

ATTRIBUTE DATA HANDLING

The main topics of that chapter dealt with the relationship between spatial data (maps) and attribute data (tables). This chapter is completely dedicated to the use of tabular data in ILWIS (Figure 3.1).

Before you start it is good understand these things:

- An attribute table is linked to a map through its domain.
- An attribute table can only be linked to maps with a class or ID domain.
- An attribute table may contain several columns. Each of these columns can have a class, ID or value domain (or other special domains, such as color, string, bool etc.)

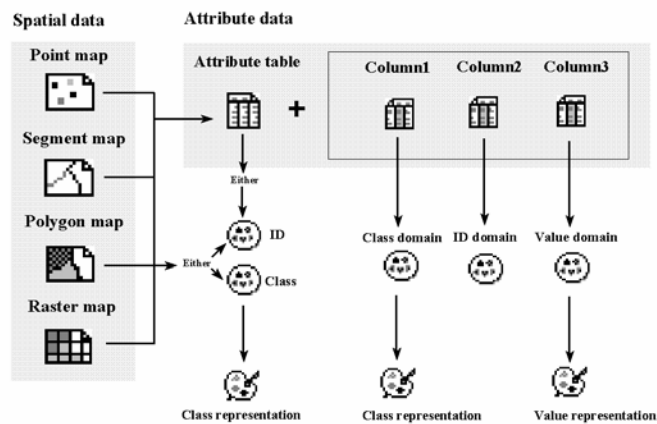


Figure 3.1: Schematic relationship between spatial data (point- segment-, polygon- or raster maps), and attribute data (tables, containing various columns)

In this chapter you will see how you can create a table, how to export or import it from another software package, how to edit it, and various ways to display table data. The main part of this chapter is dedicated to calculating with table data. Before you can start with the exercises, you should start up ILWIS and change to the working subdirectory (:EVMHRAGTRRE\Chapter3), where the data files for this chapter are stored.

- Use the Navigator to go to the working directory \\EVMHRAGTRRE\Chapter3

2.1 The data set: The 8 October 2005 Earthquake affected Tehsil of Pakistan

To import Village, Potwar Circle, Tehsil, Province level attribute data of earthquake affected areas of Pakistan into ILWIS

1. You can find the attribute data/information from various sources or you can collect yourselves.
2. For example you can visit the website of “Relief Information System for Earthquakes – Pakistan”.
3. Download the village level database as Microsoft Excel Spreadsheet from the website address: <http://www.risepak.com/ListVillages.aspx>
4. In Microsoft Excel programme, open the spreadsheet of your database prepared by RICE PAK. Create an abbreviated name for each of the columns in the row of the spreadsheet of the database/information (e.g. “1” for “Activity Date”; “4” for “Tehsil”, etc.) as shown in the figure.
5. Create separate spreadsheet with different worksheets for each Tehsil. For this you can use macro. And also one worksheet for legend for your reference on data description.

1	2	3	4	5	6	7	8	9	10	11	12	13	14
Activity Date	Village Name	Patwar Circle	Tehsil	District	Province	Total	Male Population	Female Population	Total Housing Structures	Number of Pakka Houses	Houses with potable water	Houses with Electricity	Area (acres)
1998 census	1998 census	1998 census	1998 census	1998 census	1998 census	Populati on 1998 census	1998 census	1998 census	1998 census	1998 census	1998 census	1998 census	1998 census

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	3/11/2005	GARHI DOPATT A CITY	MUZAFF ARABAD TEHSIL	MUZAFF ARABAD	MUZAFF ARABAD	AJK	5107	2863	2244	637	485	267	617	
2	4/11/2005	ASHROK E	HEER KOTLI	MUZAFF ARABAD	MUZAFF ARABAD	AJK	610	317	293	80	0	0	74	372
3	4/11/2005	GARHI DOPATT A CITY	MUZAFF ARABAD TEHSIL	MUZAFF ARABAD	MUZAFF ARABAD	AJK	5107	2863	2244	637	485	267	617	
4	4/11/2005	NOSHER A	HATTIAN DUPATTA	MUZAFF ARABAD	MUZAFF ARABAD	AJK	1222	575	647	165	58	5	165	974
5	10/10/2005	MERA MINDGRAN	HEER KOTLI	MUZAFF ARABAD	MUZAFF ARABAD	AJK	1117	572	545	140	10	3	127	595
6	11/11/2005	BOR MUZAFF AR SHAH	SERI DARRA	MUZAFF ARABAD	MUZAFF ARABAD	AJK	249	132	117	41	13	36	41	167
7	11/11/2005	KOLI	MACHHA YARA	MUZAFF ARABAD	MUZAFF ARABAD	AJK	609	305	304	104	0	0	98	428
8	11/11/2005	TIMBEE (TIMBI)	JHAND GRAN	MUZAFF ARABAD	MUZAFF ARABAD	AJK	2103	1090	1013	300	45	2	213	1619

6. Save it as “datafromrisepak” and exit the Microsoft Excel programme.
7. In the ILWIS3 you can now import Tehsil-wise “datatfromrisepak”.

- Use the following options to import the Tehsil level attribute table containing village level data in ILWIS3.
- In ILWIS3 use pull down option >File > Import> Table> select file datafromrisepak> next> select the table to be imported (worksheet)> do not select key column> select as it is in SQL query> specify output name and convert to ILWIS format.

8. Following are the Tehsil data you import in ILWIS:

- **ABBASPUR**
- **ABBOTTABAD**
- **ALLAI**
- **ALPURI**
- **ATHMUQAM**
- **BAGH**
- **BALA KOT**
- **BARNALA**
- **BATAGRAM**
- **BHIMBER**
- **BISHAM**
- **CHAKISAR**
- **DASSU SUB-DIVISION**
- **DHIR KOT**
- **DUDYAL**
- **F.R KALA DHAKA**
- **FATEHPUR THAKIALA**
- **HAJIRA**
- **HATTIAN**
- **HAVELI**
- **KOTLI**
- **MANSEHRA**
- **MARTOONG**

- **MIRPUR**
- **MUZAFFARABAD**
- **OGHI**
- **PALAS SUB-DIVISION**
- **PALLANDARI**
- **PATTAN SUB-DIVISION**
- **PURAN**
- **RAWALAKOT**
- **SAMAHNI**
- **SEHNSA**

