sas

Wajdi Ben Saad
ESSAI,Nov 2015

Wajdi Ben Saad

Conditional logic

If, Then, Else

If, Then, Else

```
IF <condition> THEN <X>;
ELSE <Y>;
```

```
If Score >= 70 Then Grade = 'Passing Grade';
Else Grade = 'Failing Grade';
```

Student	Score	Grade
Jane	75	Passing Grade
Dave	56	Failing Grade
Jack	90	Passing Grade
Sue	68	Failing Grade

Wajdi Ben Saad

If, Then, Else

```
IF <condition> THEN <X>;
ELSE IF <condition2> THEN <Y>;
ELSE <Z>;
```

```
If Score >= 70 Then Grade = 'Passing Grade';
Else If 60 <= Score <= 69 Then Grade = 'Incomplete';
Else Grade = 'Failing Grade';
```

Student	Score	Grade
Jane	75	Passing Grade
Dave	56	Failing Grade
Jack	90	Passing Grade
Sue	68	Incomplete

If, Then, Else

```
If Score >= 90 Then Grade = 'A';
If Score >= 80 Then Grade = 'B';
If Score >= 70 Then Grade = 'C';
If Score >= 60 Then Grade = 'D';
If Score < 60 Then Grade = 'F';
```

Student	Score
Jane	75
Dave	56
Jack	90
Sue	68

Wajdi Ben Saad

Operators

Arithmetic operators

Arithmetic	Symbol	Example
Addition	+	`Xplus = 4+2;`
Subtraction	−	`Xminus = 4-2;`
Multiplication	*	`Xmult = 4*2;`
Division	/	`Xdiv = 4/2;`
Exponents	**	`Xexp = 4**2;`
Negative numbers	−	`Xneg = -2;`

Comparison operators

Logical comparison	Mnemonic	Symbol
Equal to	EQ	=
Not equal to	NE	^= or ~=
Less than	LT	<
Less than or equal to	LE	<=
Greater than	GT	>
Greater than or equal to	GE	>=

Note: <> also used for not equal to, but only in PROC SQL

Logical operators

Boolean operator
And
Or
Not

```
data AgeGroup; set test_data;
If age lt 18 and gender = 'F' then group = 'Minor - Female';
else if age lt 18 and gender = 'M' then group = 'Minor - Male';
else if age ge 18 and gender = 'F' then group = 'Adult - Female';
else if age ge 18 and gender = 'M' then group = 'Adult - Male';
run;
```

```
data ORAdd; set Providers;
length Location $ 15;
If Home_State = 'OR' or Work_State = 'OR' then Location = 'Oregon';
else Location = 'Out of State';
run;
```

Wajdi Ben Saad