Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

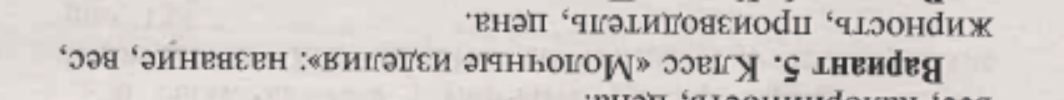
Отчёт по программе «**Практика 9**»

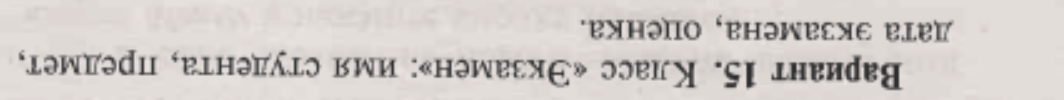
Выполнил: Молотова Виктория Анатольвена

Группа: ПР-21

Преподаватель: Мирошниченко Г.В

2023

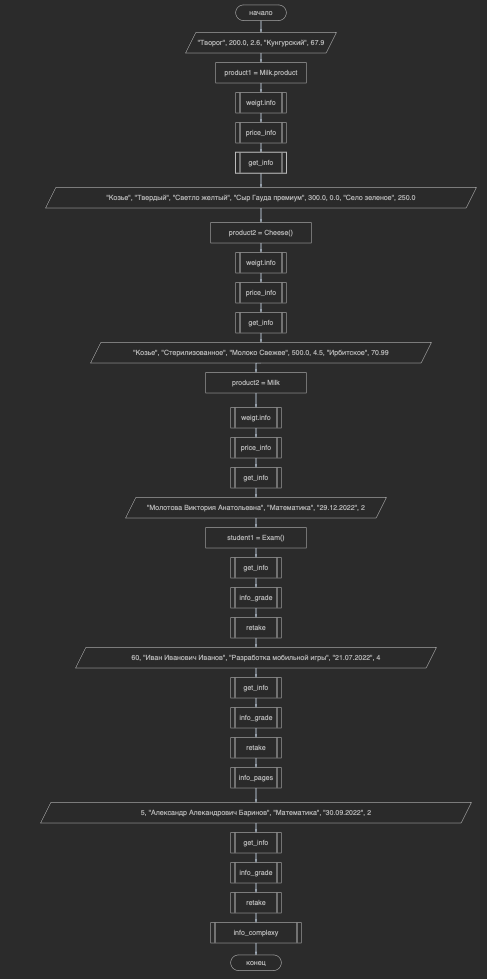
**Задание:**  
Добавить классы-наследники с дополнительным своим конструктором и функциями;  
****

