




# ZIM 9.10

## **Entity-Relationship Model (E-R Model)**



# What is Zim?

## Zim is



**a complete framework to develop and run professional and mission critical applications by tightly integrating a lean relational database, a powerful Fourth Generation Language, an integrated development tool, the integration with outside world and client user interfaces.**



# The Relational Model Basics

The Relational Model was presented by Codd and Date and is based on the customer point of view:

“How the user sees the data”.

*- Tables or Entity Sets -*

## Customers Table


Code	Name	Address	Company	City	Credit
0101	John Voight	1280 Riverside Dr	ACME Inc	New York	1000
0102	Mark Stuart	320 Colonnade Av	B & D Ltd	Boston	1200
0110	Phyllis Morris	25 Fifth Avenue	Mackormik	Ottawa	1000
0105	Mark Knut	111 Main Street	CocaCo Inc	Chicago	850





# Table Characteristics

- