposób losowy) po planszy o zadanych wymiarach. Każde z pól planszy może kryć jakiegoś rodzaju zasób (Roślinność). Symulację rozpoczynamy od utworzenia losowej planszy z zasobami o zadanych wymiarach. Następnie rozmieszczamy na niej losowo wygenerowane Zwierzęta. W każdym z kroków symulacji organizm przemieszcza się o specyficzną dla niego liczbę pól. W trakcie swojej wędrówki może znajdować zasoby, spotykać inne organizmy oraz wchodzić z nimi w interakcje.

1. Klasa „Zwierzę” - zawiera takie atrybuty:
 - 1.1. nazwę zwierzęcia (nazwa),
 - 1.2. symbol, który reprezentuje zwierzę na planszy (symbol),
 - 1.3. wartość życia, która określa, jak wiele punktów życia ma dany obiekt (życie)
2. Klasa „Mięsożerca” – klasa ta dziedziczy po klasie „Zwierzę”
 - 2.1. zawiera takie dodatkowe pole jak ilość zadawanych obrażeń (obrażenia).
 - 2.2. Ilość ofiar (ofiary)
 - 2.3. Głód (Głód);
3. Klasa „Roślinożerca” – klasa ta dziedziczy po klasie „Zwierzę”,
 - 3.1. zawiera takie dodatkowe pole jak ilość zjedzonego pokarmu (spożytyPokarm)
 - 3.2. Głód (Głód);
4. Klasa „Wszystkożerca” – klasa ta dziedziczy po klasie „Zwierzę”,
 - 4.1. zawiera takie dodatkowe pole jak ilość zjedzonego pokarmu (spożytyPokarm)
 - 4.2. Głód (Głód);
 - 4.3. zawiera takie dodatkowe pole jak ilość zadawanych obrażeń (obrażenia).
 - 4.4. Ilość ofiar (ofiary)
5. Klasa „Roślina” – zawiera takie atrybuty jak:
 - 5.1. nazwę rośliny (nazwa)
 - 5.2. symbol, który reprezentuje na planszy (symbol),
 - 5.3. Jak szybko rośnie (wzrost)
6. Klasa „Trująca” – klasa ta dziedziczy po klasie „Roślinność”,
 - 6.1. zawiera takie dodatkowe pole jak ilość zadawanych obrażeń z powodu trucizny (trucizna)
 - 6.2. sytość (sytość)
 - 6.3. Maksymalny wiek rośliny (maxWiek)
 - 6.4. Aktualny wiek (wiek)

7. Klasa „Jadalna” – klasa ta dziedziczy po klasie „Roślinność”,
 - 7.1. zawiera takie dodatkowe pole jak ilość zregenerowanych punktów zdrowia po spożyciu(Regeneracja).
 - 7.2. sytość (sytość)
 - 7.3. Maksymalny wiek rośliny (maxWiek)
 - 7.4. Aktualny wiek (wiek)

