QMM\_Assignment2

library(lpSolveAPI)  
setwd("C:/Users/nihar/OneDrive/Desktop/Fall Assignments/QMM/Assignment 2")

a linear program with 9 decision variables and 0 constraints

lp <- make.lp(0,9, verbose = "neutral")  
lp

## Model name:   
## a linear program with 9 decision variables and 0 constraints

Add the constraints

add.constraint(lp, c(1,1,1,0,0,0,0,0,0), "<=", 750 )  
add.constraint(lp, c(0,0,0,1,1,1,0,0,0), "<=", 900)  
add.constraint(lp, c(0,0,0,0,0,0,1,1,1), "<=", 450)  
add.constraint(lp, c(20,15,12,0,0,0,0,0,0), "<=", 13000)  
add.constraint(lp, c(0,0,0,20,15,12,0,0,0), "<=", 12000)  
add.constraint(lp, c(0,0,0,0,0,0,20,15,12), "<=", 5000)  
add.constraint(lp, c(1,1,1,0,0,0,0,0,0), "<=", 900)  
add.constraint(lp, c(0,0,0,1,1,1,0,0,0), "<=", 1200)  
add.constraint(lp, c(0,0,0,0,0,0,1,1,1), "<=", 750)  
add.constraint(lp, c(6, 6, 6, -5, -5, -5, 0, 0, 0), "=", 0)  
add.constraint(lp, c( 3, 3, 3, 0, 0, 0, -5, -5, -5), "=", 0)

Create objective function. We need maximum profit so change sense to max

set.objfn(lp, c(420,360,300,420,360,300,420,360,300))  
lp.control(lp, sense='max')

## $anti.degen  
## [1] "none"  
##   
## $basis.crash  
## [1] "none"  
##   
## $bb.depthlimit  
## [1] -50  
##   
## $bb.floorfirst  
## [1] "automatic"  
##   
## $bb.rule  
## [1] "pseudononint" "greedy" "dynamic" "rcostfixing"   
##   
## $break.at.first  
## [1] FALSE  
##   
## $break.at.value  
## [1] 1e+30  
##   
## $epsilon  
## epsb epsd epsel epsint epsperturb epspivot   
## 1e-10 1e-09 1e-12 1e-07 1e-05 2e-07   
##   
## $improve  
## [1] "dualfeas" "thetagap"  
##   
## $infinite  
## [1] 1e+30  
##   
## $maxpivot  
## [1] 250  
##   
## $mip.gap  
## absolute relative   
## 1e-11 1e-11   
##   
## $negrange  
## [1] -1e+06  
##   
## $obj.in.basis  
## [1] TRUE  
##   
## $pivoting  
## [1] "devex" "adaptive"  
##   
## $presolve  
## [1] "none"  
##   
## $scalelimit  
## [1] 5  
##   
## $scaling  
## [1] "geometric" "equilibrate" "integers"   
##   
## $sense  
## [1] "maximize"  
##   
## $simplextype  
## [1] "dual" "primal"  
##   
## $timeout  
## [1] 0  
##   
## $verbose  
## [1] "neutral"

To identify the variables and constraints, Set the variable names and the constraints set.bounds(lp, lower = c(0, 0, 0, 0, 0, 0, 0, 0, 0), columns = c(1,2,3,4,5,6,7,8,9))

RowNames <- c("Con1", "Con2", "Con3", "storage1", "Storage2", "Storage33", "Sale1", "Sale2", "Sale3", "%C1", "%C2")  
ColNames <- c("Large1", "Medium1", "Small1", "Large2", "Medium2", "Small2", "Large3", "Medium3", "Small3")  
dimnames(lp) <- list(RowNames, ColNames)  
lp

## Model name:   
## a linear program with 9 decision variables and 11 constraints

write.lp(lp, filename = "QMMAssignment2.lp", type = "lp")  
solve(lp)

## [1] 0

get.objective(lp)

## [1] 696000

get.variables(lp)

## [1] 516.6667 177.7778 0.0000 0.0000 666.6667 166.6667 0.0000 0.0000  
## [9] 416.6667