

```

import Data.Time.Calendar
import Data.Time.Calendar.WeekDate
import Data.Char
import System.IO
import Data.List

--header of the calendar--
getheader::Integer->Int->[Char]
getheader x xs = concat[welcome,printLine, show
    x,printLine,month,printLine,days,printLine]
    where days="Sun|Mon|Tus|Wed|Thur|Fri|Sat"
          welcome="--- Welcome to Haskell Calendar ---"
          month=months!!(xs-1)
          where
            months=
              ["JANUARY","FEBRUARY","MARCH","APRIL","MAY","JUNE"
              ,"JULY","AUGUST","SEPTEMBER","OCTOBER","NOVEMBER"
              ,"DECEMBER"]

--grind and format--
printLine = "\n"++concat [" " | w <- [0..6]]++"\n"
printLineDay="\n"++concat ["----" | w <- [0..6]]++"\n"

--test--
calendar:: Integer -> Int -> IO()
calendar year month= putStrLn $ getheader year month
                    --putStrLn $ getbody year month

--main function--
cal3:: Integer -> Int -> IO()
cal3 year month= putStrLn $ (addbody x y)
    where x = getheader year month
          y = getbody year month

--add head and body--
addbody:: [Char]->[Char]->[Char]
addbody x y = concat[x,y]

--get a tuple of days in a month--
daytuple:: Integer -> Int ->[[Int]]
daytuple year month= splitlist 7 (generate firstday total)
    where total= gregorianMonthLength year month
          firstday = case firstday of 7 ->0
                                     _ ->firstday
    where firstday=digitToInt(last(showWeekDate
        (fromGregorian year month 01)))

--generate a tuple of days in a month--
generate::Int->Int->[Int]
generate firstday total = take before (repeat 0) ++ [1..total]++take after
    (repeat 0)
    where before = firstday
          after = 35 - firstday-total

```

```

--split a tuple of days in weeks--
splitlist::Int->[a]->[[a]]
splitlist _ [] = []
splitlist n xs = as : splitlist n bs
                  where (as,bs) =splitAt n xs

--test part2--
cal2:: Integer -> Int -> IO()
cal2 year month= putStrLn $ getbody year month

--body of calendar--
getbody::Integer->Int->[Char]
getbody year month = concat[show see,printLineDay,show see2,printLineDay,show
  see3,printLineDay,show see4,printLineDay,show see5,printLineDay]
  where see = list1!!0
        where list1=daytuple year month
  see2=list1!!1
        where list1=daytuple year month
  see3=list1!!2
        where list1=daytuple year month
  see4=list1!!3
        where list1=daytuple year month
  see5=list1!!4
        where list1=daytuple year month

```