Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по программе «**Практическое занятие 12**»

Выполнил: Максимов Кирилл Эдуардович

Группа: ПР-21

Преподаватель: Мирошниченко Г.В

2023

**Задание:**

**Входные и выходные данные**

**Листинг программы (если есть)**

**Скриншот экранов**

**Тестовые ситуации**

**Вывод**

import kotlinx.coroutines.\*  
  
suspend fun main() = coroutineScope<Unit> {  
 val taskOne = TaskOne()  
 var count: Int  
  
 do {  
 *print*("Введите кол-во повторений: ")  
 count = *readln*()!!.*toInt*()  
 } while (count < 0)  
 for (i in 1..count) {  
 GlobalScope.*launch* {  
 taskOne.InputName()  
 taskOne.InputAge()  
 delay(3000L)  
 }  
 taskOne.printGroup()  
 *runBlocking* {  
 delay(10000L)  
 }  
 }  
 var taskTwo = TaskTwo()  
 try {  
 var countRep: Int  
 do {  
 *print*("Введите кол-во репрозиториев: ")  
 countRep = *readln*()!!.*toInt*()  
 } while (countRep < 0)  
  
 for (i in 1..countRep) {  
 GlobalScope.*launch* {  
 taskTwo.inputDate()  
 delay(10000L)  
 }  
 *runBlocking* {  
 taskTwo.IsBlock(false)  
 delay(10000L)  
 }  
 }  
 *println*("${taskTwo.listRep.*sortedBy* {countRep}}")  
 *println*("${taskTwo.listRep.*maxBy* {countRep}}")  
 }  
 catch (ex: Exception)  
 {  
 *println*(ex.message)  
 }  
}

Входные данные:

var name = strting

var age = int

Выходные данные:

var name = strting

var age = int

import kotlinx.coroutines.\*  
  
class TaskOne {  
 suspend fun main(){  
 GlobalScope.*launch* {  
 delay(1000L)  
 InputAge()  
 }  
 InputName()  
 *runBlocking* {  
 delay(2000L)  
 }  
 }  
 fun InputName()  
 {  
 try {  
 *print*("Введите имя: ")  
 var name = *readln*()!!.toString()  
 *println*("Привет ${name}")  
 }  
 catch (ex: Exception)  
 {  
 *println*(ex.message)  
 }  
 }  
  
 fun InputAge()  
 {  
 try{  
 var age: Int  
 do {  
 *print*("Введите возраст: ")  
 age = *readln*()!!.*toInt*()  
 }while(age < 0)  
  
 *println*("Ваш возраст ${age}")  
 }  
 catch (ex: Exception)  
 {  
 *println*(ex.message)  
 }  
 }  
 fun printGroup()  
 {  
 *println*("\nВаша группа Пр-21")  
 }  
}

Входные данные:

var login = strting

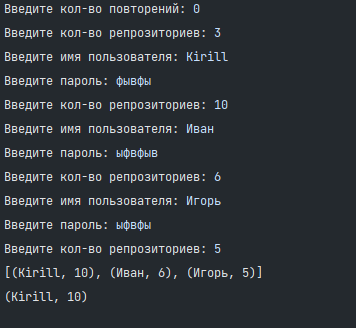
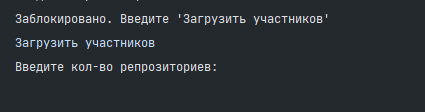
var password = strting

var countRep = Int

Выходные данные:

mutableSetOf(login, countRep)

import kotlinx.coroutines.\*  
  
class TaskTwo {  
 var block = false  
 var listRep = *mutableSetOf*<Pair<String, Int>>()  
 suspend fun main() {  
 GlobalScope.*launch* {  
 }  
 inputDate()  
 *runBlocking* {  
 delay(5000L)  
 }  
 }  
  
 fun inputDate() {  
 try {  
 var countRep: Int  
 *print*("Введите имя пользователя: ")  
 var login = *readln*()!!.toString()  
 *print*("Введите пароль: ")  
 var password = *readln*()!!.toString()  
 do {  
 *print*("Введите кол-во репрозиториев: ")  
 countRep = *readln*()!!.*toInt*()  
 } while (countRep < 0)  
 listRep.add(login *to* countRep)  
 } catch (ex: Exception) {  
 *println*(ex.message)  
 }  
 }  
  
 fun IsBlock(block: Boolean) {  
 while (block == false) {  
 *println*("Заблокировано. Введите 'Загрузить участников'")  
 var input = *readLine*()!!  
 if (input == "Загрузить участников") {  
 break  
 }  
 }  
 }  
}





**Вывод**: Научился пользоваться методами библиотеки coroutines в языке Kotlin.