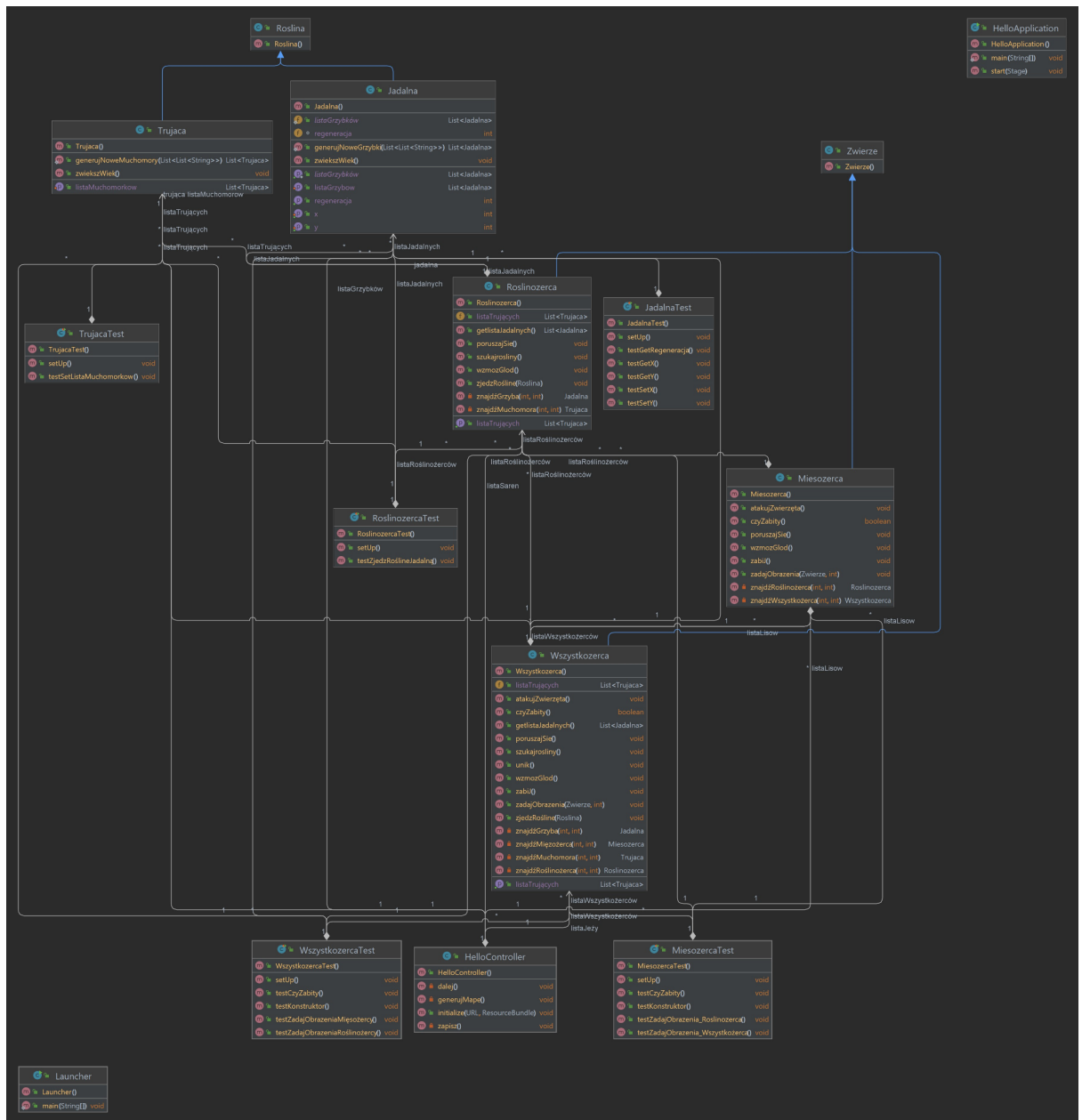
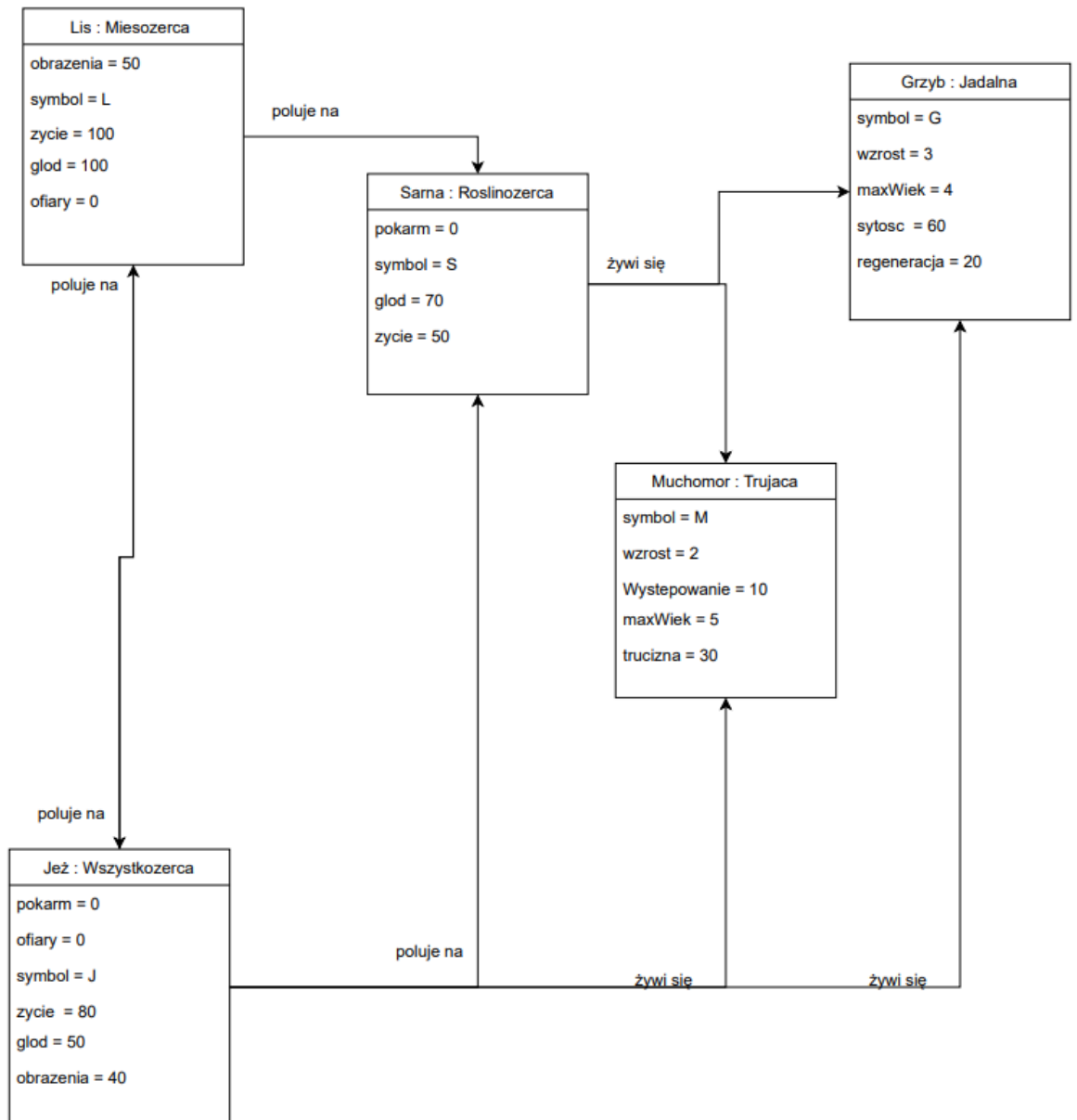