F	Variable Name	Variable Definition
1	Activity Date	Activity Date
2	Village/Town Name	Name of the Village/Town from the 1998 Population Census of Pakistan/AJK
3	Patwar Circle	Name of Patwar Circle in 1998 from the 1998 Population Census of Pakistan/AJK
4	Tehsil	Name of Tehsil in 1998 from the 1998 Population Census of Pakistan/AJK
5	District	Name of District in 1998 from the 1998 Population Census of Pakistan/AJK
6	Province	Name of Province in 1998 from the 1998 Population Census of Pakistan/AJK
7	Total Population	Total Population in 1998 in Village/Town from the 1998 Population Census of Pakistan/AJK
8	Male Population	Male Population in 1998 in Village/Town from the 1998 Population Census of Pakistan/AJK
9	Female Population	Female Population in 1998 in Village/Town from the 1998 Population Census of Pakistan/AJK
10	Total Housing Structures	Total Housing Structures in 1998 in Village/Town from the 1998 Population Census of Pakistan/AJK
11	Number of Pakka Houses in 1998	Number of Pakka (Permanent Construction) Houses in 1998 in Village/Town from the 1998 Population Census of Pakistan/AJK
12	Number Houses with potable water	Number Houses with potable water in 1998 in Village/Town from the 1998 Population Census of Pakistan/AJK
13	Number of Houses with electricity	Number of Houses with electricity in 1998 in Village/Town from the 1998 Population Census of Pakistan/AJK
14	Area (acres)	Area (acres) in 1998 of Village/Town from the 1998 Population Census of Pakistan/AJK. Area not available for Urban regions.
15	Access Indicator	Subjective Indicator of access based on field reports
16	Distance From islamabad (km)	Distance in km from the village to islamabad, coordinate of Islamabad provided by esri cities 2002
17	Distance from District HQ (km)	Distance in km from the village to the district headquarter.
18	Distance from Major Road (km)	Distance in km from the village to nearest major road, roads dataset is a composition of Digital Chart of the Web (DCW) and VMAP-level1
19	Disaster Indicator	Subjective Indicator of physical damage based on Field Report
20	Distance from epicenter (km)	Distance in km from the village to epicenter, coordinate of epicenter provided by USGS:34.493N, 73.629E. Villages with the distance to epicenter 999 are in districts thought not to be affected by the earthquake
21	Houses Unlivable	Categorical Indicator for how many houses are no longer livable - based on Field Reports
22	Status of School	Categorical Indicator for whether school exists and is functional - based on Field Reports
23	Status of Health Facility	Categorical Indicator for whether any Health center, BHU etc. exists and is functional - based on Field Reports
24	People needing medical assistance	Categorical Indicator for how many individuals need medical attention - based on Field Reports
25	Mortality Count	Categorical Indicator for mortality count - based on Field Reports
26	Status of Electricity	Categorical Indicator for whether village/town has electricity - based on Field Reports
27	Drinking Water Available	Categorical Indicator for whether village/town has drinking water available -

		based on Field Reports
28	Food Available	Categorical Indicator for whether village/town has adequate food supplies - based on Field Reports
29	Security Situation	Categorical Indicator for security situation in village/town - based on Field Reports
30	Assistance Indicator	Categorical Indicator for village/town has received any assistance - based on Field Reports
31	NGO/Organization/Individual Name	Name of the organization, NGO or individuals providing assistance in the Village/Town - Based on Field reports
32	Date Relief Distributed	Date organization, NGO or individuals provided assistance in the Village/Town - Based on Field reports
33	Number of Blankets Distributed	Reported number of blankets provided in Village/Town by NGO/Organization/Individuals - Based on Field Reports
34	Number of Tents Distributed	Reported number of tents provided in Village/Town by NGO/Organization/Individuals - Based on Field Reports
35	Volume of Food Distributed	Reported volume/quantity of Food provided in Village/Town by NGO/Organization/Individuals - Based on Field Reports
36	Food Units	Kgs, units, etc
37	Amount of Cash Distributed	Reported amount of cash assistance provided in Village/Town by NGO/Organization/Individuals - Based on Field Reports
38	Volume of Medicine Distributed	Reported volume/quantity of medicine provided in Village/Town by NGO/Organization/Individuals - Based on Field Reports
39	Medicine Units	???
40	Type of Medicine	Reported type of medicine provided in Village/Town by NGO/Organization/Individuals - Based on Field Reports
41	Most Important need	Most important need identified in Village/Town by NGO/Organization/Individuals - Based on Field Reports
42	Distance to nearest Relief Base Camp	Distance of village/town to the nearest Relief Base camp - Based on Field Report
43	Doctors Available	Whether any doctors are available in the village/town - Based on Field reports
44	People evacuated	Categorical Indicator for how many individuals have been evacuated from the village/town - based on Field Reports
45	(Relief Info Consistency) Across Record Consistency	Consistency is High if all information posted regarding the Village/Town is in agreement; Average if there is some discrepancy and Low if there are large discrepancies in Information posted
46	(Relief Info Consistency) Within Record Consistency	Consistency is High if all information posted regarding the Village/Town is in agreement; Average if there is some discrepancy and Low if there are large discrepancies in Information posted
47	Notice Board (remark)	

[Source: website of RISEPAK (Relief Information System for Earthquakes – Pakistan) <http://www.risepak.com/ListVillages.aspx> 15 February 2006]

9. You can join all the table step by step or using script in ILWIS.

☞

- To join table use pull down menu > Operations> Table operations> Glue Tables> select number of input table as 4 > Select tables> give output table name> select vertical option> show.

10. Join up to final table tehsil_data_rice_pak containing all the tehsil tables.
11. Make a copy of table tehsil_data_rice_pak giving new name working_tehsil with break dependency. Now you can work in this table working_tehsil.
12. The table working_tehsil contain following variables in columns F1 to F46.
13. In working_tehsil table convert column F3, F4, F5 and F6 as ID domain and give names risepak_patwarcircle_name, risepak_tehsil_name, risepak_district_name and risepak_province_name.
14. Create table working-tehsil_name with domain risepak_tehsil_name_id.
15. Now your Tehsil attribute database from RISE PAK source is compatible with your spatial database.

Here in this exercise to save time you do not have to carry out exercise as mentioned above, this is for your information/reference only. Throughout this chapter you will work on a dataset dealing with the Tehsils of 8 October 2005 earthquake affected areas of Pakistan (prepared for you based on RISE PAK). The Tehsil boundaries are have been digitized from available map source. We will create tables for each tehsil attribute information.

- ☞

 - Open polygon map working_tehsil_name and click on a few different tehsil to know their contents.
 - Open domain working_tehsil_name and look at the contents.
 - Calculate histogram for polygon map working_tehsil_name, For that click the right mouse on the working_tehsil_name polygon map in the catalog, Select Statistics then Histogram.
 - Calculate histogram dialog box displays Map name working_tehsil_name.
 - Click Show button or press <Enter>.
 - The histogram table will show the NrPol (number of polygon in that ID domain), Perimeter (perimeter of the polygon), Area (area of each polygon in square meters) and the graph in one side.
 - Close the histogram table, domain and the map window.

Each tehsil is coded by a unique ID and information from each individual tehsil can be obtained from the table connected to the map. The polygon map working_tehsil_name_id has domain *ID (Identifier)*. Now you will create the table contains the necessary columns:

- Expand the Create item in the Operation Tree and double-click the New Table item. The Create Table dialog box is opened.
- Create table dialog box display option will display.
- Type in the text box Table name `working_tehsil_name`, For description type Attribute table for tehsils of earthquake affected areas of northern Pakistan.
- Choose domain `ricepak_tehsil_name_id` by using the drop-down list box. We are using this domain because the domains of `working_tehsil_name` polygon map and `tahsil location point map` is same. We can link this table as attribute table for the tehsil maps
- The table `working_tehsil_name` will open with only domain column without any other column.

– Now you have to keep all the attribute data of the Tehsils level only of the earthquake affected area of northern Pakistan (instead of village level) as the project requirements might be of Tehsil level general planning, by creating the columns. Some of the information (e.g. Area, Latitude, Longitude, etc.) you have to bring from other table by using the option Join Column. Some information you use the table calculation, and some information using key-board entry.

