name
value
sets
summation
for every

Sets:
$C \in Course = \{ML, OP\}$
$B \in Block = \{SatMorn, SatAfter, SunAfter\}$

Inputs:
$Base_C = \{80, 50\}$
$GradePerBlock_C = \{10, 20\}$
$MaxGrade_C = \{100\}$

Decision Var:
Let $StudyBlock_{C,B}$ *be the study decisions, specifically:*

$$StudyBlock_{C,B} = \begin{cases} 1 & \text{if we decide to study course } C \text{ for Block } B \\ 0 & \text{otherwise} \end{cases}$$

Formulation: Summation

$$Max\ Grade = \sum_{C \in Course} Base_C + \left(\sum_{B \in Block} StudyBlock_{C,B} \times GradePerBlock_C \right)$$

$S.t.$

$$Base_C + \sum_{B \in Block} StudyBlock_{C,B} \times GradePerBlock_C \leq MaxGrade_C \quad \forall C \in Course$$

$$\sum_{C \in Course} StudyBlock_{C,B} \leq 1 \quad \forall B \in Block$$

```
proc optmodel;

set <str> Course;
read data LinearProblem into Course = [Course];
set <str> Block = {'SatMorn', 'SatAfter', 'SunAfter'};

num Base{Course};
num GradePerHr{Course};
num CourseMax{Course};
read data LinearProblem into [Course] Base = Base GradePerHr =
GradePerBlock CourseMax = CourseMax;

var StudyBlock{Course, Block} binary;

Max Grade= sum{C in Course}(Base[C] + sum{B in
Block}(GradePerHr[C] * StudyBlock[C,B]));

Con Perfect{C in Course}: Base[C] + sum{B in
Block}(GradePerHr[C]*StudyBlock[C,B]) <= CourseMax[C];
Con Time{B in Block}: sum{C in Course}(StudyBlock[C,B]) <= 1;

solve with MILP objective Grade;

print Grade;
print StudyBlock;

quit;
```

```
num CourseMax{Course};

read data LinearProblem into [Course] Base = Base GradePerHr =
GradePerBlock CourseMax = CourseMax;
```

	Course	Base	GradePerHr	GradePerBlock	CourseMax
1	ML	80	2	10	100
2	OP	50	5	20	100

1) take value from set "course"
2) go to table "LinearProblem"
3) look up the row w/ primary key "course" under column "course"
4) in same row, find value in column "coursemax"
5) put said value back to numeric "coursemax"