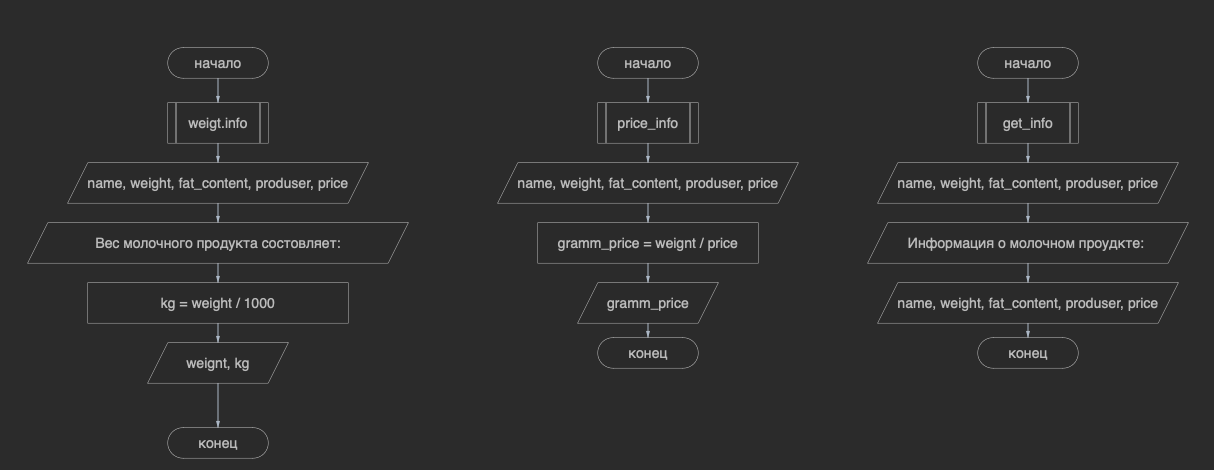
****

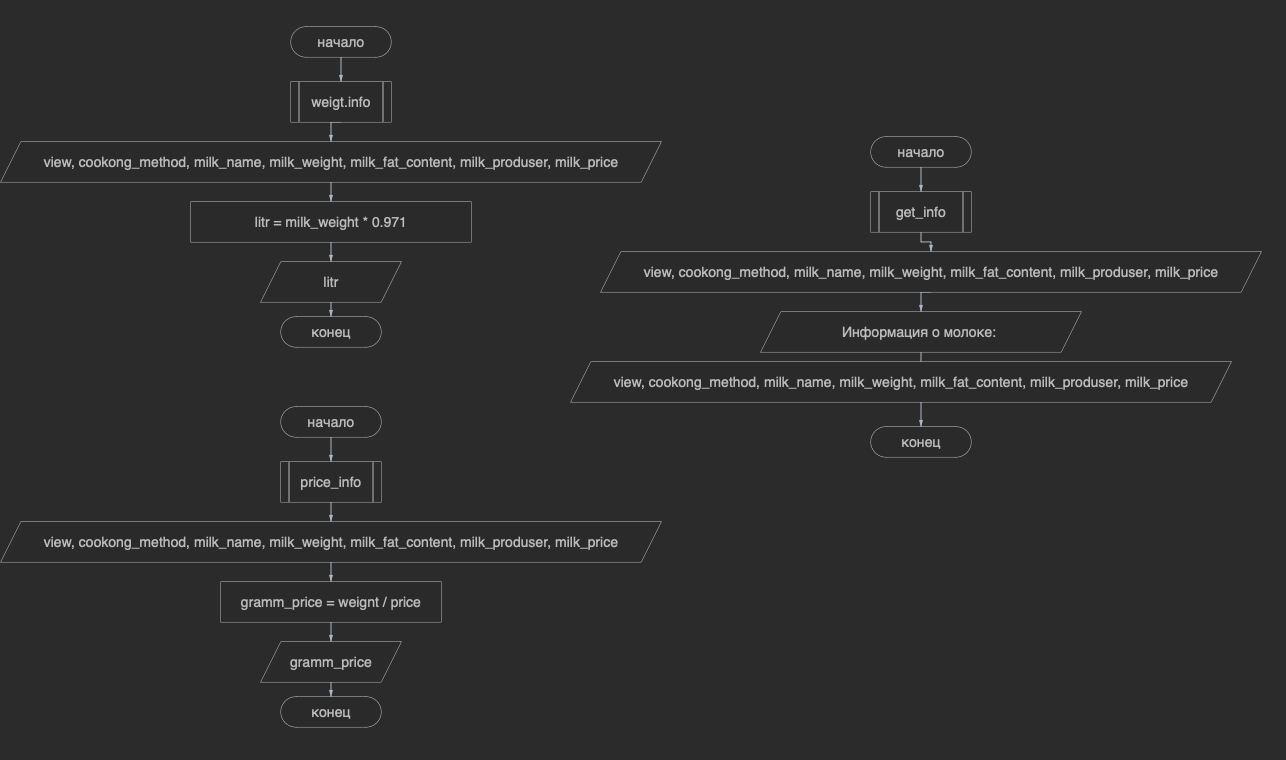
**Входные и выходные данные**

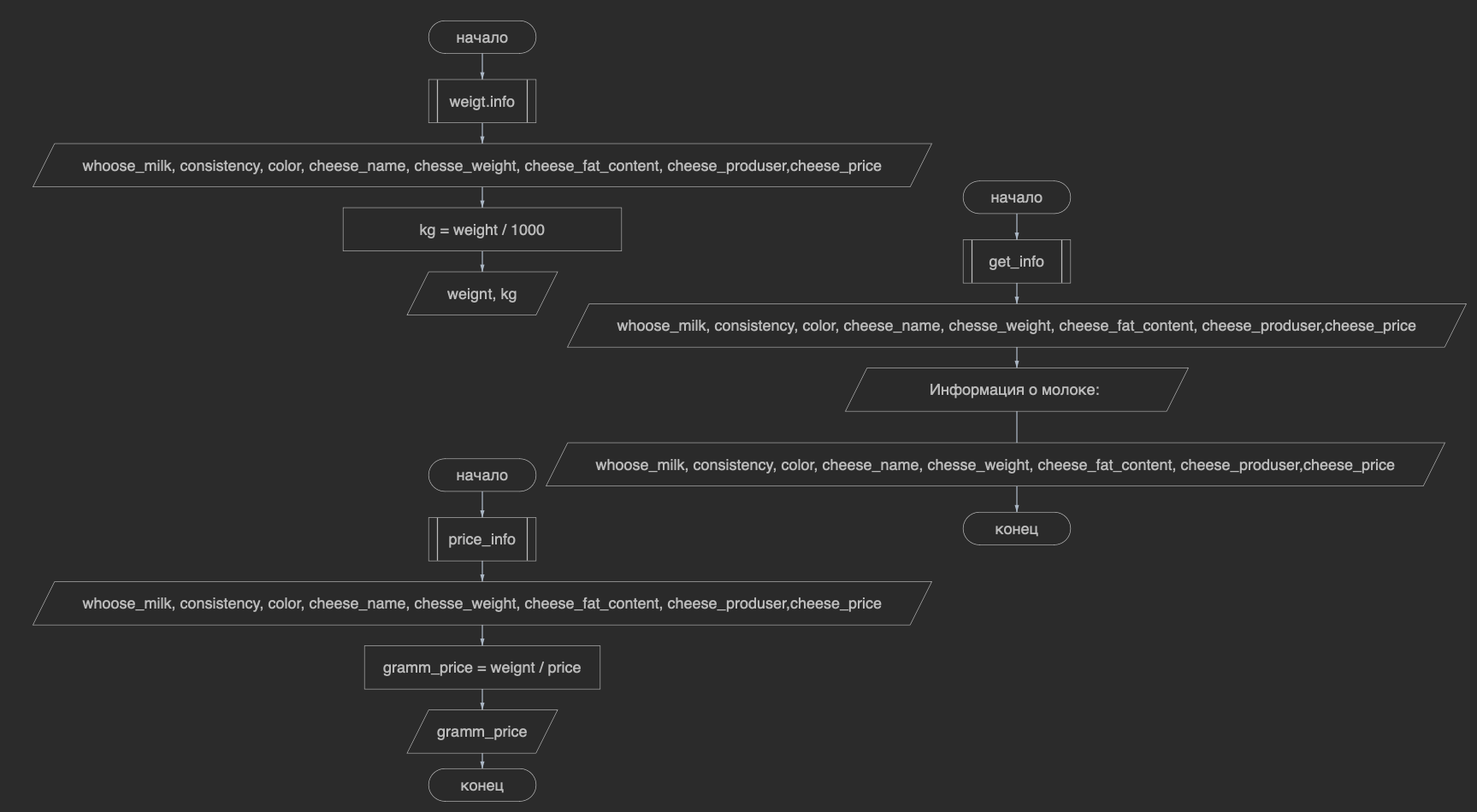
* *Входные данные:* var name: String, var weight : Double, var fat\_content:Double, var producer:String, var price:Double, var whoose\_milk : String, var consistency:String, var color: String, var chees\_name:String, var chees\_weight:Double, var chees\_fat\_content:Double, var chees\_prosucer:String, var chees\_price:Double, var view : String, var cooking\_method:String, var milk\_name:String, var milk\_weight:Double, var milk\_fat\_content:Double, var milk\_prosucer:String, var milk\_price:Double, var name:String, var lesson:String, var data:String, var grade:Int, var pages:Int, var diplom\_student\_name: String, var diplom\_tema:String, var diplom\_data: String, var diplom\_grade:Int, var complexity:Int, var hw\_student\_bame:String, var hw\_lesson:String, var hw\_data:String, var hw\_grade:Int
* *Выходные данные: kg:Double, gramm\_price: Double, litr:Double,* var name: String, var weight : Double, var fat\_content:Double, var producer:String, var price:Double, var whoose\_milk : String, var consistency:String, var color: String, var chees\_name:String, var chees\_weight:Double, var chees\_fat\_content:Double, var chees\_prosucer:String, var chees\_price:Double, var view : String, var cooking\_method:String, var milk\_name:String, var milk\_weight:Double, var milk\_fat\_content:Double, var milk\_prosucer:String, var milk\_price:Double, var name:String, var lesson:String, var data:String, var grade:Int, var pages:Int, var diplom\_student\_name: String, var diplom\_tema:String, var diplom\_data: String, var diplom\_grade:Int, var complexity:Int, var hw\_student\_bame:String, var hw\_lesson:String, var hw\_data:String, var hw\_grade:Int