You have to develop attribute table with all the necessary topics. The items you will try to include now in tehsil attribute table as follows:

Column Name	Item (Variable Name)	Domain	Variable Definition	F column
	Activity Date			F1
	Village/Town Name		Name of the Village/Town from the 1998 Population Census of Pakistan/AJK	F2
	Patwar Circle		Name of Patwar Circle in 1998 from the 1998 Population Census of Pakistan/AJK	F3
Tehsil	Tehsil	Ricepak_tehsil_name	Name of Tehsil in 1998 from the 1998 Population Census of Pakistan/AJK	F4
Area		value	Area from Histogram table	
? Tehsil_its	Tehsil name as given in ITC data	TEHSIL_NAM	Tehsil name as given in data downloaded from ITC site	
? Area_from_its		value		
	District		Name of District in 1998 from the 1998 Population Census of	F5

			Pakistan/AJK	
	Province		Name of Province in 1998 from the 1998 Population Census of Pakistan/AJK	F6
Tot_population	Total Population		Total Population in 1998 in Village/Town from the 1998 Population Census of Pakistan/AJK	F7
Tot_male_population	Male Population	value	Male Population in 1998 in Village/Town from the 1998 Population Census of Pakistan/AJK	F8
Tot_female_population	Female Population	value	Female Population in 1998 in Village/Town from the 1998 Population Census of Pakistan/AJK	F9
	Total Housing Structures		Total Housing Structures in 1998 in Village/Town from the 1998 Population Census of Pakistan/AJK	F10
Tot_pakkahouse	Number of Pakka Houses in 1998	value	Number of Pakka (Permanent Construction) Houses in 1998 in Village/Town from the 1998 Population Census of Pakistan/AJK	F11
Tot_house_water	Number Houses with potable water	value	Number Houses with potable water in 1998 in Village/Town from the 1998 Population Census of Pakistan/AJK	F12
	Number of Houses with electricity	value	Number of Houses with electricity in 1998 in Village/Town from the 1998 Population Census of Pakistan/AJK	F13
? Area_census	Area (acres)	value	Area (acres) in 1998 of Village/Town from the 1998 Population Census of Pakistan/AJK. Area not available for Urban regions.	F14
	Access Indicator		Subjective Indicator of access based on field reports	F15
Avg_dist_islamabad	Distance From islamabad (km)	value	Distance in km from the village to islamabad, coordinate of Islamabad provided by esri cities 2002	F16
Avg_dist_maj_road	Distance from Major Road (km)	value	Distance in km from the village to nearest major road, roads dataset is a composition of Digital Chart of the Web (DCW) and VMAP-level1	F18
disasterindicator	Disaster Indicator	class	Subjective Indicator of physical damage based on Field Report	F19
Avg_dist_epicenter	Distance from epicenter (km)	value	Distance in km from the village to epicenter, coordinate of epicenter provided by USGS:34.493N, 73.629E. Villages with the distance to epicenter 999 are in districts thought not to be affected by the earthquake	F20
	Houses Unlivable		Categorical Indicator for how many houses are no longer	F21

			livable - based on Field Reports	
	Status of School		Categorical Indicator for whether school exists and is functional - based on Field Reports	F22
	Status of Health Facility		Categorical Indicator for whether any Health center, BHU etc. exists and is functional - based on Field Reports	F23
Injury_medi-assit	People needing medical assistance	class	Categorical Indicator for how many individuals need medical attention - based on Field Reports	F24
Death_count	Mortality Count	class	Categorical Indicator for mortality count - based on Field Reports	F25
	Status of Electricity		Categorical Indicator for whether village/town has electricity - based on Field Reports	F26
	Drinking Water Available		Categorical Indicator for whether village/town has drinking water available - based on Field Reports	F27
	Food Available		Categorical Indicator for whether village/town has adequate food supplies - based on Field Reports	F28
securitysituation	Security Situation	class	Categorical Indicator for security situation in village/town - based on Field Reports	F29
	Assistance Indicator		Categorical Indicator for village/town has received any assistance - based on Field Reports	F30
	NGO/Organization/Individual Name		Name of the organization, NGO or individuals providing assistance in the Village/Town - Based on Field reports	F31
	Date Relief Distributed		Date organization, NGO or individuals provided assistance in the Village/Town - Based on Field reports	F32
	Number of Blankets Distributed		Reported number of blankets provided in Village/Town by NGO/Organization/Individuals - Based on Field Reports	F33
	Number of Tents Distributed		Reported number of tents provided in Village/Town by NGO/Organization/Individuals - Based on Field Reports	F34
	Volume of Food Distributed		Reported volume/quantity of Food provided in Village/Town by NGO/Organization/Individuals - Based on Field Reports	F35
	Food units		Kgs, units, etc.	F36
	Amount of Cash Distributed		Reported amount of cash assistance provided in Village/Town by NGO/Organization/Individuals	F37

			- Based on Field Reports	
	Volume of Medicine Distributed		Reported volume/quantity of medicine provided in Village/Town by NGO/Organization/Individuals - Based on Field Reports	F38
	Medicine units		????	F39
	Type of Medicine		Reported type of medicine provided in Village/Town by NGO/Organization/Individuals - Based on Field Reports	F40
mostimportantneed	Most Important need	class	Most important need identified in Village/Town by NGO/Organization/Individuals - Based on Field Reports	F41
	Distance to nearest Relief Base Camp		Distance of village/town to the nearest Relief Base camp - Based on Field Report	F42
	Doctors Available		Whether any doctors are available in the village/town - Based on Field reports	F43
	People evacuated		Categorical Indicator for how many individuals have been evacuated from the village/town - based on Field Reports	F44
	(Relief Info Consistency) Across Record Consistency		Consistency is High if all information posted regarding the Village/Town is in agreement; Average if there is some discrepancy and Low if there are large discrepancies in Information posted	F45
	(Relief Info Consistency) Within Record Consistency		Consistency is High if all information posted regarding the Village/Town is in agreement; Average if there is some discrepancy and Low if there are large discrepancies in Information posted	F46
	Notice Board (remark)		Remarks	F47

First of all you have to bring the area in square meter of each tehsil polygons and the geographic coordinate of tehsils points. For this you have to use the table joining options.

2.2 Table joining

You can store results of an aggregation in another table, which has the same domain as the group by column used in the aggregation, in order to reduce the data redundancy. Now you will see how you can use columns from one table in another table. The process of linking tables is called *joining*. To join tables you need two tables, one that receives data and one that provides data. Furthermore, you need a *common domain* in the two tables. When you want to combine information from two tables you may have one of the following four situations:

- **1. The domain of the current table is the same as the domain of the other table from which you want to join a column** (see Figure 3.2).

You can directly obtain data from the second table: the link between the two tables is through the common domain of both tables.

