Каріна Зубко 4 варіант

Лаболаторна 10-11:

abstract class LibraryItem(val title: String)

case class Book(author: String, override val title: String) extends LibraryItem(title)

case class Magazine(override val title: String, month: Int, year: Int) extends LibraryItem(title)

case class Newspaper(override val title: String, date: String) extends LibraryItem(title)

//Zubko

object Library extends App {

def isPeriodic(item: LibraryItem): Boolean = item match {

case \_: Magazine | \_: Newspaper => true

case \_ => false

}

def getByTitle(title: String, items: List[LibraryItem]): List[LibraryItem] =

items.filter(\_.title == title)

def getByMonth(month: Int, year: Int, items: List[LibraryItem]): List[LibraryItem] =

items.filter {

case magazine: Magazine => magazine.month == month && magazine.year == year

case \_ => false

}

def getByMonths(months: List[Int], year: Int, items: List[LibraryItem]): List[LibraryItem] =

items.filter {

case magazine: Magazine => months.contains(magazine.month) && magazine.year == year

case \_ => false

}

def getAuthors(items: List[LibraryItem]): List[String] =

items.collect { case book: Book => book.author }

val items = List(

Book("Author1", "Book1"),

Magazine("Magazine1", 6, 2021),

Newspaper("Newspaper1", "2021-06-01"),

Book("Author2", "Book2"),

Magazine("Magazine2", 7, 2021)

)

println(s"Periodic items: ${items.filter(isPeriodic)}")

println(s"Items with title 'Magazine1': ${getByTitle("Magazine1", items)}")

println(s"Periodicals published in June 2021: ${getByMonth(6, 2021, items)}")

println(s"Periodicals published in June and July 2021: ${getByMonths(List(6, 7), 2021, items)}")

println(s"Authors of publications: ${getAuthors(items)}")

}

