In [1]:
```
from gurobipy import *
```

In [2]:
```
products = [1, 2, 3]
```

In [3]:
```
lines = [1, 2]
```

In [4]:
```
months = [1, 2, 3, 4, 5, 6]
```

In [5]:
```
demand = [[0, 0, 0], [50, 40, 30], [30, 60, 40], [40, 50, 20], [60, 30, 70], [
20, 30, 40], [45, 55, 30] ]
```

In [9]:
```
holding = [0.5, 0.35, 0.45]
```

In [10]:
```
production = [[10, 8, 15], [12, 6, 10]]
```

In [11]:
```
switching = [[200, 180, 300], [250, 200, 174]]
```

In [12]:
```
rate = [[40, 90], [60, 70], [80, 60]]
```

In [13]:
```
Productionmodel = Model('Production model')
```

```
Using license file C:\Users\Mustapha\gurobi.lic
Academic license - for non-commercial use only
```

In [14]:
```
Zijm = Productionmodel.addVars(products, lines, months, name = "z_ijm", vtype
= GRB.BINARY)
```

In [15]:
```
Tijm = Productionmodel.addVars(products, lines, months, name = "t_ijm", vtype
= GRB.BINARY)
```

In [16]:
```
numProduced_ijm = Productionmodel.addVars(products, lines, months, lb = 0.0, n
ame = "produced_ijm", vtype = GRB.CONTINUOUS)
```

In [17]:
```
numStored_im = Productionmodel.addVars(products, months, lb = 0.0, name = "sto
red_im", vtype = GRB.CONTINUOUS)
```

In [18]:
```
Productionmodel.addConstrs(quicksum(Zijm[i, j, m] for i in products) <= 1 for
j in lines for m in months)
```

Out[18]:
```
{(1, 1): <gurobi.Constr *Awaiting Model Update*>,
 (1, 2): <gurobi.Constr *Awaiting Model Update*>,
 (1, 3): <gurobi.Constr *Awaiting Model Update*>,
 (1, 4): <gurobi.Constr *Awaiting Model Update*>,
 (1, 5): <gurobi.Constr *Awaiting Model Update*>,
 (1, 6): <gurobi.Constr *Awaiting Model Update*>,
 (2, 1): <gurobi.Constr *Awaiting Model Update*>,
 (2, 2): <gurobi.Constr *Awaiting Model Update*>,
 (2, 3): <gurobi.Constr *Awaiting Model Update*>,
 (2, 4): <gurobi.Constr *Awaiting Model Update*>,
 (2, 5): <gurobi.Constr *Awaiting Model Update*>,
 (2, 6): <gurobi.Constr *Awaiting Model Update*>}
```

In [22]:
```
obj = quicksum(holding[p]*Tijm[p][l][m] + production[l][p]*numproduced_ijm[p][
l][m] + switching[l][p]*numstored_im[p][m] for p in products for l in lines fo
r m in months)
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-22-41c6c29cbdbd> in <module>()
----> 1 obj = quicksum(holding[p]*Tijm[p][l][m] + production[l][p]*numproduce
d_ijm[p][l][m] + switching[l][p]*numstored_im[p][m] for p in products for l i
n lines for m in months)

gurobi.pxi in gurobipy.quicksum()

<ipython-input-22-41c6c29cbdbd> in <genexpr>((p,))
----> 1 obj = quicksum(holding[p]*Tijm[p][l][m] + production[l][p]*numproduce
d_ijm[p][l][m] + switching[l][p]*numstored_im[p][m] for p in products for l i
n lines for m in months)

KeyError: 1
```

In [24]:
```
Productionmodel.addConstrs(numProduced_ijm[p][l][m] <= Zijm[p, l, m]*rate[p][l
] for p in products for j in lines for m in months)
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-24-f6f5c167e76f> in <module>()
----> 1 Productionmodel.addConstrs(numProduced_ijm[p][l][m] <= Zijm[p, l, m]*
rate[p][l] for p in products for j in lines for m in months)

model.pxi in gurobipy.Model.addConstrs()

<ipython-input-24-f6f5c167e76f> in <genexpr>((p,))
----> 1 Productionmodel.addConstrs(numProduced_ijm[p][l][m] <= Zijm[p, l, m]*
rate[p][l] for p in products for j in lines for m in months)

KeyError: 1
```