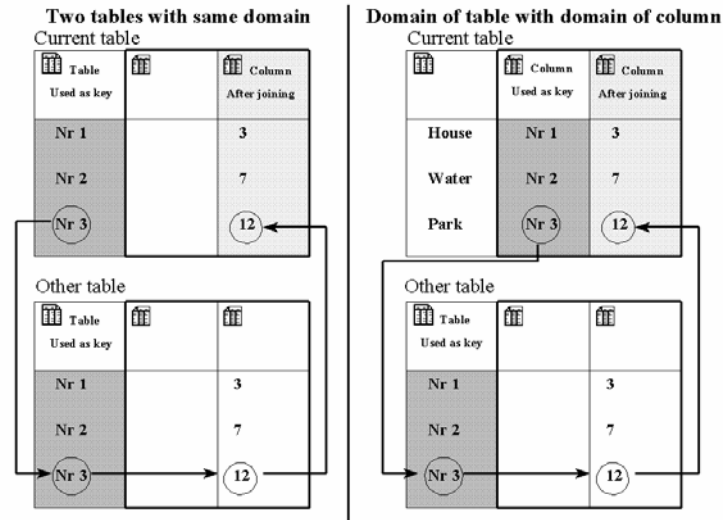


Figure 3.2: Two different ways of table joining. Dark shaded columns are used as key. The arrows indicate how the link is made. Fig 9-a (left): The domain of the current table is the same as the domain of the other table from which you want to join a column. Fig 9-b (right): The domain of a column in the current table is the same as the domain of the other table from which you want to join a column

- **2. The domain of a column in the current table is the same as the domain of the other table from which you want to join a column.** Specify a *key column*, in the current table; you can then directly obtain data from the second table. The link between the two tables is through the selected key column in the current table and the domain of the second table.
- **3. The domain of the current table is the same as the domain of a key-column in the other table from which you want to join a column.** In this case you have two possibilities:

The key column in the other table contains unique class names, ID's or values. In that case you only have to specify the *key column* in that table (Figure 3.3-A)

The key column in the other table does not contain unique class names, ID's or values. In that case direct joining is not possible, since there may be more than one possibility to join. To solve that, you need to *aggregate* the values via the *key column* (Figure 3.3-B)

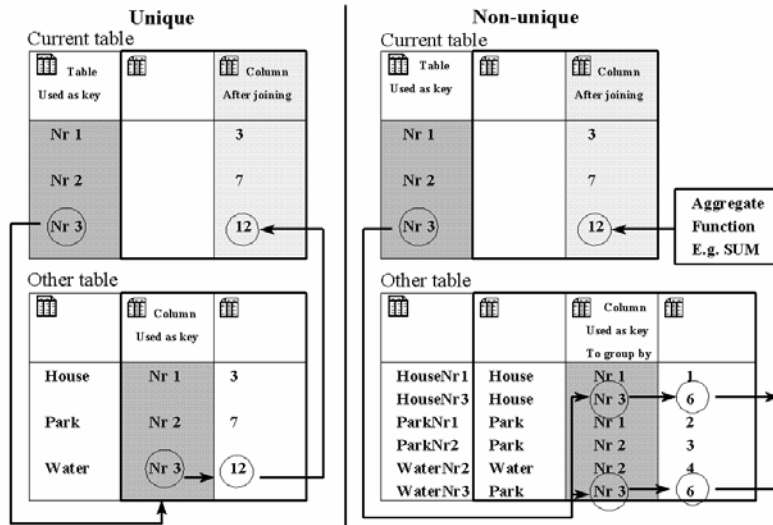


Figure 3.3: The domain of the current table is the same as the domain of a column in the other table from which you want to join a column. Dark shaded columns are used as key. The arrows indicate how the link is made. Fig 3.3-a (left): The column in the other table contains unique class names, ID's or values. Fig 3.3-b (right): The column in the other table does not contain unique class names, ID's or values

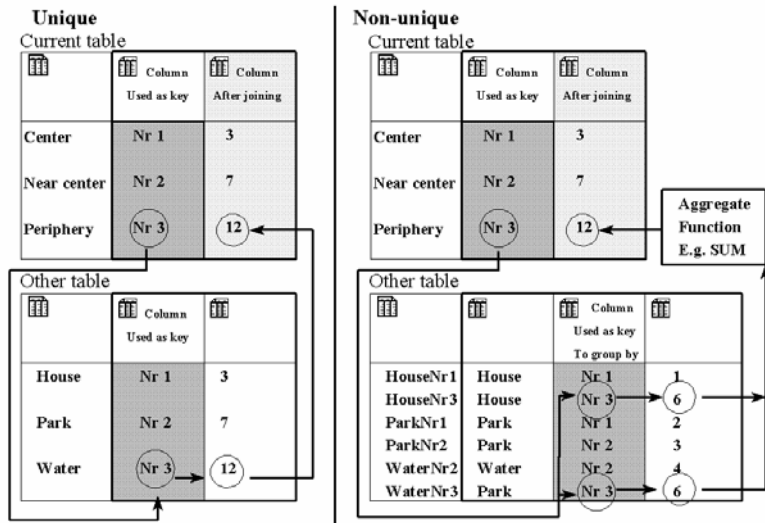


Figure 3.4: The domain of a column in the current table is the same as a domain of a key column in the other table from which you want to obtain data. Dark shaded columns are used as key. The arrows indicate how the link is made. Fig 3.4-a (left): The key column in the other table contains unique class names, ID's or values. Fig 3.4-b (right): The key column in the other table does not contain unique class names, ID's or values


- **4. The domain of a column in the current table is the same as a domain of a key column in the other table from which you want to obtain data.**
In this case we also have the same two possibilities as in the third situation shown above (see Figure 3.4).

Joining via table domains




Polygon histogram

First you will join two tables which both have the same domain (figure 3.3-a). The first table is the attribute table `working_tehsil_name`, with which you have been working. To this current table, a column from *Polygon Histogram* `working_tehsil_name` will be joined. This polygon histogram contains statistical information on each tehsil: the number of polygons, the perimeter and the area.

- 
- Open the polygon histogram `working_tehsil_name` and evaluate its contents.
 - Open the File menu and select the Properties command to find out the domain of the table. Click Cancel.

The polygon histogram `working_tehsil_name` has the same domain as the attribute table `working_tehsil_name`. Note that the column `NrPol` (number of polygons) only contains the value 1. This is logical, since each tehsil has a unique identifier (due to this unique name you make the domain of the `working_tehsil_name` as ID domain “`working_tehsil_name_id`”), and therefore occurs only once. The column `Perimeter` contains the length of the border of each tehsils (in meters). The column `Area` contains the area of each tehsil (in square meters). We want to read this column into the attribute table `tehsil`.

- 
- Close the polygon histogram `working_tehsil_name` and open the attribute table `working_tehsil_name`.
 - From the Columns menu, select the Join command. The Join Wizard will display.
 - Select for the Table: the polygon histogram `working_tehsil_name` and for the Column: `Area`.

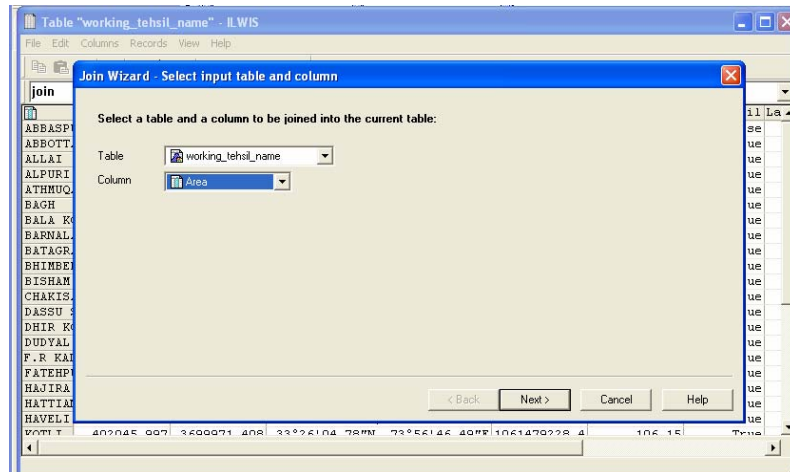


Figure 3.5: The join wizard used for joining a column from two tables with the same domain

Note that the output column name also changes to Area.

