Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Дисциплина: современные языки программирования

**Лабораторная работа №1**

**«Использование языка программирования Kotlin»**

Вариант 4

Выполнил: Гарбарук Д.А.

студент гр. 310102

Проверил: Усенко Ф.В.

Минск 2024

Цель: Выполнить разработку приложения с использованием языка программирования Kotlin.

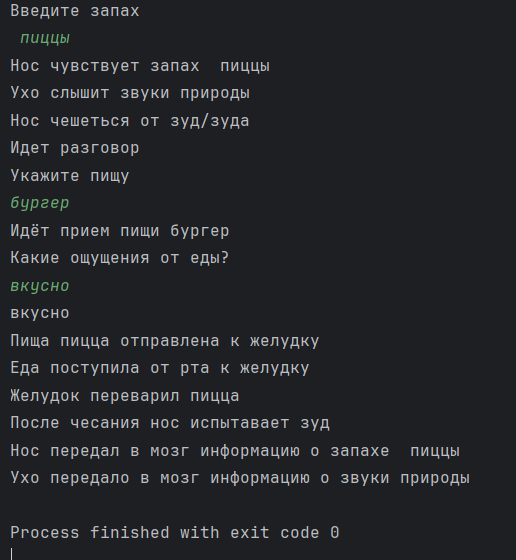
Задание: Добавить классы Орган тела и Абстрактный орган. Добавить класс ухо, наследующий от класса Орган головы, класс Желудок, наследующий от класса Орган тела. Организовать взаимодействие класса Рот с классом Желудок и класса Ухо с классом Мозг.

Листинг кода:

abstract class Organ(open val \_weight: Double?) {}  
  
open class Head(weight: Double?) :Organ(weight){  
 private var list\_head: MutableList<Head> =*mutableListOf*()  
 fun getValue(): MutableList<Head>{  
 return list\_head  
 }  
 fun setValue(obj: Head): Head  
 {  
 list\_head.add(obj)  
 return obj  
 }  
}  
  
open class Body(weight: Double?) :Organ(weight){  
 private var list\_body: MutableList<Body> =*mutableListOf*()  
 fun getValue(): MutableList<Body>{  
 return list\_body  
 }  
 fun setValue(obj: Body): Body  
 {  
 list\_body.add(obj)  
 return obj  
 }  
}  
open class Organ\_Head(weight: Double?) :Head(weight){}  
open class Organ\_Body(weight: Double?) :Body(weight){}  
  
class Ear(weight: Double?, var earring: Boolean) :Organ\_Head(weight){  
 fun isEarring(): Boolean{  
 return earring  
 }  
 fun Hear(\_sound: String):String{  
 *println*("Ухо слышит $\_sound")  
 return \_sound  
 }  
}  
class Nose(weight: Double?,private var shape: String) :Organ\_Head(weight) {  
 fun Sniff(\_smell: String?): String? {  
 *println*("Нос чувствует запах $\_smell")  
 return \_smell  
 }  
  
 fun Itch(\_itch: String): String {  
 *println*("Нос чешеться от $\_itch")  
 return \_itch  
 }  
}  
class Brain(weight: Double?,private var kpd: Int) : Organ\_Head(weight) {  
 var temp\_smell:String?=null  
 fun Manage\_Mouth(manage:Mouth):String?{  
 *println*("Укажите пищу")  
 var food: String?= *readLine*()  
 var temp:String?=manage.Eat(food)  
 return temp  
 }  
 fun Manage\_Nose(\_manage:Nose):String?{  
 *println*("Введите запах")  
 var \_smell: String?= *readLine*()  
 temp\_smell=\_smell  
 var temp:String?=\_manage.Sniff(\_smell)  
 return temp  
 }  
 fun getSound(func: (String) -> String, sound: String) {  
 *println*("Ухо передало в мозг информацию о $sound")  
 }  
  
 fun getSmell() {  
 *println*("Нос передал в мозг информацию о запахе $temp\_smell")  
 }  
  
 fun getItch(func: (String) -> String, itch: String) {  
 *println*("После чесания нос испытавает $itch")  
 }  
}  
  
class Mouth(weight: Double?, var size: String) : Organ\_Body(weight) {  
 private fun Info():String? {  
 *println*("Какие ощущения от еды?")  
 var info:String?= *readLine*()  
 return info  
 }  
 fun Eat(food: String?):String? {  
 *println*("Идёт прием пищи $food")  
 var \_temp:String?=Info()  
 *println*(\_temp)  
 return \_temp  
 }  
 fun Speak(words: String){  
 *println*(words)  
 }  
 private fun Transf(food: String?):String?{  
 *println*("Еда поступила от рта к желудку")  
 return food  
 }  
 fun Transfering(food: String?):String?{  
 var temp:String?  
 temp=Transf(food)  
 return temp  
 }  
}  
  
class Stomach(weight: Double?, private var volume: Double,var mouth:Mouth) : Organ\_Body(weight) {  
 private fun Digestion(\_food:String?)  
 {  
 var food:String?=mouth.Transfering(\_food)  
 *println*("Желудок переварил $food")  
 }  
 fun getFood(func: (String?) -> String?, food: String) {  
 var \_food:String?=food  
 *println*("Пища $food отправлена к желудку")  
 Digestion(\_food)  
 }  
  
}

fun main()  
{  
 val brain=Brain(44.2,78)  
 val nose=Nose(13.5,"большой")  
 val mouth=Mouth(21.3,"средний")  
 val stomach=Stomach(58.4,59.6,mouth)  
 val ear=Ear(35.7,true)  
 brain.Manage\_Nose(nose)  
 ear.Hear("звуки природы")  
 nose.Itch("зуд/зуда")  
 mouth.Speak("Идет разговор")  
 brain.Manage\_Mouth(mouth)  
 stomach.getFood(mouth::Eat,"пицца")  
 brain.getItch(nose::Itch,"зуд")  
 brain.getSmell()  
 brain.getSound(ear::Hear,"звуки природы")  
}

Результат выполнения кода:



Вывод: разработана программа на языке программирования Kotlin с помощью двух диаграмм (диаграмма классов и диаграмма последовательности). В результате разработки программы на языке Kotlin была успешно реализована функциональность, позволяющая реализовать поведение органов и их связь друг с другом. Была проделана работа с функциями, классами, полями и методами.