# Задание 4

## Условие

5. В библиотеке хранятся книги, газеты и журналы.

Книга характеризуется именем автора и названием;

журнал - названием, месяцем и годом выпуска;

газета - названием и датой выпуска. База данных представляет собой список этих объектов.

Разработайте тип данных, представляющий объекты библиотечного хранения. Определите следующие функции:

1. isPeriodic, проверяющую, что ее аргумент является перио-дическим изданием.
2. getByTytle, выбирающая из списка объектов хранения (базы данных) объекты с указанным названием.
3. getByMonth, выбирающая из базы данных периодические из-дания, выпущенные в указанный месяц и указанный год (за-метьте, что газеты выходят несколько раз в месяц).
4. getByMonths, действующая так же, как и предыдущая, но при-нимающая список месяцев.
5. get Authors, возвращающая список авторов изданий, храня¬щихся в базе данных.

## Код

-- Задание 4

data MyLibraryType = Book String String | Magazine String Int Int | NewsPaper String Int Int Int deriving (Eq,Show)

-- Проверяет, что аргумент является периодическим изданием.

isPeriodic :: MyLibraryType -> Bool

isPeriodic (Book \_ \_) = False

isPeriodic (Magazine \_ \_ \_) = True

isPeriodic (NewsPaper \_ \_ \_ \_) = True

-- Проверка на наличие заданного названия в базе данных

checkTitle :: MyLibraryType -> String -> Bool

checkTitle (Book t a) s = (s == t)

checkTitle (Magazine t y m) s = (s == t)

checkTitle (NewsPaper t y m d) s = (s == t)

-- Выбирает из списка объектов БД объекты с указанным названием.

getByTitle :: [MyLibraryType] -> String -> [MyLibraryType]

getByTitle [] \_ = []

getByTitle (o:os) s | (checkTitle o s) = o : getByTitle os s

| otherwise = getByTitle os s

getMon :: MyLibraryType -> Int

getMon (Book \_ \_) = 0

getMon (Magazine \_ y m) = m

getMon (NewsPaper \_ y m d) = m

getYear :: MyLibraryType -> Int

getYear (Book \_ \_) = 0

getYear (Magazine \_ y m) = y

getYear (NewsPaper \_ y m d) = y

-- Выбирает из БД периодические издания, выпущенные в указанный месяц и указанный год

getByMonth :: [MyLibraryType] -> Int -> Int -> [MyLibraryType]

getByMonth [] \_ \_ = []

getByMonth (o:os) y m = if ((getMon o) == m) && ((getYear o) == y) then o : (getByMonth os y m) else (getByMonth os y m)

getByMo :: [MyLibraryType] -> Int -> [MyLibraryType]

getByMo [] \_ = []

getByMo (o:os) m = if ((getMon o) == m) then o : (getByMo os m) else (getByMo os m)

-- Действует, как и предыдущая, но принимающая список месяцев.

getByMonths :: [MyLibraryType] -> [Int] -> [MyLibraryType]

getByMonths [] \_ = []

getByMonths \_ [] = []

getByMonths o (m:ms) = (getByMo o m) ++ getByMonths o ms

-- Возвращает список авторов изданий, хранящихся в базе данных.

getAuthors :: [MyLibraryType] -> [String]

getAuthors [] = []

getAuthors ((Book a \_):os) = a : getAuthors os

getAuthors ((Magazine \_ \_ \_):os) = getAuthors os

getAuthors ((NewsPaper \_ \_ \_ \_):os) = getAuthors os

## Скриншот