- Click Next in the Join Wizard. Accept Area for output column and click Finish.
- Click OK in the Column properties dialog box.
- Check the result and close the table working_tehsil_name.

Column Area is now added to the attribute table working_tehsil_name you have made as per the polygon spatial database you have. We could use this column, together with the column Perimeter (that was in fact also obtained from the polygon histogram through the same procedure) to evaluate the shape of the tehsil. A shape factor can be obtained by the ratio of the perimeter and the area. hsa is the polygon histogram. Shape factor is useful in some application which is not of use here tehsil database.



Point map Open as table

Now again, you will join two tables which both have the same domain (Figure 3.2-a). The first table is the attribute table working_tehsil_name, with which you have been working. To this current table, a column from *Point Map* working_tehsil_name will be joined. This point map as table contains coordinates X and Y information Name on each tehsil as location. When you study the point map working_tehsil_name open as table, the domain of the table of the point map working_tehsil_name_id and the column Name in that table is same as the domain of the table you created working_tehsil_name_id.

- Right click on the point map working_tehsil_name and select Open as Table and evaluate its contents.
- Open the File menu and select the Properties command to find out the domain of the table. Click Cancel.
- Open the View pull down menu and select the Command Line in the table of the point map.
- In the command line type $x=crdx(\text{coordinate})$ and enter and column properties window will appear, accept it by selecting ok button .
- Similarly for y value type in the command line $y=crdy(\text{coordinate})$ and enter and column properties window will appear, accept it by selecting ok button .
- Close the point map working_tehsil_name which is open as table.

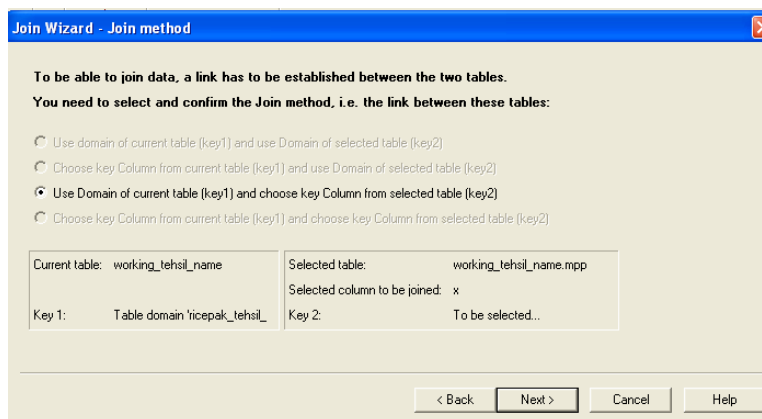


Figure 3.6: The join wizard – Join method page, using a key column in the current table to join a column from another table. The other table has the same domain as the key column in the current table.

We want to read the columns X and Y (coordinate) into the attribute table working_tehsil_name.

- Again right click on the point map working_tehsil_name in the Main Window and select opened as table.
- Open the attribute table working_tehsil_name.
- Adjust both tables side by side.

- From the Columns menu in the attribute table working_tehsil_name select the Join command. The Join Wizard dialog box is opened.
- Select the working_tehsil_name point map using the drag-down list box, and for the Column: Select x
- Click Next in the Join Wizard. Accept the Join method, Link between two tables and select output column by clicking Next and then Finish.

The output column name in the attribute table working_tehsil_name will be x. You can give new column name also.



- Check the result and repeat the same procedure for Y.

Column X and Y are now added to the attribute table working_tehsil_name you have made. We could use this column, together with the coordinate system (you have used to digitized the map to create the working_tehsil_name) to find out the longitude and latitude of the tehsils as the general location of the tehsil in the degree-decimal latitude-longitude coordinate. To find out the latitude and longitude values you can use the following syntax in the command line of your attribute table working_tehsil_name

Latlong=transform(coord(x,y,utm43wgs84,latlon)

Where Latlong is the output column with latitude and longitude values of latlon coordinate system, x and y are the values in column x and y related to the coordinate system utm43wgs84.

Create the other necessary columns with appropriate domain, data range, precision, etc. which you have to enter the data in the field of the table by keyboard entries.

You can use different options for *table calculations*. These calculations will be done with *Table Calculation formulas*.

The formulas can be typed on the command line of a table window. A table calculation formula consists of an output column that will contain the result of the calculation, the definition symbol, or the assignment symbol, and an expression:

Output column=Expression

or

Output column:=Expression

When the Output column does not exist, a new column is created in the table. When the Output column exists in the table, the content of the column is replaced by the result of the formula. Column names should start with a non-numeric character and there is no limit to its length.

- When the assignment symbol ($:=$) is used, a column is created, that is not depending on other columns (a source data object). You can edit the data stored in such a column directly, e.g., by double-clicking on the field to be edited.
- When the definition symbol ($=$) is used and the output column does not exist yet, a *dependent column* is created. The data in a dependent column depend on data from other columns via the formula, which was used. When data in one of the source columns is changed, the result also changes, when you make the column up to date. Fields of a dependent column cannot be edited directly to protect the result of a calculation.

The expression usually contains operators and/or functions to specify the calculation to be performed. The expression calculates results for all records of the output column. The available TabCalc operators and functions are listed in the On-line Help and in the Reference Guide of ILWIS manual. You can press F1 at any time in the table window to access detailed information about them.

A formula can be very long. You can move within the formula using the left and right arrow keys, in combination with the CTRL key, or by using the mouse pointer. To retrieve a previous formula, use the arrow up key on the keyboard: this is called the *history*.

-
- ! You can use the following short-cut keys:
- Ctrl+V (to paste information from the Clipboard to the Command line),
 - Ctrl+C (to copy from the Command line to the Clipboard).
- This is extremely useful when you have to edit complicated and long formulas. For an overview of the keyboard shortcuts see the Appendix I of the Reference Guide of ILWIS Manual.
-

2.3 Calculations with value columns

You will now practice with calculations using columns. There are quite a lot of differences in the way you calculate with value columns on the one hand, and class and ID columns on the other. Let's start by using some operators on columns with a value domain. We use the table `working_tehsil_name` to calculate the area of each tehsil in square kilometers, by dividing the column `Area` with 1000,000.



- Open the table `working_tehsil_name` if it is closed.
- In the table window, position the mouse pointer on the command line and type the following formula:
`Area_km=Area/1000000`

- Press Enter.
- The formula appears in the Description and if you need you can type some text in the Description text box, e.g.:
Area of the Tehsils in Square Kilometers.
- The rest of the properties we will leave as they are; accept the default values suggested by ILWIS by clicking OK.

The dialog box is closed, and the program executes the formula. The results are displayed in the column Area_km.

Now the Column properties dialog box is opened. Since the column Area_km did not exist in the table, it now has to be defined. This dialog box provides possibilities to define the new column: domain type, range, precision, position of the column, width of the column, number of decimals and description. The value range defines the range of possible output values in the output column. Calculated values outside this range are assigned the Undefined value (indicated by a question mark: ?). It is thus important to select a range, which encompasses all the possible output values. In the Column properties dialog box, the value range is defined in two text boxes: the minimum value in the first text box, and the maximum value in the other one. The precision text box is used to define the resolution of output values in the output column. A precision of 1 means that output values will be rounded to whole numbers. A precision of 0.1 means that the output values will have 1 decimal. The description text box can be used to enter a text explaining what the column represents. The use of description text boxes is optional, but it is highly recommended, as it helps to remind you what the data means.

