Санкт-Петербургский Национальный Исследовательский Университет

Информационных Технологий, Механики и Оптики

 ФКТиУ, кафедра Вычислительной техники

Лабораторная работа №4

Вариант 10767.4

Преподаватель: Письмак Алексей

Выполнила: Степанов Максим

Р3110

Санкт-Петербург

2021

# Задание:

## **Программа должна удовлетворять следующим требованиям:**

1. В программе должны быть реализованы 2 собственных класса исключений (checked и unchecked), а также обработка исключений этих классов.
2. В программу необходимо добавить использование локальных, анонимных и вложенных классов (static и non-static).

## **Порядок выполнения работы:**

1. Доработать объектную модель приложения.
2. Перерисовать диаграмму классов в соответствии с внесёнными в модель изменениями.
3. Согласовать с преподавателем изменения, внесённые в модель.
4. Модифицировать программу в соответствии с внесёнными в модель изменениями.

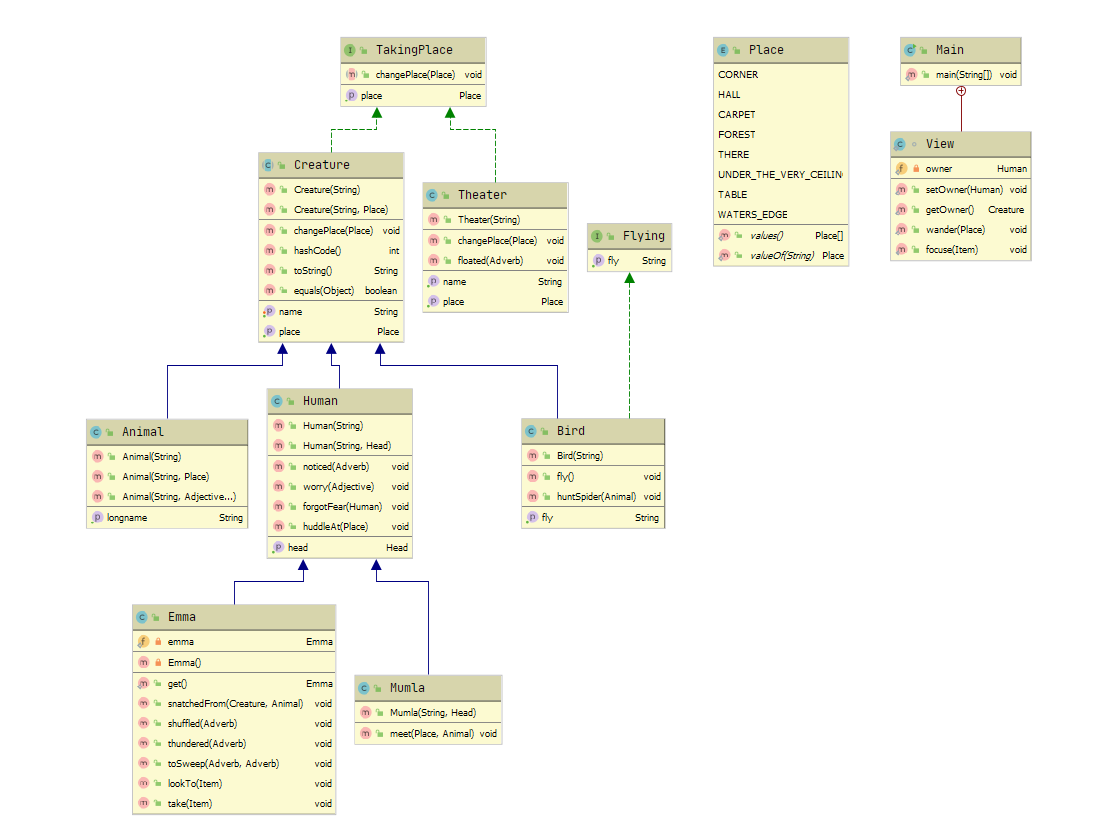
## **Отчёт по работе должен содержать:**

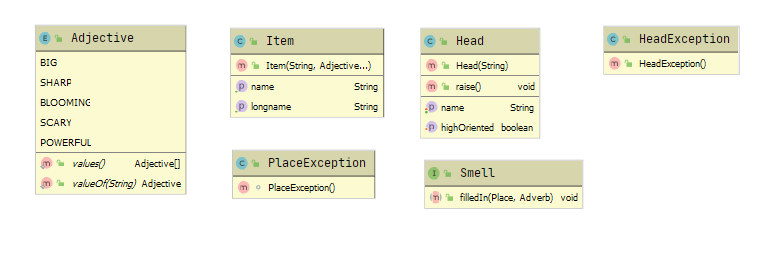
1. Текст задания.
2. Диаграмма классов объектной модели.
3. Исходный код программы.
4. Результат работы программы.
5. Выводы по работе.

# Описание предметной области:

**Эмма смотрела на сыр... Потом схватила ломтик сыра и сунула его в карман. Ее взгляд блуждал по столу и остановился на блинчике.** Эммa так и выхватила у него рыбку из лапы и с высоко поднятой головой прошоркала в свой угол. Она долго гремела там и, вытащив наконец большую метлу, принялась усердно мести. Вечером острый запах цветущей рябины заполнил зал. Птички порхали под самым потолком, охотясь за пауками, а малышка Мю повстречала на ковре в зале большого страшного муравья. Только теперь все заметили, что театр плыл уже в лесу. **Все пришли в сильное волнение. Забыв свой страх перед Эммой, они сгрудились у самой воды.**

# Диаграмма классов:





# Исходный код:

Main.java - класс отвечающий за работу программы.

