First tutorial:

**Exercises for students to do: define input masks for the following**

(TA: describe what the input mask is supposed to do and then given students a few minutes to specify it. Don’t just immediately give them the answer! Let them try to work it out their own first).

* Feet component of height:
	+ Format: A single digit followed by a single quote e.g. 5'
	+ Mask: 0\'
* Inches component of height:
	+ Format: One or two digits followed by double quotes e.g. 8" (the feet and height value are generally specified together e.g. 5' 8")
	+ Mask: 90\"
* Visa number:
	+ Format: 4 digits<SP>4 digits<SP>4 digits<SP>4 digits
	+ Mask: 0000\ 0000\ 0000\ 0000
* Credit card expiry number:
	+ 1 or 2 digits<forward slash>2 digits
	+ Mask: 09\/00
* ERD representation of database tables: exert of my database slides (remind them of the notation).
* Show them how to ‘draw’ an ERD in PowerPoint and save it as a ‘.png’ image
* My notation from lecture notes below



* **Employees table (tracks information about individual employees)**
	+ SIN (PK)
	+ LastName
	+ FirstName
	+ Address
	+ Etc.
* **Departments table**
	+ DepartmentID (PK)
	+ DepartmentName
	+ Budget
* ERD of above example (key and one non-key field)



* Tell them now that if they just submit a screenshot from Access for their ERD they will get NO MARKS

**Example database for this week**



**Exercise for students to do: draw an ERD for the example database**

Tell them the above form is incorrect, draw it in the correct form using PowerPoint. To reduce the amount of repetitive work just have them specify the following:

* Two entities: Provinces and Employees
* Use the proper notation to specify the primary key and any non-key field
* The multiplicity
* To make the exercise more valuable you can take few minutes to check their work (Make sure you tell them when they get it wrong! The above diagram has errors for the primary keys and the ‘many’ part of the relationship.

Second tutorial:

**Validation rules vs. Input masks**

Input masks are rules to specify the format of data, validation rules can do that and more (e.g., verify that data falls within a certain range). Sometimes a format rule may be slightly easier to define with an input mask which is why input masks exist.

**Validation rules to show them from the example database**:

**Pay: version1**

Data requirement:

Pay cannot be negative

Validation rule:

>=0

**Pay: version2**

Data requirement:

Pay cannot exceed $100

Validation rule:

<=100

**Pay: version3 (solution in database)**

Data requirement:

Rate must be in the range from $10 - $100

Validation rule:

>=10 And <=100

**Birthdate: version 1**

Data requirement:

Age cannot be greater than that of the oldest person (say it’s 118 years). Based on the year 2016 that would mean that the year of birth cannot be earlier than the end of1898.

Validation rule (note that dates in Access must be enclosed in number signs)

>= “#1/1/1898#

**Birthdate: version 2 (solution in the database)**

Data requirement:

Date must be between Jan 1 1900 and December 31 2007

Validation rule:

>=#1900-01-01# And <=#2007-12-31#

**Birthdate: version 2**

Data requirement:

Date of birth cannot be in the future

Validation rule:

<=Now()

**Email: version 1**

Format requirement:

Must be in the form <at least one alpha> <any number of characters> at sign <at least one alpha> <any number of characters> and end in .ca

Validation rule:

Like "[A-Z]\*@[A-Z]\*.ca"

**Email: version 2**

Format requirement:

Must be in the form <at least one alpha> <any number of characters> at sign <at least one alpha> <any number of characters> and end in one of the following .ca, .com, .org

**Wonky web address:**

Format requirement:

Must be in the form <www> . <1 numerical digit> <any number of characters> and end in .com

Validation rule:

Like "www.[0-9]\*.com"

Validation rule:

Like "[A-Z]\*@[A-Z]\*.ca" Or

Like "[A-Z]\*@[A-Z]\*.com" Or

Like "[A-Z]\*@[A-Z]\*.org"

**Exercises for students to do: define the validation rule for the following**

(TA: describe what the input mask is supposed to do and then given students a few minutes to specify it. Don’t just immediately give them the answer! Let them try to work it out their own first).

**Exercise 1**: Web address

Format requirement:

Must be in the form www dot <at least one alpha> <any number of any char> .ca

Validation rule:

Like www.[A-Z]\*.ca

**Exercise 2**: Web domain

Format requirement:

 Must end in dot-ca OR dot-com

Validation rule:

Like ".ca" Or ".com"

**Exercise 3**: WebAddressCanadaCommerical

Format requirement:

 www dot <at least one alpha> <any number of any char> .ca OR .com suffix

Validation rule:

Like "www.[A-Z]\*.ca" Or Like www.[A-Z]\*.com

**Using the example database show them how to create graphical interface for entering new data:** (create->form).