Operators for value columns

There are several types of operators and functions that can be used in expressions with value columns.

! You can get an overview of all operators and functions that can be used for value columns in the On-Line Help. Open the Help menu, and select Help on this Window. The Help is opened on the Table window page. Select the hypertext link Command line. Select the hypertext link Table Calculation. Select the hypertext link Operators and functions on value columns. Now you will see an overview of the operators and functions on value columns. Click the hypertext links to get more information and examples of the various operators.

Arithmetic operators

These are the most simple operators, which are used for the multiplication, division, subtraction or addition of columns. Another example of these arithmetic operators will be used to calculate the percent of the mortality in each tehsil in relation to total population according to 1998 Census. Find out the total area of the 33 tehshils under study of earthquake affected areas of northern Pakistan using the table window the

columns option in the Menu-bar, statistics, sum function of column area_km. Or you can use the command line in the table window by typing ?sum(area_km) ↵, you will get the answer something like sum(area_km)=..... That is the total area of the the 33 tehshils under study of earthquake affected areas of northern Pakistan (.....km²). Or you can also find percentage of each tehsil out of 33 tehshils using the syntax; Pc_tehsil=100*(Area_km)/(sum(Area_km)).

- ☞
- In the table window, position the mouse pointer on the command line and type the following formula:
Pc_tehsil=100*(Area_km)/(sum(Area_km)) ↵
From now on the symbol ↵ will be used to indicate that you have to press Enter.
The Column properties dialog box is opened.
 - Type the following description in the Column properties dialog box: percentage of each tehsil out of 33 tehshils and Click OK.

The table window now also contains the column Pc_tehsil with the percentage of the each tehsil out of 33 tehshils of earthquake affected area of northern Pakistan.

Relational operators

Relational operators (=, <, <=, >, >=, <>) test whether one expression is equal, smaller, smaller or equal, larger, larger or equal, or different than another expression. Let us find out, for example, which tehshils larger than 10 square kilometer in area.


- ☞ Let us assume that the tehshils having surface area greater than ten square kilometer are big tehshils.
- Type the following formula on the command line:
Big_tehsil=Area_km>10 ↵
 - The Column properties dialog box is opened.
 - Type the following description: tehsil of larger than 10 square kilometer in area.

Note that the suggested domain in the Column Properties dialog box is Bool. A Bool domain (abbreviation of *Boolean*) has only three possible values: True, False and undefined.

- ☞
- Click OK.


Logical operators

Logical operators such as AND, OR, XOR and NOT, compare two expressions and check if both are true (AND), at least one is true (OR), or only one is true (XOR). The NOT operation checks if an expression is true or false. If the expression is true, the NOT operation will result in false and vice versa. As an example, the AND operator is used to find the tehsils that have between 10 and 50 square kilometer in surface area.



- Type the following formula on the command line:
Large_tehsil=(area_km>10)and(area_km<50) ↵
The Column properties dialog box is opened.
- Type the following description: Larger tehsil having surface area between 10 and 50 square kilometer surface area.
- Click OK.

The same result could be obtained with the INRANGE function.




- Type the following formula on the command line:
Larger_tehsil2=INRANGE(Area_km,10,50)↵
- Click OK in the Column properties dialog box.

Conditional function

The examples that we have used for the relational and logical operators all give output values, which are either true or false or undefined. In practice we use these operators mostly in combination with the *conditional IFF function*, which has the form of an IF...THEN..ELSE statement. The format for a conditional statement in ILWIS is: IFF(*a,b,c*). If condition *a* is true, then return expression *b*, otherwise return expression *c*.

! In the previous DOS versions of ILWIS this function was written as IF(*a,b,c*), with only one F. Now it is defined as a conditional IF Function (IFF).




- Type the following formula on the command line:
Larger_tehsil3=IFF((Area_km>10)and(Area_km<50),
Area_km,?)↵
- Click OK in the Column properties dialog box.

The resulting column (Larger_gr3) contains values for those records where the condition (Area_km>10)and(Area_km<50)was true. The other fields contain a question mark (?). These are so- called *undefined values*.


Conditional functions are used very extensively in ILWIS. You will find that 3 out of 4 formulas you create will contain an IFF statement.

Conditional functions can also be part of another conditional function. In that case we call them nested IFF functions. The calculation performed above can also be written as such a nested IFF function.


- 
- Type the following formula on the command line:
Larger_tehsil4=IFF(Area_km>10,IFF(Area_km<50,
Area_km,?),?)↵
 - Click OK in the Column properties dialog box.

When you use nested IFF functions, you should be careful in determining how many closing brackets you must use (always as many closing brackets as there are IFF's in the formula), and where you put them.

Before continuing it is better to delete the columns that were made in this exercise using the table calculations, except for the columns you have made using join column and column Area_km.

- 
- Select column Pc_tehsil by clicking the name of the column.
 - From the Edit menu choose Delete. Confirm the deletion.
 - Select column Big_tehsil and press the Delete key on the keyboard. Confirm the deletion.
 - Delete the columns, Larger_tehsil1, Larger_tehsil2, Larger_tehsil3, Larger_tehsil4.
 - Close table working_tehsil_name, if you are going to continue the exercise, do not close the table.

Now you create the other required columns in the table working_tehsil_name as per your project requirements. And enter the data in the fields obtained from your analysis using topographic maps, aerial photographs, satellite images, other information sources, etc.

- 
- Open table working_tehsil_name. If it is closed.
 - Select Column and then Add Column.
 - Or you can direct click on the gray part of the table head.
 - Give the column name, domain (if value domain, give the

value range and precision), and description for each of the columns you have created required as per the required table provided to you.

- Columns you have to create are; for example; *disaterindicator* column with class domain (high (h), medium (m), low (l)), *securitysituation* column with class domain (poor (p), normal (n), notknown (nk) , and so on
- If you want to make certain column as class domain before entering the data, then you create domain with domain name of that column name from the File option of ILWIS window, select only as simple class and insert the name and code in the domain editor.
- Create the column you want to make and Enter the fields.
- To calculate the longitude and latitude from the column X and Y based on your coordinate system (utm43wgs84) you have used to prepare the point map you can type in the command line
- long =
CRDX(TRANSFORM(COORD(x,y,utm43wgs84),Latlon))
- lat =
CRDY(TRANSFORM(COORD(x,y,utm43wgs84),Latlon))
- You complete the attribute table with all necessary entry of data and necessary table calculations. This attribute table for the tehsil can later be linked as attribute table for the tehsil polygon map (working_tehsil_name).
- Close table working_tehsil_name. If you are going to continue the exercise, do not close the table.

Operators used for class or ID columns

Columns that have a class or ID domain can also be used in formulas. When using class names or IDs within an expression, these class names and IDs should be put between double quotes, e.g. Disaster Indicator.

The number of operators and functions that you can use for class or ID columns is quite limited. Of course, arithmetic operators make no sense (you can hardly divide one word by another). Relational operators are only limited to two: = (check whether two names are equal) and <> (check if they are unequal). Logical operators and conditional functions, however, are used extensively on class or ID columns.

Some examples are shown below. The first example is finding out the tehsil, which have High Disaster Indicator and Poor Security Situation.

- Open table working_tehsil_name. If it is closed.
- Type the following formula on the command line:
`H_D_Indicator_P_SSituation`
`=(disaterindicator="h")and(securitysituation="p")`
- Click OK in the Column properties dialog box.