**Блок-схема**

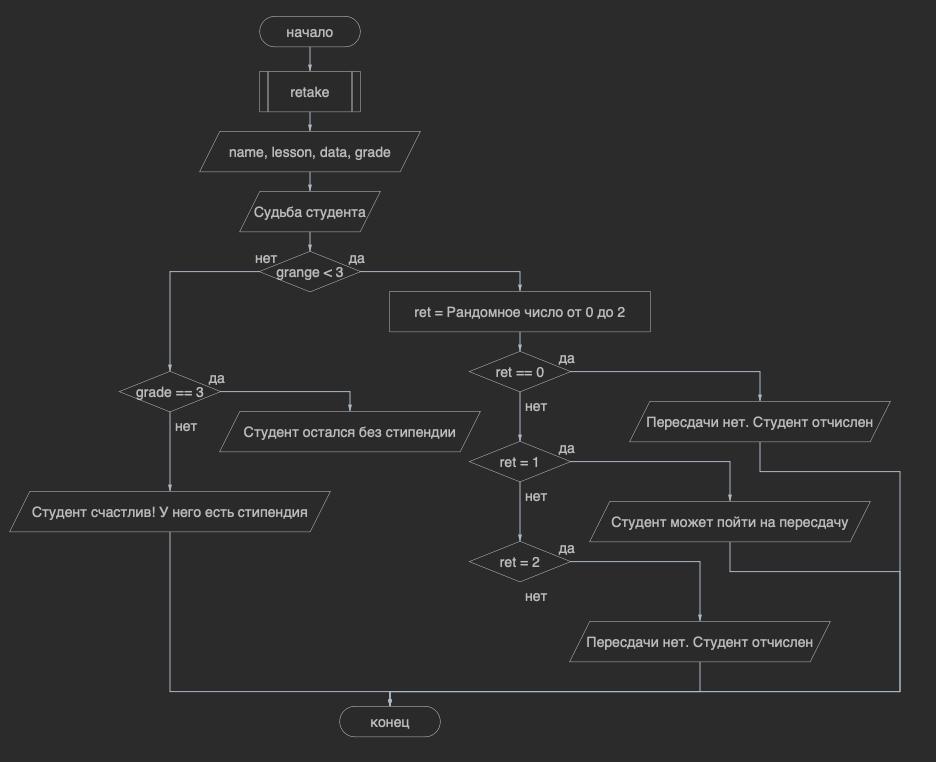
****

****

****

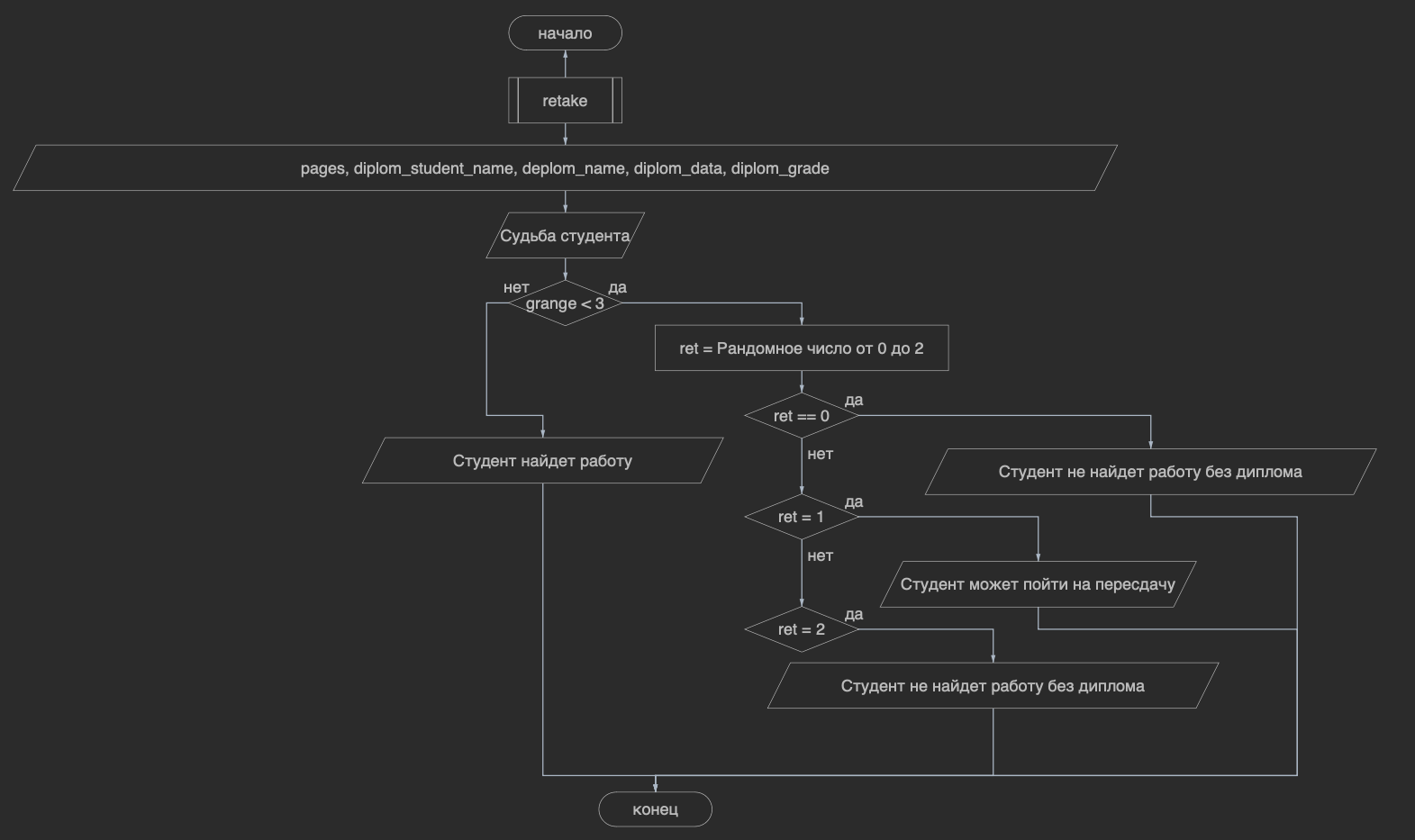
****

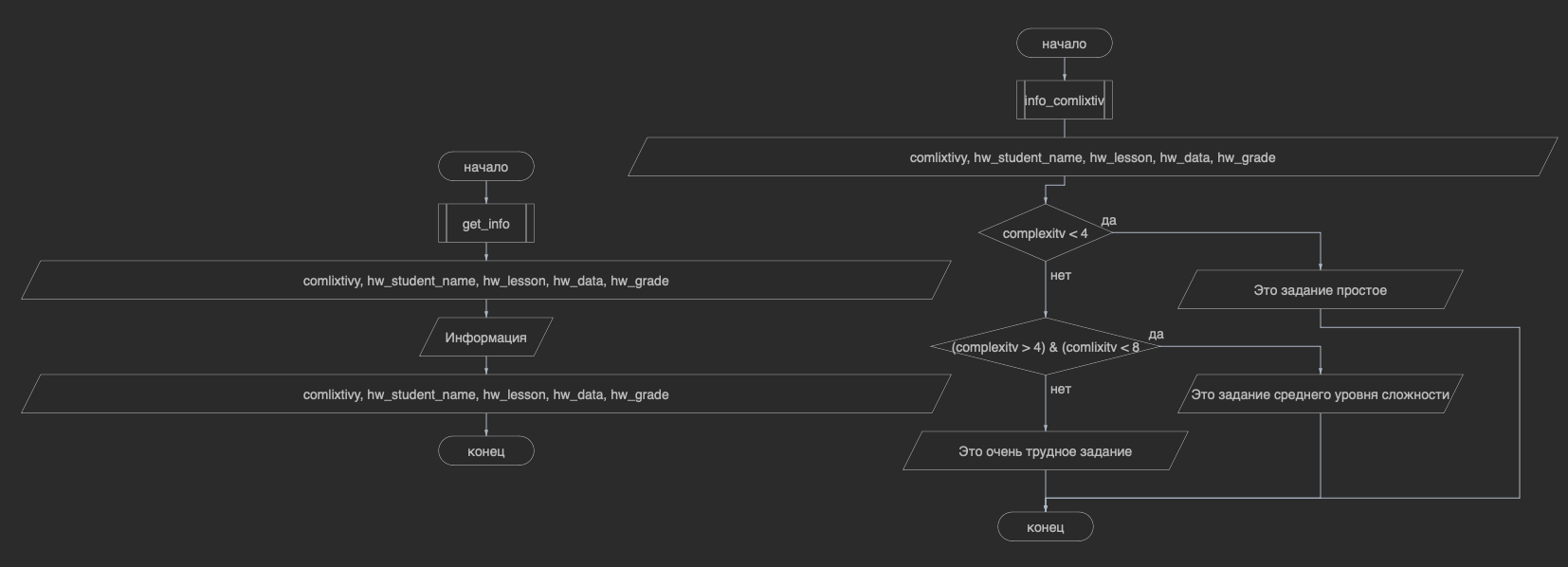