**public class** Main {  
 **public static void** main(String[] args) {  
 Item cheese = **new** Item(**"cheese"**);  
 Emma.*get*().lookTo(cheese);  
 Emma.*get*().take(cheese);  
  
 View.*setOwner*(Emma.*get*());  
 View.*wander*(Place.***TABLE***);  
 View.*focuse*(**new** Item(**"pancake"**));  
  
 *// Локальный класс, реализует кусок текста* **class** He **extends** Human{  
 **private final** Paw **paw**=**new** Paw(**"paw"**);  
  
 **public** He(String name) {  
 **super**(name);  
 }  
  
 **public** Paw getPaw() {  
 **return paw**;  
 }  
  
 **class** Paw{  
 **private final** Animal **fish** = **new** Animal(**"fish"**);  
 **private** String **name**;  
  
 **public** Paw(String name){  
 **this**.**name**=name;  
 }  
  
 **public** Animal getFish() {  
 **return fish**;  
 }  
 }  
 }  
 He he = **new** He(**"his"**);  
 Emma.*get*().snatchedFrom(he, he.getPaw().getFish());  
  
 Emma.*get*().getHead().setHighOriented(**true**);  
 Emma.*get*().getHead().raise();  
 Emma.*get*().changePlace(Place.***CORNER***);  
 Emma.*get*().shuffled(Adverb.***OWN***);  
  
  
 Emma.*get*().thundered(Adverb.***ALONG\_WHILE***);  
 Emma.*get*().toSweep(Adverb.***FINALLY***,Adverb.***DILIGENTLY***);  
  
 *// Анонимный клсс, реализует функциональный интерфейс Запах* Smell smell = **new** Smell() {  
 @Override  
 **public void** filledIn(Place place, Adverb adverb) {  
 Item rowan = **new** Item(**"rowan"**,Adjective.***BLOOMING***);  
 System.***out***.print(adverb + **" smell of "**);  
 System.***out***.println(rowan.getLongname()+**" filled in the "**+place.toString());  
 }  
 };  
 smell.filledIn(Place.***HALL***, Adverb.***IN\_THE\_EVENING***);  
  
  
 **for** (**int** i = 0; i < (**int**)(Math.*random*() \* 3+3); i++) {  
 Bird bird = **new** Bird(**"bird"**);  
 Animal spider = **new** Animal(**"spider"**);  
 bird.fly();  
 bird.huntSpider(spider);  
 }  
  
  
 Mumla mu = **new** Mumla(**"baby mumla"**,**new** Head(**"head"**));  
 Animal ant = **new** Animal(**"Ant"**,Adjective.***SCARY***,Adjective.***BIG***);  
 mu.meet(Place.***HALL***,ant);  
  
  
 mu.noticed(Adverb.***ONLY\_NOW***);  
 Emma.*get*().noticed(Adverb.***ONLY\_NOW***);  
 Theater theater = **new** Theater(**"theater"**);  
  
 theater.changePlace(Place.***FOREST***); *// Если убрать, или поменять, выйдет ошибка* **try** {  
 theater.floated(Adverb.***ALREADY***);  
 } **catch** (PlaceException e){  
 e.printStackTrace();  
 }  
  
 mu.worry(Adjective.***POWERFUL***);  
 Emma.*get*().worry(Adjective.***POWERFUL***);  
  
 mu.forgotFear(Emma.*get*());  
 he.forgotFear(Emma.*get*());  
 mu.huddleAt(Place.***WATERS\_EDGE***);  
 he.huddleAt(Place.***WATERS\_EDGE***);  
 }  
  
 *// Внутренний запах, реализует взгляд человека* **static class** View{  
 **private static** Human *owner*;  
  
 **public static void** setOwner(Human owner) **throws** HeadException {  
 **if**(owner.getHead()==**null**){  
 **throw new** HeadException();  
 }  
 View.*owner* = owner;  
 }  
  
 **public static** Creature getOwner() {  
 **return** *owner*;  
 }  
  
 **public static void** wander(Place place){  
 System.***out***.println(*getOwner*().getName()+**"'s view wandered over the "**+place.toString());  
 }  
  
 **public static void** focuse(Item item) {  
 System.***out***.println(*getOwner*().getName()+**" view focused on "**+item.getName());  
 }  
 }  
}

# Результат работы:

Emma looking to cheese

Emma took cheese and put in the pocket

Emma's view wandered over the TABLE

Emma view focused on pancake

Emma snatched fish from his paw

with head held high

shuffled OWN CORNER

She ALONG\_WHILE thundered THERE

and FINALLY began to sweep DILIGENTLY with a BIG broom

IN\_THE\_EVENING smell of BLOOMING rowan filled in the HALL

bird flitter UNDER\_THE\_VERY\_CEILING

bird hunting for spider

bird flitter UNDER\_THE\_VERY\_CEILING

bird hunting for spider

bird flitter UNDER\_THE\_VERY\_CEILING

bird hunting for spider

bird flitter UNDER\_THE\_VERY\_CEILING

bird hunting for spider

bird flitter UNDER\_THE\_VERY\_CEILING

bird hunting for spider

baby baby mumla met on the CARPET of the HALL BIG

ONLY\_NOW baby mumla noticed, that

ONLY\_NOW She noticed, that

theater floated ALREADY in the FOREST

baby mumla came to POWERFUL worry

She came to POWERFUL worry

baby mumla forgot fear up to She

his forgot fear up to She

baby mumla huddled at the WATERS\_EDGE

his huddled at the WATERS\_EDGE

Process finished with exit code 0

# Вывод:

Во время выполнения лабораторной работы укрепил свои знания объектно-ориентированного программирования, научился использовать вложенные и анонимные классы, а также создавать собственные классы исключений (checked и unchecked).