CIT 381

Data Types

- data types
- create table statement
- constraints
Data Types

• For each attribute, we need to consider how it is stored.
• Want to store it appropriately and compactly.
• Since our ER Model is exported to MySQL, we need to know about MySQL types.
Numeric Types in MySQL

- TINYINT( )  -128 to 127
- SMALLINT( ) -32768 to 32767
- MEDIUMINT( ) -8388608 to 8388607
- INT( ) -2147483648 to 2147483647
- BIGINT( ) -9223372036854775808 to 9223372036854775807
- FLOAT  A small number with a floating decimal point.
- DOUBLE( , )  A large number with a floating decimal point.
- DECIMAL( , )  A DOUBLE stored as a string , allowing for a fixed decimal point.
Text Types in MySQL

- **CHAR( )** A fixed section from 0 to 255 characters long.
- **VARCHAR( )** A variable section from 0 to 255 characters long.
- **TINYTEXT** A string with a maximum length of 255 characters.
- **TEXT** A string with a maximum length of 65535 characters.
- **BLOB** A string with a maximum length of 65535 characters.
- **MEDIUMTEXT** A string with a maximum length of 16777215 characters.
- **MEDIUMBLOB** A string with a maximum length of 16777215 characters.
- **LONGTEXT** A string with a maximum length of 4294967295 characters.
- **LONGBLOB** A string with a maximum length of 4294967295 characters.
Date/Time Types in MySQL

- DATE YYYY-MM-DD.
- DATETIME YYYY-MM-DD HH:MM:SS.
- TIMESTAMP YYYYMMDDHHMMSS.
- TIME HH:MM:SS.
Foreign Key Constraints

Relationship Editor - Trigger Tab
The table below describes the options and functionality on the Trigger tab of the Relationships Editor:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parent Action</td>
<td>Lets you select a trigger template to maintain referential integrity for different data modification operations: INSERT, UPDATE, and DELETE. Select a trigger for each operation.</td>
</tr>
<tr>
<td>Restrict</td>
<td>Verifies the existence of foreign key values in the parent table's primary key and prevents the insertion, updating, or deleting of data if the values cannot be validated.</td>
</tr>
<tr>
<td>Cascade</td>
<td>Propagates any modification of a primary key value to the corresponding foreign key values in the child table.</td>
</tr>
<tr>
<td>Set Null</td>
<td>Verifies the existence of the foreign key values in the parent table's primary key. If the values cannot be validated, the trigger sets the foreign key values to null in the child table and lets the data modification operation proceed</td>
</tr>
<tr>
<td>Child Action</td>
<td>Lets you select a trigger for Child table modification operations. Select a trigger for each operation.</td>
</tr>
<tr>
<td>Restrict</td>
<td>Verifies the existence of foreign key values in the parent table's primary key and prevents the insertion, updating, or deleting of data if the values cannot be validated.</td>
</tr>
</tbody>
</table>

See Also
Relationship Editor
CREATE TABLE `project` (  
`pname` char(20) default NULL,  
`pnumber` smallint(2) NOT NULL,  
`plocation` char(30) default NULL,  
`dnum` smallint(2) NOT NULL,  
PRIMARY KEY (`pnumber`),  
CONSTRAINT `fk_proj2dept` FOREIGN KEY (`dnum`)  
REFERENCES `department` (`dnumber`)  
ON DELETE CASCADE ON UPDATE CASCADE  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
Foreign Key Example

How do we enter data into these tables? Try it.

Have to populate the crew table first since the ship foreign key can be null.
Maybe Too Constrained

How do we enter data into the following?

Not possible without transactions or something similarly advanced.