****

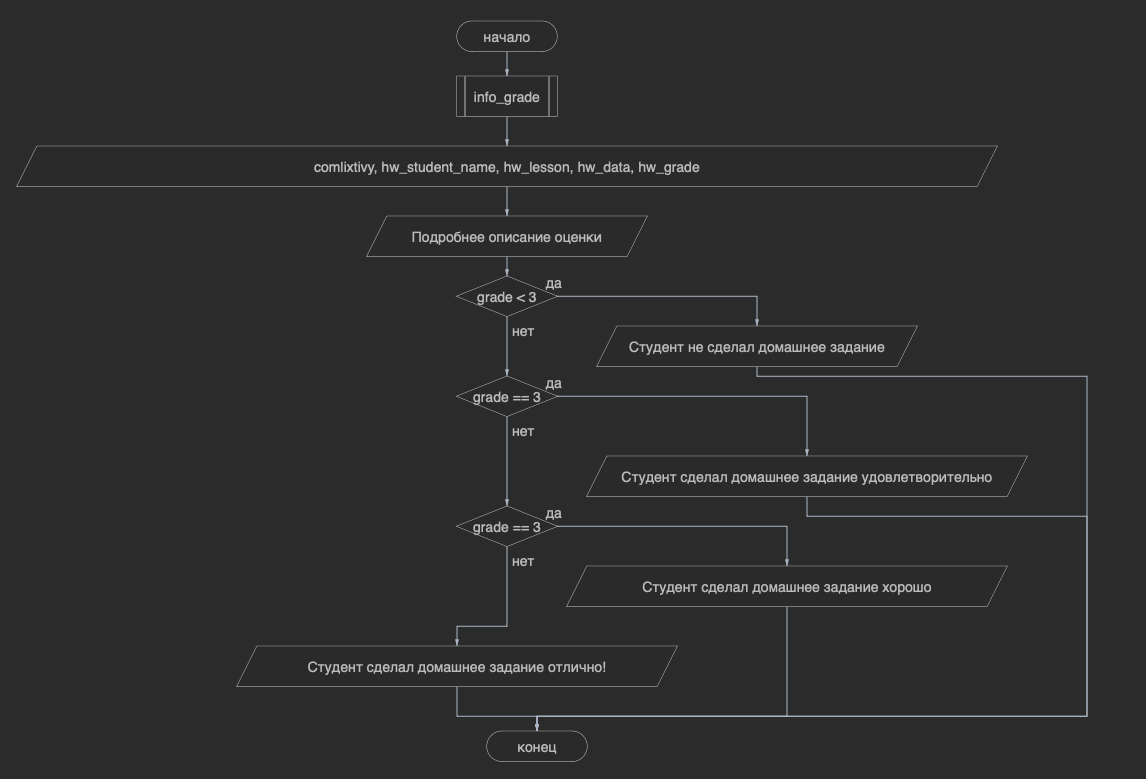
****

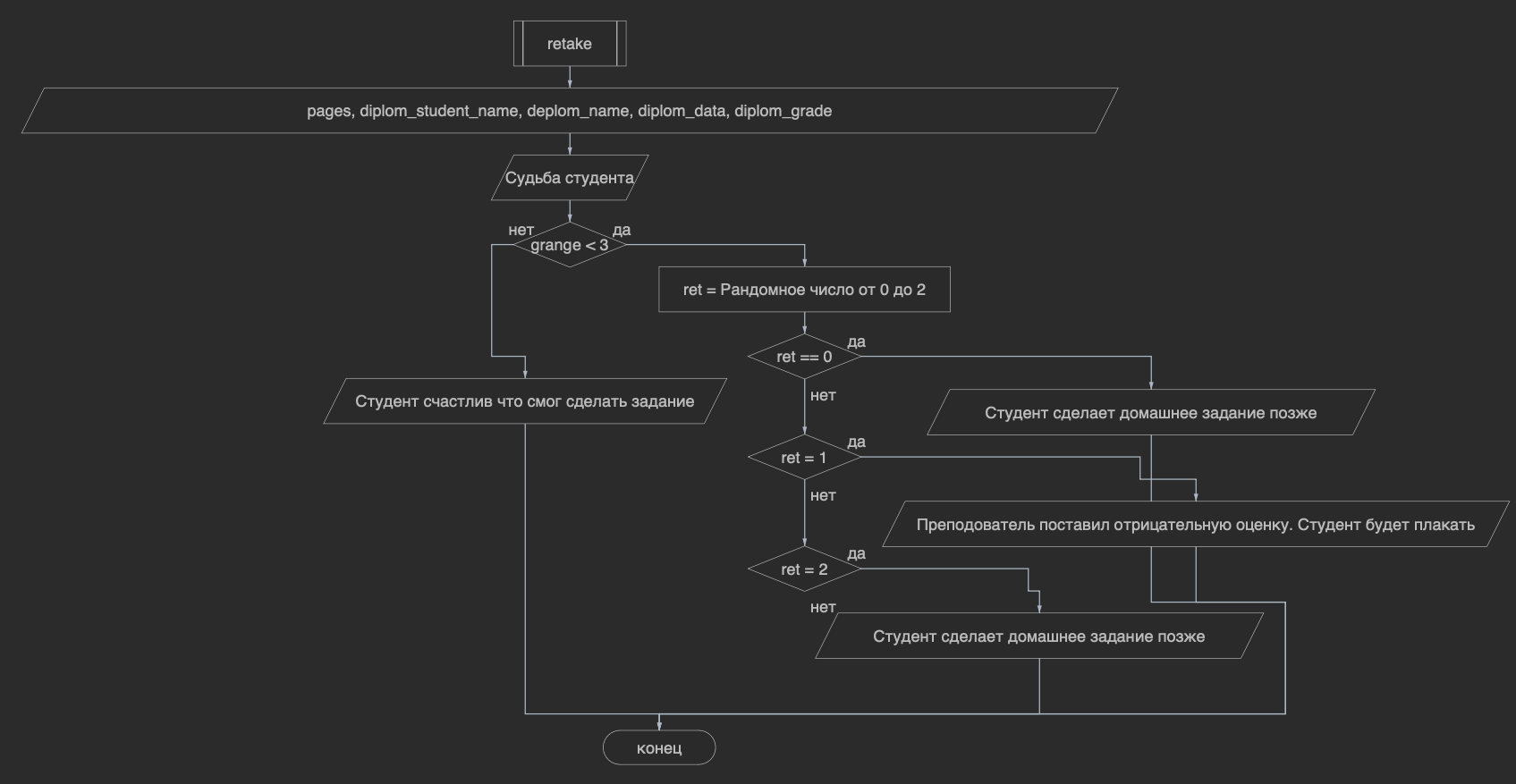
****

****

****

****

****

****

**Листинг программы**

//main.kt  
import Milk\_product.Milk\_product  
import Exam.Exam  
fun main(args: Array<String>) {  
 //Вариант 5  
  