Diagram obiektów:



Package com.example.meadowfx

package com.example.meadowfx

Classes	
Class	Description
HelloApplication	
HelloController	
Jadalna	
Launcher	
Miesozerca	
Roslina	
Roslinozerca	
Trujaca	
Wszystkozerca	
Zwierze	

Package `com.example.meadowfx`

Class HelloApplication

`java.lang.Object`
`javafx.application.Application`
`com.example.meadowfx>HelloApplication`

`public class HelloApplication`
`extends javafx.application.Application`

Nested Class Summary

Nested classes/interfaces inherited from class `javafx.application.Application`

`javafx.application.Application.Parameters`

Field Summary

Fields inherited from class `javafx.application.Application`

`STYLE SHEET_CASPIAN`, `STYLE SHEET_MODENA`

Constructor Summary

Constructors

Constructor	Description
<code>HelloApplication()</code>	

Method Summary

All Methods **Static Methods** **Instance Methods** **Concrete Methods**

Modifier and Type	Method	Description
<code>static void</code>	<code>main(String [] args)</code>	
<code>void</code>	<code>start(javafx.stage.Stage stage)</code>	

Methods inherited from class `javafx.application.Application`

`getHostServices`, `getParameters`, `getUserAgentStylesheet`, `init`, `launch`, `launch`, `notifyPreloader`, `setUserAgentStylesheet`, `stop`

Methods inherited from class `java.lang.Object`

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

Constructor Details**HelloApplication**

```
public HelloApplication()
```

Method Details**start**

```
public void start(javafx.stage.Stage stage)  
    throws IOException
```

Specified by:

start in class `javafx.application.Application`

Throws:

`IOException`

main

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

Package `com.example.meadowfx`

Class HelloController

`java.lang.Object`
`com.example.meadowfx>HelloController`

All Implemented Interfaces:
`javafx.fxml.Initializable`

```
public class HelloController
extends Object
implements javafx.fxml.Initializable
```

Field Summary

Fields		
Modifier and Type	Field	Description
private <code>javafx.scene.text.Text</code>	<code>aktualneGrzyby</code>	
private <code>javafx.scene.text.Text</code>	<code>aktualneJeze</code>	
private <code>javafx.scene.text.Text</code>	<code>aktualneLisy</code>	
private <code>javafx.scene.text.Text</code>	<code>aktualneMuchomory</code>	
private <code>javafx.scene.text.Text</code>	<code>aktualneSarny</code>	
private <code>javafx.scene.text.Text</code>	<code>aktualnie</code>	
private <code>javafx.scene.control.Button</code>	<code>dalej</code>	
private <code>javafx.scene.text.Text</code>	<code>etap</code>	
(package private) <code>Random</code>	<code>generator</code>	
private <code>javafx.scene.control.Button</code>	<code>generuj</code>	
private <code>javafx.scene.text.Text</code>	<code>igreki</code>	
private <code>javafx.scene.text.Text</code>	<code>iksy</code>	
(package private) <code>int</code>	<code>iloscEtapow</code>	
private <code>javafx.scene.control.TextField</code>	<code>iloscJadalne</code>	
private <code>javafx.scene.control.TextField</code>	<code>iloscMiesozercy</code>	
private <code>javafx.scene.control.TextField</code>	<code>iloscRoslinozercy</code>	
private <code>javafx.scene.control.TextField</code>	<code>iloscTrujace</code>	
private <code>javafx.scene.control.TextField</code>	<code>iloscWszystkozercy</code>	
private <code>javafx.scene.text.Text</code>	<code>liczba</code>	
(package private) <code>List <Jadalna></code>	<code>listaGrzybków</code>	

```
(package private) List <Wszystkozerca> listaJezy
(package private) List <Miesozerca> listaLisow
(package private) List <Trujaca> listaMuchomorow
(package private) List <Roslinozerca> listaSaren
(package private) List <List <String >> Mapa
private javafx.scene.text.Text mapa1
private javafx.scene.text.TextFlow mapka
private javafx.scene.text.Text poczatkowo
private javafx.scene.text.Text populacjaMiesozercow
private javafx.scene.text.Text populacjaRoslinozercow
private javafx.scene.text.Text populacjaWszystkozercow
private javafx.scene.text.Text wystepowanieJadalne
private javafx.scene.text.Text wystepowanieTrujace
private javafx.scene.control.TextField xwprowadzone
private javafx.scene.control.TextField ywprowadzone
private javafx.scene.control.Button zapisz
```