In [26]:
```
Productionmodel.addConstrs(numProduced_ijm[p][l][m] + numstored_im[p][m] - num
stored_im[p][m] == demand[p][m+1] for p in products for j in lines for m in mo
nths)
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-26-c82c68c55bb7> in <module>()
----> 1 Productionmodel.addConstrs(numProduced_ijm[p][l][m] + numstored_im[p]
[m] - numstored_im[p][m] == demand[p][m+1] for p in products for j in lines f
or m in months)

model.pxi in gurobipy.Model.addConstrs()

<ipython-input-26-c82c68c55bb7> in <genexpr>((p,))
----> 1 Productionmodel.addConstrs(numProduced_ijm[p][l][m] + numstored_im[p]
[m] - numstored_im[p][m] == demand[p][m+1] for p in products for j in lines f
or m in months)

KeyError: 1
```

In [29]:
```
Productionmodel.addConstrs(Tijm[p][l][m] >= Zijm[p][l][m+1] - Zijm[p][l][m] fo
r p in products for l in lines for m in months)
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-29-9556c4e9fb59> in <module>()
----> 1 Productionmodel.addConstrs(Tijm[p][l][m] >= Zijm[p][l][m+1] - Zijm[p]
[l][m] for p in products for l in lines for m in months)

model.pxi in gurobipy.Model.addConstrs()

<ipython-input-29-9556c4e9fb59> in <genexpr>((p,))
----> 1 Productionmodel.addConstrs(Tijm[p][l][m] >= Zijm[p][l][m+1] - Zijm[p]
[l][m] for p in products for l in lines for m in months)

KeyError: 1
```

In [33]:
```
Productionmodel.addConstrs((Tijm[p][l][m],Zijm[p][l][m] == {0,1}) for p in pro
ducts for l in lines for m in months)
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-33-d9245e482a8c> in <module>()
----> 1 Productionmodel.addConstrs((Tijm[p][l][m],Zijm[p][l][m] == {0,1}) for
p in products for l in lines for m in months)

model.pxi in gurobipy.Model.addConstrs()

<ipython-input-33-d9245e482a8c> in <genexpr>((p,))
----> 1 Productionmodel.addConstrs((Tijm[p][l][m],Zijm[p][l][m] == {0,1}) for
p in products for l in lines for m in months)

KeyError: 1
```

In [34]:
```
Productionmodel.addConstrs((numProduced_ijm[p][l][m], numstored_im[p][m] >= 0)
for p in products for l in lines for m in months)
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-34-b6c6b30d2bf3> in <module>()
----> 1 Productionmodel.addConstrs((numProduced_ijm[p][l][m], numstored_im[p]
[m] >= 0) for p in products for l in lines for m in months)

model.pxi in gurobipy.Model.addConstrs()

<ipython-input-34-b6c6b30d2bf3> in <genexpr>((p,))
----> 1 Productionmodel.addConstrs((numProduced_ijm[p][l][m], numstored_im[p]
[m] >= 0) for p in products for l in lines for m in months)

KeyError: 1
```

In [35]:
```
Productionmodel.addConstrs((Zijm[p][l][0] >= 0) for p in products for l in lin
es for m in months)
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-35-af1965bcebf8> in <module>()
----> 1 Productionmodel.addConstrs((Zijm[p][l][0] >= 0) for p in products for
l in lines for m in months)

model.pxi in gurobipy.Model.addConstrs()

<ipython-input-35-af1965bcebf8> in <genexpr>((p,))
----> 1 Productionmodel.addConstrs((Zijm[p][l][0] >= 0) for p in products for
l in lines for m in months)

KeyError: 1
```

In [38]:
```
Productionmodel.addConstrs(numStored_im[1][0] == 55, numStored_im[2][0] == 75,
numStored_im[3][0] == 60)
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-38-5e0c293679ce> in <module>()
----> 1 Productionmodel.addConstrs(numStored_im[1][0] == 55, numStored_im[2][
0] == 75, numStored_im[3][0] == 60)

KeyError: 1
```

In [1]:
```
obj = quicksum(holding[p]*Tijm[p][l][m] + production[l][p]*numproduced_ijm[p][
l][m] + switching[l][p]*numstored_im[p][m] for p in products for l in lines fo
r m in months)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-1-41c6c29cbdbd> in <module>()
----> 1 obj = quicksum(holding[p]*Tijm[p][l][m] + production[l][p]*numproduce
d_ijm[p][l][m] + switching[l][p]*numstored_im[p][m] for p in products for l i
n lines for m in months)

NameError: name 'quicksum' is not defined
```

In [ ]:
```

```