 val product1 = Milk\_product("Творог", 200.0, 2.6, "Кунгурский", 67.9)  
 product1.weigt\_info()  
 product1.price\_info()  
 product1.get\_info()  
 *println*()  
  
 val product2 = Cheese("Козье", "Твердый", "Светло желтый", "Сыр Гауда премиум", 300.0, 0.0, "Село зеленое", 250.0)  
 product2.weigt\_info()  
 product2.price\_info()  
 product2.get\_info()  
 *println*()  
  
 val product3 = Milk("Козье", "Стерилизованное", "Молоко Свежее", 500.0, 4.5, "Ирбитское", 70.99)  
 product3.weigt\_info()  
 product3.price\_info()  
 product3.get\_info()  
 *println*()  
  
  
 //Вариант 15  
 //(var name:String, var lesson:String, var data:String, var grade:Int )\  
  
 val student1 = Exam("Молотова Виктория Анатольевна", "Математика", "29.12.2022", 2)  
 student1.get\_info()  
 student1.info\_grade()  
 student1.retake()  
 *println*()  
  
 val student2 = Diplom(60, "Иван Иванович Иванов", "Разработка мобильной игры", "21.07.2022", 4)  
 student2.get\_info()  
 student2.info\_grade()  
 student2.retake()  
 student2.info\_pages()  
 *println*()  
  
 val student3 = Homework(5, "Александ Александрович Баринов", "Математика", "30.09.2022", 2)  
 student3.get\_info()  
 student3.info\_grade()  
 student3.retake()  
 student3.info\_complexy()  
 *println*()  
  
}

//Milk\_product.kt  
package Milk\_product  
//вес указывается в граммах  
open class Milk\_product (var name: String, var weight : Double, var fat\_content:Double, var producer:String, var price:Double ){  
  
 open fun weigt\_info(){  
 *println*("Вес молочного продукта состовляет:")  
 val kg: Double = weight / 1000  
 *println*("$weight грамм или $kg килограмм")  
 }  
  
 open fun price\_info(){  
 val gramm\_price = weight / price  
 *println*("Цена одного грамма молочного изделия равна $gramm\_price рублей")  
 }  
  
 open fun get\_info(){  
 *println*("Информация о молочном продукте:")  
 *println*("Название: $name\nВес: $weight грамм\nЖирность: $fat\_content%\nПроизводитель: $producer\nЦена: $price")  
 }  
}

//Cheese.kt  
import Milk\_product.\*  
class Cheese(var whoose\_milk : String, var consistency:String, var color: String, var chees\_name:String, var chees\_weight:Double, var chees\_fat\_content:Double, var chees\_prosucer:String, var chees\_price:Double):  
 Milk\_product(chees\_name, chees\_weight, chees\_fat\_content, chees\_prosucer, chees\_price) {  
 //whoose\_milk чьё молоко использовалось для приготовления  
  
 override fun weigt\_info() {  
 *println*("Вес сыра состовляет:")  
 val kg: Double = weight / 1000  
 *println*("$weight грамм или $kg килограмм")  
 }  
  
 override fun price\_info() {  
 val gramm\_price = weight / price  
 *println*("Цена одного грамма сыра равна $gramm\_price рублей")  
 }  
  
 override fun get\_info() {  
 super.get\_info()  
 *println*("Для изготовления cыра используется молоко: $whoose\_milk\nКонсистенция: $consistency\nЦвет: $color")  
 }  
}

//Milk.kt  
import Milk\_product.\*  
class Milk (var view : String, var cooking\_method:String, var milk\_name:String, var milk\_weight:Double, var milk\_fat\_content:Double, var milk\_prosucer:String, var milk\_price:Double):  
 Milk\_product(milk\_name, milk\_weight, milk\_fat\_content, milk\_prosucer, milk\_price){  
 private var plotnoct : Double = 0.971  
 public fun P() : Double{  
 return plotnoct  
 }  
  
 override fun weigt\_info() {  
 val pl:Double = P();  
 val litr = milk\_weight \* pl  
 *println*("Молоко имеет обьем $litr литр(а, ов)")  
 }  
 override fun price\_info() {  
 val gramm\_price = weight / price  
 *println*("Цена одного грамма молока равна $gramm\_price рублей")  
 }  
  
 override fun get\_info() {  
 super.get\_info()  
 *println*("Вид молока: $view\nМетод приготовления: $cooking\_method")  
 }  
  