Constructor Summary

Constructors

Constructor	Description
HelloController()	

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
private void	dalej()	
private void	generujMape()	
void	initialize(URL url, ResourceBundle resourceBundle)	
private void	zapisz()	

Methods inherited from class java.lang.Object


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

Field Details

iloscEtapow

```
int iloscEtapow
```

generator

```
Random generator
```

listaGrzybków

```
List <Jadalna> listaGrzybków
```

listaMuchomorow

```
List <Trujaca> listaMuchomorow
```

listaSaren

```
List <Roslinozerca> listaSaren
```

listaLisow

```
List <Miesozerca> listaLisow
```

listaJeży

```
List <Wszystkozerca> listaJeży
```

Mapa

```
List <List <String >> Mapa
```

mapka

```
private javafx.scene.text.TextFlow mapa
```

xwprowadzone

```
private javafx.scene.control.TextField xwprowadzone
```

ywprowadzone

```
private javafx.scene.control.TextField ywprowadzone
```

iloscJadalne

```
private javafx.scene.control.TextField iloscJadalne
```

iloscTrujace

```
private javafx.scene.control.TextField iloscTrujace
```

iloscWszystkozercy

```
private javafx.scene.control.TextField iloscWszystkozercy
```

iloscMiesozercy

```
private javafx.scene.control.TextField iloscMiesozercy
```

iloscRoslinozercy

```
private javafx.scene.control.TextField iloscRoslinozercy
```

generuj

```
private javafx.scene.control.Button generuj
```

dalej

```
private javafx.scene.control.Button dalej
```

zapisz

```
private javafx.scene.control.Button zapisz
```

iksy

```
private javafx.scene.text.Text iksy
```

igreki

```
private javafx.scene.text.Text igreki
```

mapa1

```
private javafx.scene.text.Text mapa1
```

populacjaMiesozercow

```
private javafx.scene.text.Text populacjaMiesozercow
```

populacjaRoslinozercow

```
private javafx.scene.text.Text populacjaRoslinozercow
```

populacjaWszystkozercow

```
private javafx.scene.text.Text populacjaWszystkozercow
```

wystepowanieJadalne

```
private javafx.scene.text.Text wystepowanieJadalne
```

wystepowanieTrujace

```
private javafx.scene.text.Text wystepowanieTrujace
```

etap

```
private javafx.scene.text.Text etap
```

liczba

```
private javafx.scene.text.Text liczba
```

początkowo

```
private javafx.scene.text.Text początkowo
```

aktualnie

```
private javafx.scene.text.Text aktualnie
```

aktualneLisy

```
private javafx.scene.text.Text aktualneLisy
```

aktualneSarny

```
private javafx.scene.text.Text aktualneSarny
```

aktualneJeze

```
private javafx.scene.text.Text aktualneJeze
```

aktualneGrzyby

```
private javafx.scene.text.Text aktualneGrzyby
```

aktualneMuchomory

```
private javafx.scene.text.Text aktualneMuchomory
```

Constructor Details

HelloController

```
public HelloController()
```

Method Details

generujMape

```
private void generujMape()
```

dalej

```
private void dalej()
```

zapisz

```
private void zapisz()
```

initialize

```
public void initialize(URL url,  
                       ResourceBundle resourceBundle)
```

Specified by:

initialize in interface `javafx.fxml.Initializable`

Package `com.example.meadowfx`

Class Jadalna

`java.lang.Object`
`com.example.meadowfx.Roslina`
`com.example.meadowfx.Jadalna`

`public class Jadalna`
`extends Roslina`

Field Summary

Fields

Modifier and Type	Field	Description
static <code>List</code> < <code>Jadalna</code> >	<code>listaGrzybków</code>	
<code>List</code> < <code>List</code> < <code>String</code> >>	<code>Mapa</code>	
(package private) <code>int</code>	<code>Maxwiek</code>	
(package private) <code>int</code>	<code>regeneracja</code>	
(package private) <code>int</code>	<code>sytosc</code>	
(package private) <code>int</code>	<code>wiek</code>	

Fields inherited from class `com.example.meadowfx.Roslina`

`nazwa`, `symbol`, `wzrost`, `x`, `y`

Constructor Summary

Constructors

Constructor	Description
<code>Jadalna()</code>	

Method Summary

All Methods **Static Methods** **Instance Methods** **Concrete Methods**

Modifier and Type	Method	Description
static <code>List</code> < <code>Jadalna</code> >	<code>generujNoweGrzybki(List <List <String</code> <code>>> mapa)</code>	
static <code>List</code> < <code>Jadalna</code> >	<code>getListaGrzybków()</code>	

int	<code>getRegeneracja()</code>
int	<code>getX()</code>
int	<code>getY()</code>
void	<code>setListaGrzybow(List <Jadalna> listaGrzybka)</code>
void	<code>setX(int wartosc)</code>
void	<code>setY(int wartosc)</code>
void	<code>zwiekszWiek()</code>