The result of this formula is a column, using domain Bool. Now let us combine several operators and an IFF function. We will find out the bigger tehsils with surface area larger than 10 square kilometer and having Security Situation either "poor" or "not know".

- Type the following formula on the command line:
`Big_tehsil_and_sspnk=IFF(securitysituation="p")or(securitysituation="kn"))and(area_km>10),%k,"?")`

Note that in the Column properties dialog box the suggested domain of the output column is working_tehsil_name. %K means the record name of the domain working_tehsil_name (which fulfil the condition in the table calculation). This is because the possible results of the formula are either tehsil ID from the domain working_tehsil_name_id, or ? (undefined).

- Click OK in the Column Properties dialog box.

After studying the column H_D_Indicator_P_SSituation and column Big_tehsil_and_sspnk, delete both of these columns.

How to define domains for IFF functions with class or ID

When we work with IFF functions that give class or ID results, we may have four different situations:

- 1. *IFF(expression, domain1, "?")*. The result of the expression fits in an existing class or ID domain (mostly one of the input columns), or is the undefined value. This is illustrated by the example shown above;
- 2. *IFF(expression, domain1, "name")*. The result of the expression fits in a existing domain or is a name which is not yet in the domain. In this case you must add one item Name to the existing domain. A similar situation is *IFF(expression, "name", domain1)*;

- 3. *IFF(expression, domain1, domain2)*. The result of the expression fits either in domain1 or in domain2. In this case you must merge the two domains;
- 4. *IFF(expression, "name", "name")*. The result of the expression is two names, which fit, or may not fit, in an existing domain.

In situations 2, 3, and 4 you cannot simply use an existing domain for the output column. You should either add items to the existing domain, or you should create a new domain, depending on the THEN and ELSE parts of the IFF function. In case 2 you can add an extra name to the existing domain. In case 3 you can merge the two domains, either by adding the items of one domain to the other or by making a new domain which contains the items of both. In case 4 you can make a new domain, which contains both names.

When ILWIS encounters in an IFF function one of the first three possibilities, the program will suggest that you use an existing domain. When you decide to use that existing domain and press OK in the Column properties dialog box, you will get a warning, and you are asked whether you want to add the missing items to that domain. However, it is not advisable to generate large domains with a mixture of information. In many cases it is better to generate a new domain. To do so, you will have to press the Create domain button in the Column properties form.

Note that it is always possible to use the output domain String for all four situations given above. However, it is not advised, as a string column cannot be used in combination with a map. Let us now look at the situations 2, 3, and 4 with some examples.

☞

- Type the following formula on the command line:
`securitysituation_not_poor=IFF(ecuritysituation<>"p", securitysituation,"security situation is poor")`


This statement attempts to replace the class p with security situation is poor. This is an example of situation 2, *IFF(expression, domain1, "name")*. If the security situation is not poor then take the names from the column securitysituation, otherwise use the word (phrase) "security situation is poor". So in fact we replace the word p (poor) with the word (phrase) "security situation is poor". This is a situation in which you could still use the existing domain securitysituation (belonging to the column securitysituation) and add the item "security situation is poor".

☞

- Click OK in the Column properties dialog box.

The new column securitysituation_not_poor shows the tehsils of the security situation, which are not poor, as in the column securitysituation, and the new name “security situation is poor” for the “p” securitysituation.


Let us now look at situation 3: *IFF(expression, domain1, domain2)*. We will find out the most important need of tehsils which are having the security situation is poor .

- 
- Type the following formula on the command line:
poorsituation=IFF(securitysituation<>“p”,securitysituation,
mostimportantneed)↵


What we have done here, is to look at whether the securitysituation is “p”. If that is not the case, the class names from the securitysituation column are used, otherwise the class names from the column mostimportantneed.

- Click OK in the Column properties dialog box.

Finally, let us give an example of situation 4, *IFF(Expression, "name", "name")*. Let us assume that the tahsils with securitysituation are in p (poor), n (normal) securitysituation class are “no need to inquiry” and rest [nk (notknown) and undefined] are “need to inquiry”.

- 
- Type the following formula on the command line:
inquiry=IFF((securitysituation =“p”)or(securitysituation
=“n”)),” no need to inquiry”,” need to inquiry”) ↵

In this case it would make no sense to add the new names no need to inquiry and need to inquiry to an existing domain. Since the ELSE and THEN parts of the IFF function contain names that ILWIS cannot identify as being part of an existing domain, the program will display the String domain as the default domain for the output column. In this case, however, we would like to make a new domain: inquiry

- 
- Click the Create domain button (located next to the arrow of the domain list box). The Create Domain dialog box is opened.
 - Enter the Domain Name: inquiry. Select Class and then Click OK.
 - The Domain Editor is opened. Add the two items: no need to inquiry and need to inquiry with codes nni and ni. Close the

Domain Editor. Now you are back in the Column properties dialog box. The domain inquiry is now selected. Click OK.



- Type the following formula on the command line
`inquiry1=IFF(ISUNDEF(securitysituation,inquiry,securitysituation))`
- The Domain Editor is opened. Add the two items: no need to inquiry and need to inquiry with codes nni and ni. Close the Domain Editor. Now you are back in the Column properties dialog box. The domain inquiry is now selected. Click OK.

What we have done here, is to look at whether fields in the column securitysituation have undefined values, using a special function ISUNDEF. If that is the case, the names from the inquiry column are used, otherwise the names from the column securitysituation. Since the items in the column securitysituation also refer to inquiry.

There are many more operators and functions that can be used for class or ID domains than the ones we have treated in this exercise.

-
- ! You can get an overview of all operators and functions that can be used for class or ID columns in the On-Line Help and the Reference Guide of ILWIS Manual. There is a series of functions that allows you to calculate with coordinates of maps, and a group of functions to calculate with colors.
-

Before you continue with the next steps, it is better to delete most of the unnecessary columns you have created.



- Select column inquiry by clicking the name of the column, and press the Delete key on the keyboard. Confirm the deletion.
- Delete also the other unnecessary columns (inquiry1, poorsituation, securitysituation_not_poor, H_D_Indicator_P_SSituation, Big_tehsil_and_sspnk,) that you have made. Not the initial ones.
- Close the table working_tehsil_name. If you are going to continue the exercise, do not close the table.

Summary: Table Calculation

- Calculations are typed on the command line of the table window with formulas. Each formula contains the name of an output column, a definition or assignment symbol (which determines whether the output column will be dependent or not) and an expression.
- An expression contains operators, functions and operands (names of columns, strings and values).
- There is a different set of operators and functions on value columns and on class, ID, and string columns. You cannot mix values and strings in an output column. The result column should be either value or string.
- On value columns we can use arithmetic, relational, and conditional operators, and a whole set of functions, of which the IFF function is the most important.
- *IFF(expression, then, else)* statements are very common in ILWIS. IFF functions can be combined into nested IFF statements.
- On class, ID or string columns we can use a smaller set of operators and functions (some relational and conditional operators).
- When the result of an IFF statement is a string, you can have 4 possibilities:

IFF(expression, domain1, "?"), *IFF(expression, domain1, "name")*,
IFF(expression, domain1, domain2), and *IFF(expression, "name", "name")*.

2.4 Classifying data in a column

In the previous section you have seen a number of different operators and functions that can be used in calculations with columns. In this section you will learn another useful function: *classification of values in a column*.