}

//Exam.kt  
package Exam  
import java.util.\*  
open class Exam (var name:String, var lesson:String, var data:String, var grade:Int ){  
  
 open fun get\_info(){  
 *println*("\nИнформация")  
 *println*("Имя студента $name\nПредмет: $lesson\nДата: $data\nОценка: $grade")  
 }  
  
 open fun info\_grade(){  
 *println*("\nПодробное описание оценки:")  
 when(grade){  
 0 -> *println*("Прогул")  
 1 -> *println*("Экзамен не сдан")  
 2 -> *println*("Экзамен не сдан")  
 3 -> *println*("Экзамен сдан удовлетворительное")  
 4 -> *println*("Экзамен сдан хорошо")  
 5 -> *println*("Экзамен сдан отлично!")  
 }  
 }  
  
 open fun retake(){  
 *println*("\nСудьба студента:")  
 if(grade < 3){  
 val ret:Int = Random().nextInt((3- 1) + 1)  
 when(ret){  
 0 -> *println*("Пересдачи нет. Студент отчислен")  
 1 -> *println*("Студент может пойти на пересдачу")  
 2 -> *println*("Пересдачи нет. Студент отчислен")  
  
 }  
 }  
 else if(grade == 3) *println*("Студент остался без степендии")  
 else{  
 *println*("Студент счастлив. У него есть стипендия!")  
 }  
 }  
}

//Diplim.kt  
import Exam.\*  
import java.util.\*  
  
//open class Exam (var name:String, var lesson:String, var data:String, var grade:Int ){  
class Diplom(var pages:Int, var diplom\_student\_name: String, var diplom\_tema:String, var diplom\_data: String, var diplom\_grade:Int ) : Exam(diplom\_student\_name, diplom\_tema, diplom\_data, diplom\_grade) {  
  
 override fun get\_info() {  
 *println*("\nИнформация")  
 *println*("Имя студента $name\nТема: $lesson\nДата: $data\nОценка: $grade\nКоличество страниц: $pages")  
 }  
  
 override fun info\_grade() {  
 *println*("\nПодробное описание оценки:")  
 when(grade){  
 0 -> *println*("Студент отчислился прежде чем защитить диплом")  
 1 -> *println*("Диплом защищен отвратителен")  
 2 -> *println*("Диплом защищен ужасно")  
 3 -> *println*("Диплом защищен удовлетворительное")  
 4 -> *println*("Диплом защищен хорошо")  
 5 -> *println*("Диплон защищен отлично!")  
 }  
 }  
  
 override fun retake() {  
 *println*("Будущее студента:")  
 if(grade < 3){  
 val ret:Int = Random().nextInt((3- 1) + 1)  
 when(ret){  
 0 -> *println*("Студент не найдет работу без диплома")  
 1 -> *println*("Студент не найдет работу без диплома")  
 2 -> *println*("Студент откроет свое дело и станет богатым")  
  
 }  
 }  
 else{  
 *println*("Студент найдет работу")  
 }  
 }  
 fun info\_pages(){  
 if(pages == 0) *println*("У студента нет дипломной работы")  
 else if((pages > 0) && (pages < 50)) *println*("Дипломаная работа очень маленькая")  
 else if((pages > 50) && (pages < 100)) *println*("Дипломная работа среднего размера")  
 else *println*("Дипломная работа огромная")  
 }  
}

//Homework.kt  
import Exam.\*  
import java.util.\*  
  
class Homework(var complexity:Int, var hw\_student\_bame:String, var hw\_lesson:String, var hw\_data:String, var hw\_grade:Int, ) : Exam(hw\_student\_bame, hw\_lesson, hw\_data, hw\_grade){  
  
 override fun get\_info() {  
 *println*("\nИнформация")  
 *println*("Имя студента $name\nПредмет: $lesson\nДата: $data\nОценка: $grade\nУровень сложность: $complexity из 10")  
 }  
  
 override fun info\_grade() {  
 *println*("\nПодробное описание оценки:")  
 if(grade < 3) *println*("Студент не сделал домашнее задание")  
 else if(grade == 3) *println*("Студент сделал домашнее задание удовлетворительно")  
 else if(grade == 4) *println*("Студент сделал домашнее задание хорошо")  
 else *println*("Студент сделал домашнее задание отлично!")  
 }  
  
 override fun retake() {  
 *println*("Будущее студента:")  
 if(grade < 3){  
 val ret:Int = Random().nextInt((3- 1) + 1)  
 when(ret){  
 0 -> *println*("Студент сделает домашнее задание позже")  
 1 -> *println*("Студент сделает домашнее задание позже")  
 2 -> *println*("Преподаватель поставил отрицательную оценку в журан. Студент будет плакать и не сможет доделать задание")  
  
 }  
 }  
 else{  
 *println*("Студент счастлив что смог сделать задание")  
 }  
 }  
 fun info\_complexy(){  
 *println*("Сложность задания:")  
 if (complexity < 4) *println*("Это задание простое")  
 else if((complexity > 4) && (complexity < 8)) *println*("Это задание среднего уровня сложности")  
 else *println*("Это очень трудное задание")  
 }  
}

**Вывод**

Благодаря этой практической я научилась лучше работать с класса