Methods inherited from class `java.lang.Object`

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

Field Details

regeneracja

int regeneracja

sytosc

int sytosc

wiek

int wiek

Maxwiek

int Maxwiek

Mapa

public List <List <String >> Mapa

listaGrzybków

```
public static List <Jadalna> listaGrzybków
```

Constructor Details

Jadalna

```
public Jadalna()
```

Method Details

getListaGrzybków

```
public static List <Jadalna> getListaGrzybków()
```

zwiększWiek

```
public void zwiększWiek()
```

setListaGrzybow

```
public void setListaGrzybow(List <Jadalna> listaGrzybka)
```

generujNoweGrzybki

```
public static List <Jadalna> generujNoweGrzybki(List <List <String >> mapa)
```

getX

```
public int getX()
```

getY

```
public int getY()
```

getRegeneracja

```
public int getRegeneracja()
```


setX

```
public void setX(int wartosc)
```

setY

```
public void setY(int wartosc)
```

Package `com.example.meadowfx`

Class Launcher

`java.lang.Object`
`com.example.meadowfx.Launcher`

`public class Launcher`
`extends Object`

Constructor Summary

Constructors

Constructor	Description
<code>Launcher()</code>	

Method Summary

All Methods **Static Methods** **Concrete Methods**

Modifier and Type	Method	Description
<code>static void</code>	<code>main(String [] args)</code>	

Methods inherited from class `java.lang.Object`

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

Constructor Details

Launcher

`public Launcher()`

Method Details

main

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

Package `com.example.meadowfx`

Class Miesozerca

`java.lang.Object`
`com.example.meadowfx.Zwierze`
`com.example.meadowfx.Miesozerca`

`public class Miesozerca`
`extends Zwierze`

Field Summary

Fields

Modifier and Type	Field	Description
(package private) int	<code>glod</code>	
<code>List <Miesozerca></code>	<code>listaLisow</code>	
<code>List <Roslinozerca></code>	<code>listaRoślinożerców</code>	
<code>List <Wszystkozerca></code>	<code>listaWszystkożerców</code>	
<code>List <List <String>></code>	<code>Mapa</code>	
(package private) int	<code>obrazenia</code>	
(package private) int	<code>ofiary</code>	
private boolean	<code>zabity</code>	

Fields inherited from class `com.example.meadowfx.Zwierze`

`nazwa`, `symbol`, `x`, `y`, `życie`

Constructor Summary

Constructors

Constructor	Description
<code>Miesozerca()</code>	

Method Summary

All Methods **Instance Methods** **Concrete Methods**

Modifier and Type	Method	Description
void	<code>atakujZwierzęta()</code>	

```
boolean      czyZabity()
void         poruszajSie()
void         wzmozGlod()
void         zabiJ()
void         zadajObrazenia(Zwierze zwierzę,
                           int obrazenia)

private Roslinozerca  znajdźRoślinożerca(int x, int y)
private Wszystkozerca  znajdźWszystkożerca(int x, int y)
```

Methods inherited from class java.lang.Object

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

Field Details

obrazenia

int obrazenia

glod

int glod

ofiary

int ofiary

zabity

private boolean zabity

listaLisow

public List <Miesozerca> listaLisow

listaRoślinożerców

public List <Roslinozerca> listaRoślinożerców

listaWszystkożerców

```
public List <Wszystkozerca> listaWszystkożerców
```

Mapa

```
public List <List <String >> Mapa
```

Constructor Details**Miesozerca**

```
public Miesozerca()
```

Method Details**wzmozGlod**

```
public void wzmozGlod()
```

poruszajSie

```
public void poruszajSie()
```

zadajObrazenia

```
public void zadajObrazenia(Zwierze zwierzę,  
                           int obrazenia)
```

atakujZwierzęta

```
public void atakujZwierzęta()
```

znajdźRoślinożerca

```
private Roslinozerca znajdźRoślinożerca(int x,  
                                           int y)
```

znajdźWszystkożerca

```
private Wszystkozerca znajdźWszystkożerca(int x,  
                                           int y)
```

czyZabity

```
public boolean czyZabity()
```

zabiJ

```
public void zabiJ()
```

Package `com.example.meadowfx`

Class Roslina

`java.lang.Object`
`com.example.meadowfx.Roslina`

Direct Known Subclasses:

`Jadalna`, `Trujaca`

```
public class Roslina
extends Object
```

Field Summary

Fields

Modifier and Type	Field	Description
(package private) <code>String</code>	<code>nazwa</code>	
(package private) <code>String</code>	<code>symbol</code>	
(package private) <code>int</code>	<code>wzrost</code>	
(package private) <code>int</code>	<code>x</code>	
(package private) <code>int</code>	<code>y</code>	

Constructor Summary

Constructors

Constructor	Description
<code>Roslina()</code>	

Method Summary

Methods inherited from class `java.lang.Object`

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

Field Details

`symbol`

`String` symbol

nazwa

`String` nazwa

wzrost

`int` wzrost

x

`int` x

y

`int` y

Constructor Details

Roslina

`public` Roslina()

Package `com.example.meadowfx`

Class Roslinozerca

`java.lang.Object`
`com.example.meadowfx.Zwierze`
`com.example.meadowfx.Roslinozerca`

`public class Roslinozerca`
`extends Zwierze`

Field Summary

Fields

Modifier and Type	Field	Description
(package private) int	<code>glod</code>	
List <Jadalna>	<code>listaJadalnych</code>	
List <Roslinozerca>	<code>listaRoślinozerców</code>	
List <Trujaca>	<code>listaTrujących</code>	
List <List <String >>	<code>Mapa</code>	
(package private) int	<code>spożytyPokarm</code>	
private boolean	<code>zabity</code>	

Fields inherited from class `com.example.meadowfx.Zwierze`

`nazwa`, `symbol`, `x`, `y`, `życie`

Constructor Summary

Constructors

Constructor	Description
<code>Roslinozerca()</code>	

Method Summary

All Methods	Instance Methods	Concrete Methods
Modifier and Type	Method	Description
List <Jadalna>	<code>getlistaJadalnych()</code>	
List <Trujaca>	<code>getListTrujących()</code>	

```
void          poruszajSie()
void          szukajrosliny()
void          wzmozGlod()
void          zjedzRośline(Roslina roslina)
private Jadalna znajdzGrzyba(int x, int y)
private Trujaca znajdzMuchomora(int x, int y)
```

Methods inherited from class java.lang.Object

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

Field Details

spożytyPokarm

int spożytyPokarm

zabity

private boolean zabity

glod

int glod

listaRoślinożerców

public List <Roslinozerca> listaRoślinożerców

listaJadalnych

public List <Jadalna> listaJadalnych

listaTrujących

public List <Trujaca> listaTrujących

Mapa

```
public List <List <String >> Mapa
```

Constructor Details

Roslinozerca

```
public Roslinozerca()
```

Method Details

wzmozGlod

```
public void wzmozGlod()
```

poruszajSie

```
public void poruszajSie()
```

zjedzRośline

```
public void zjedzRośline(Roslina roslina)
```

getlistaJadalnych

```
public List <Jadalna> getlistaJadalnych()
```

getListaTrujących

```
public List <Trujaca> getListaTrujących()
```

szukajrosliny

```
public void szukajrosliny()
```

znajdźGrzyba

```
private Jadalna znajdźGrzyba(int x,  
                             int y)
```

znajdźMuchomora

```
private Trujaca znajdźMuchomora(int x,  
                                 int y)
```

Package `com.example.meadowfx`

Class Trujaca

`java.lang.Object`
`com.example.meadowfx.Roslina`
`com.example.meadowfx.Trujaca`

`public class Trujaca`
`extends Roslina`

Field Summary

Fields

Modifier and Type	Field	Description
static <code>List</code> < <code>Trujaca</code> >	<code>listaMuchomorow</code>	
<code>List</code> < <code>List</code> < <code>String</code> >>	<code>Mapa</code>	
(package private) <code>int</code>	<code>Maxwiek</code>	
(package private) <code>int</code>	<code>sytosc</code>	
(package private) <code>int</code>	<code>trucizna</code>	
(package private) <code>int</code>	<code>wiek</code>	

Fields inherited from class `com.example.meadowfx.Roslina`

`nazwa`, `symbol`, `wzrost`, `x`, `y`

Constructor Summary

Constructors

Constructor	Description
<code>Trujaca()</code>	

Method Summary

All Methods	Static Methods	Instance Methods	Concrete Methods
Modifier and Type	Method		Description
static <code>List</code> < <code>Trujaca</code> >	<code>generujNoweMuchomory(List <List <String</code> <code>>> mapa)</code>		

```
void                setListaMuchomorkow(List
                    <Trujaca> listaMuchomora)

void                zwiekszWiek()
```

Methods inherited from class java.lang.Object

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

Field Details

trucizna

int trucizna

sytosc

int sytosc

wiek

int wiek

Maxwiek

int Maxwiek

Mapa

public List <List <String >> Mapa

listaMuchomorow

public static List <Trujaca> listaMuchomorow

Constructor Details

Trujaca

```
public Trujaca()
```

Method Details

zwiekszWiek

```
public void zwiekszWiek()
```

setListaMuchomorkow

```
public void setListaMuchomorkow(List <Trujaca> listaMuchomora)
```

generujNoweMuchomory

```
public static List <Trujaca> generujNoweMuchomory(List <List <String >> mapa)
```


Package `com.example.meadowfx`

Class Wszystkozerca

`java.lang.Object`
`com.example.meadowfx.Zwierze`
`com.example.meadowfx.Wszystkozerca`