The table `working_tehsil_name`, containing information on the tehsils of earthquake affected areas of northern Pakistan, will be used again as an example.



Domain Group

The column `Area_km` shows different values for each tehsil. You might want to have the group that is having the certain ranges of area of the tehsil. To simplify this information it would be handy if we could classify it into a number of classes. This is done with the *CLFY function*. This function classifies values into a number of classes. The classes should be defined beforehand, and are stored in a so-called *domain Group*. A Group domain is a special type of class domain, in which for each class a boundary value is given.

Table 2: Boundary values for classifying the area of tehsils

Upper Bound	Name
-------------	------

0	0 square kilometer
10	>0 - 10 square kilometer
20	>10 - 20 square kilometer
50	>20 - 50 square kilometer
1000	> 50 square kilometer

You will first create a group domain, called Area_cl.

- Open table working_tehsil_name, if it is closed and activate the main window of ILWIS, open the File menu and select the Create, and the Create Domain commands. The Create Domain dialog is opened.
- Type for the Domain Name: Area_cl. Click the option: Class and the check box Group. Click OK. The Domain Group is opened.
- From the Edit menu, select the Add Item command. The Add Domain Item dialog box is opened.
- Type for the Upper Bound: 0. Name: 0 square Kilometer. Click OK.
- Press the INS key. The Add Domain Item dialog box is opened.
- Type for the Upper Bound: 10. Name: >0 - 10 square Kilometer. Click OK.
- Repeat the procedure for the other classes listed in the table.
- Close the Domain Group.

Now that the group domain is created we can classify the area of the tehsils given in the table.

- Activate the Table window of ILWIS with the table working_tehsil_name, and type the following formula on the command line:
Area_cl=CLFY(Area_km,Area_cl)↵
- Click OK in the Column properties dialog box. Now you will see a new column with the description for each tehsil grouping.

2.5 Aggregate functions

In the previous sections, the use of several operators and functions that can be used in the calculation of columns were presented. All of these operators and functions work on each field individually, without looking at other fields in the same column.

In several cases, you may want to take into account the value for the same unit in the whole column or in sub-sets of records in a column. For example, you may want to calculate the Area_km, not for each tehsil individually, but for an entire 33 tehsils of earthquake affected areas in northern Pakistan. The functions that allow you to do such kind of operations are called *aggregate functions* (see Figure 3.7). Aggregations are performed on:

- all records of a column, or
- all records that belong to the same group of records as determined by a key or grouping column,

Aggregate functions can be performed on value columns as well as on class and ID columns.

- In case of value columns, all available statistical functions might be meaningful. In short, the following aggregations can be performed: *average, weighted average, count, minimum, median, weighted median, maximum, predominant, weighted predominant, standard deviation, weighted standard deviation, and sum.*

In case of class or ID columns, only the *median*, the *count* and the *predominant* aggregate functions are meaningful.

Aggregation commands can be selected from the Columns menu in a table window, or by typing a statement on the command line of a table window. The first method is shown here. The result of an aggregation can be written into a new column of the same table, in a new column of a new table, or in a new column of another existing table.

Some examples will be shown, again with the table `working_tehsil_name`. The first example is the calculation of the total area of all 33 tehsils of earthquake affected areas in northern Pakistan.

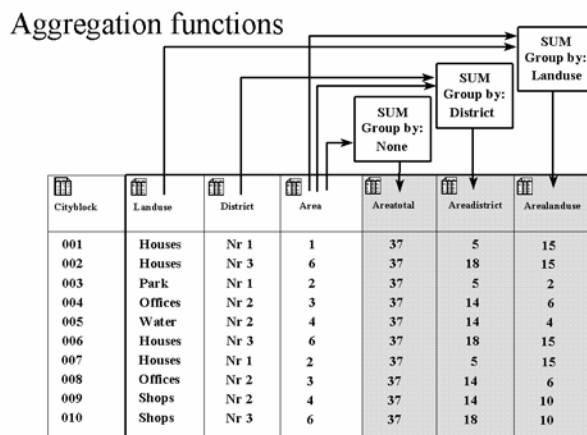


Figure 3.7 : An example of the use of the aggregation function SUM for three situations. Group by: none: all values are summed up. Group by the column District: all values are summed up for each District. Group by: column Landuse

- Open the Columns menu, and select the Aggregation command. The Aggregate Column dialog box is opened.
- Select for the Column: Area_km, the function: Sum, and type as Output Column: Totalarea33. Click OK.
- Type the description: Total area of all 33 tehsils of earthquake affected areas in northern Pakistan in the Column properties dialog box. Click OK.

You will see that the column Totalarea33 contains one value for all records. That is because you have calculated the total area summing up all records. It is evident that the column Totalarea33 is not useful in combination with a map. Since the column only contains 1 value, it says nothing about the different units in the map.

Instead, you can also calculate the total area per securitysituation. Then areas are summed up only for those records that have the same name in the column securitysituation. The column securitysituation is used to group the records. This way we can calculate the total area of each tehsil classified as per your requirement, or any other group column.

- Open the Columns menu and select the Aggregation command. The Aggregate Column dialog box is opened.
- Select for the Column: Area_km, the function: Sum.
- Select the check box Group by, and select the column

securitysituation.

- Type as Output Column: Area_ securitysituation. Click OK.
- Type the description: Total area of each security situation in the Column properties dialog box. Click OK.

The values in the column Area_ securitysituation now show different areas for different securitysituation.



- Check this by opening the Columns menu, and choosing the Sort command. The Sort dialog box is opened. Click Column, and select the column securitysituation. Click OK. The table is sorted according to the column securitysituation.
- In the table window, use the scroll button on the right-hand side and check whether the values in column Area_ securitysituation change for each securitysituation.
- Sort the table again on the domain.

It is rather uneconomical to store the data from the column Area_ securitysituation within the table working_tehsil_name. Out of ... different records, there are only ... (very few) different values in this column. These values do not relate to the tehsils, but to the securitysituation. To reduce the *data redundancy* it would be better to store the results in the securitysituation table.



- Perform the same aggregation; use the same name for the output column, but choose to write the output in the table: securitysituation.
- Open the table securitysituation and look at the result.

Now the data is stored in a more useful way. The table has ... records (...security situations), and each security situation has a value for Area_ securitysituation.

Delete the unnecessary columns you have created before you close the table. You can link the table as attribute table for the tehsil polygon map working_tehsil_name by using Properties option.

Discuss with the resource persons and fellow participants regarding the criteria to find out the most crucial Tehsils for relief activity, reconstruction and rehabilitation.

Develop the attribute columns to determine the most crucial Tehsils for relief activity, reconstruction and rehabilitation and prepare the map.

List down some generalized steps, calculations and syntaxes with reference to your system available for the study of earthquake affected Tehsils of northern Pakistan and to determine the most crucial tehils for relief activity, reconstruction and rehabilitation.

Inventory of Tehsils

1. Import of Tehsil database from existing information/database.
2. Manual enter of the database (.....)
3. Capture the data by joining with the histograms of point map and polygon map (Area, Coordinates X, and Y)
4.
5.
6.
7.

Calculations and syntaxes to create useful/meaningful criteria (column)

8. Latitude_Longitude = transform(coord(x,y,utm43wgs84),latlon)
9.
10.
11.
12.
13.
14.
15.

Write down with flow chart(s) with functions, syntaxes for your project on decision support system for relief activity, reconstruction and rehabilitation.

.....
.....
.....