`public class Wszystkozerca`
`extends Zwierze`

Field Summary

Fields

Modifier and Type	Field	Description
(package private) int	<code>glod</code>	
List <Jadalna>	<code>listaJadalnych</code>	
List <Miesozerca>	<code>listaLisow</code>	
List <Roslinozerca>	<code>listaRoślinożerców</code>	
List <Trujaca>	<code>listaTrujących</code>	
List <Wszystkozerca>	<code>listaWszystkożerców</code>	
List <List <String >>	<code>Mapa</code>	
(package private) int	<code>obrazenia</code>	
(package private) int	<code>ofiary</code>	
(package private) int	<code>spożytyPokarm</code>	
private boolean	<code>zabity</code>	

Fields inherited from class `com.example.meadowfx.Zwierze`

`nazwa`, `symbol`, `x`, `y`, `życie`

Constructor Summary

Constructors

Constructor	Description
<code>Wszystkozerca()</code>	

Method Summary

All Methods	Instance Methods	Concrete Methods
Modifier and Type	Method	Description
void	atakujZwierzęta()	
boolean	czyZabity()	
List <Jadalna>	getlistaJadalnych()	
List <Trujaca>	getListaTrujących()	
void	poruszajSie()	
void	szukajrosliny()	
void	unik()	
void	wzmozGlod()	
void	zabiJ()	
void	zadajObrazenia(Zwierze zwierzę, int obrazenia)	
void	zjedzRośline(Roslina roslina)	
private Jadalna	znajdźGrzyba(int x, int y)	
private Miesozerca	znajdźMięzożerca(int x, int y)	
private Trujaca	znajdźMuchomora(int x, int y)	
private Roslinozerca	znajdźRoślinożerca(int x, int y)	

Methods inherited from class java.lang.Object

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

Field Details

obrazenia
int obrazenia
ofiary
int ofiary
glod
int glod

spożytyPokarm

```
int spożytyPokarm
```

zabity

```
private boolean zabity
```

listaLisow

```
public List <Miesozerca> listaLisow
```

listaRoślinożerców

```
public List <Roslinozerca> listaRoślinożerców
```

listaWszystkożerców

```
public List <Wszystkozerca> listaWszystkożerców
```

listaJadalnych

```
public List <Jadalna> listaJadalnych
```

listaTrujących

```
public List <Trujaca> listaTrujących
```

Mapa

```
public List <List <String >> Mapa
```

Constructor Details**Wszystkozerca**

```
public Wszystkozerca()
```

Method Details

wzmozGlod

```
public void wzmozGlod()
```

poruszajSie

```
public void poruszajSie()
```

zjedzRośline

```
public void zjedzRośline(Roslina roslina)
```

getlistaJadalnych

```
public List <Jadalna> getlistaJadalnych()
```

getListaTrujących

```
public List <Trujaca> getListaTrujących()
```

szukajrosliny

```
public void szukajrosliny()
```

znajdźGrzyba

```
private Jadalna znajdźGrzyba(int x,  
                             int y)
```

znajdźMuchomora

```
private Trujaca znajdźMuchomora(int x,  
                                 int y)
```

zadajObrazenia

```
public void zadajObrazenia(Zwierze zwierze,  
                           int obrazenia)
```

atakujZwierzęta

```
public void atakujZwierzęta()
```

znajdźRoślinożerca

```
private Roslinozerca znajdźRoślinożerca(int x,  
                                         int y)
```

znajdźMięzożerca

```
private Miesozerca znajdźMięzożerca(int x,  
                                     int y)
```

unik

```
public void unik()
```

czyZabity

```
public boolean czyZabity()
```

zabiJ

```
public void zabiJ()
```

Package `com.example.meadowfx`

Class `Zwierze`

`java.lang.Object`
`com.example.meadowfx.Zwierze`

Direct Known Subclasses:
`Miesozerca`, `Roslinozerca`, `Wszystkozerca`

`public class Zwierze`
`extends Object`

Field Summary

Fields

Modifier and Type	Field	Description
(package private) <code>String</code>	<code>nazwa</code>	
(package private) <code>String</code>	<code>symbol</code>	
(package private) <code>int</code>	<code>x</code>	
(package private) <code>int</code>	<code>y</code>	
(package private) <code>int</code>	<code>życie</code>	

Constructor Summary

Constructors

Constructor	Description
<code>Zwierze()</code>	

Method Summary

Methods inherited from class `java.lang.Object`

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

Field Details

nazwa

`String` nazwa

symbol

`String` symbol

życie

`int` życie

x

`int` x

y

`int` y

Constructor Details

Zwierze

`public` Zwierze()

Hierarchy For Package com.example.meadowfx

Class Hierarchy

- java.lang.**Object**
 - javafx.application.Application
 - com.example.meadowfx.**HelloApplication**
 - com.example.meadowfx.**HelloController** (implements javafx.fxml.Initializable)
 - com.example.meadowfx.**Launcher**
 - com.example.meadowfx.**Roslina**
 - com.example.meadowfx.**Jadalna**
 - com.example.meadowfx.**Trujaca**
 - com.example.meadowfx.**Zwierze**
 - com.example.meadowfx.**Miesozerca**
 - com.example.meadowfx.**Roslinozerca**
 - com.example.meadowfx.**Wszystkozerca**

Index

[A](#) [C](#) [D](#) [E](#) [G](#) [H](#) [I](#) [J](#) [L](#) [M](#) [N](#) [O](#) [P](#) [R](#) [S](#) [T](#) [U](#) [W](#) [X](#) [Y](#) [Z](#) [Ż](#)

[All Classes and Interfaces](#) | [All Packages](#)

A

aktualneGrzyby - Variable in class [com.example.meadowfx.HelloController](#)

aktualneJeze - Variable in class [com.example.meadowfx.HelloController](#)

aktualneLisy - Variable in class [com.example.meadowfx.HelloController](#)

aktualneMuchomory - Variable in class [com.example.meadowfx.HelloController](#)

aktualneSarny - Variable in class [com.example.meadowfx.HelloController](#)

aktualnie - Variable in class [com.example.meadowfx.HelloController](#)

atakujZwierzęta() - Method in class [com.example.meadowfx.Miesozerca](#)

atakujZwierzęta() - Method in class [com.example.meadowfx.Wszystkozerca](#)

[A](#) [C](#) [D](#) [E](#) [G](#) [H](#) [I](#) [J](#) [L](#) [M](#) [N](#) [O](#) [P](#) [R](#) [S](#) [T](#) [U](#) [W](#) [X](#) [Y](#) [Z](#) [Ż](#)

[All Classes and Interfaces](#) | [All Packages](#)

JavaDoc Help

- **Navigation:**
 - [Search](#)
 - **Kinds of Pages:**
 - [Package](#) • [Class or Interface](#) • [Other Files](#) • [Tree \(Class Hierarchy\)](#) • [All Packages](#)
 - [All Classes and Interfaces](#) • [Index](#)
-

Navigation

Starting from the [Overview](#) page, you can browse the documentation using the links in each page, and in the navigation bar at the top of each page. The [Index](#) and Search box allow you to navigate to specific declarations and summary pages, including: [All Packages](#), [All Classes and Interfaces](#)

Search

You can search for definitions of modules, packages, types, fields, methods, system properties and other terms defined in the API. These items can be searched using part or all of the name, optionally using "camelCase" abbreviations, or multiple search terms separated by whitespace. Some examples:

- "j.l.obj" matches "java.lang.Object"
- "InpStr" matches "java.io.InputStream"
- "math exact long" matches "java.lang.Math.absExact(long)"

Refer to the [Javadoc Search Specification](#) for a full description of search features.

Kinds of Pages

The following sections describe the different kinds of pages in this collection.

Package

Each package has a page that contains a list of its classes and interfaces, with a summary for each. These pages may contain the following categories:

- Interfaces
- Classes
- Enum Classes
- Exception Classes
- Annotation Interfaces

Class or Interface

Each class, interface, nested class and nested interface has its own separate page. Each of these pages has three sections consisting of a declaration and description, member summary tables, and detailed member descriptions. Entries in each of these sections are omitted if they are empty or not applicable.

- Class Inheritance Diagram
- Direct Subclasses
- All Known Subinterfaces
- All Known Implementing Classes
- Class or Interface Declaration
- Class or Interface Description
- Nested Class Summary
- Enum Constant Summary
- Field Summary
- Property Summary

- Constructor Summary
 - Method Summary
 - Required Element Summary
 - Optional Element Summary
-
- Enum Constant Details
 - Field Details
 - Property Details
 - Constructor Details
 - Method Details
 - Element Details

Note: Annotation interfaces have required and optional elements, but not methods. Only enum classes have enum constants. The components of a record class are displayed as part of the declaration of the record class. Properties are a feature of JavaFX.

The summary entries are alphabetical, while the detailed descriptions are in the order they appear in the source code. This preserves the logical groupings established by the programmer.

Other Files

Packages and modules may contain pages with additional information related to the declarations nearby.

Tree (Class Hierarchy)

There is a [Class Hierarchy](#) page for all packages, plus a hierarchy for each package. Each hierarchy page contains a list of classes and a list of interfaces. Classes are organized by inheritance structure starting with `java.lang.Object`. Interfaces do not inherit from `java.lang.Object`.

- When viewing the Overview page, clicking on TREE displays the hierarchy for all packages.
- When viewing a particular package, class or interface page, clicking on TREE displays the hierarchy for only that package.

All Packages

The [All Packages](#) page contains an alphabetic index of all packages contained in the documentation.

All Classes and Interfaces

The [All Classes and Interfaces](#) page contains an alphabetic index of all classes and interfaces contained in the documentation, including annotation interfaces, enum classes, and record classes.

Index

The [Index](#) contains an alphabetic index of all classes, interfaces, constructors, methods, and fields in the documentation, as well as summary pages such as [All Packages](#), [All Classes and Interfaces](#).

This help file applies to API documentation generated by the standard doclet.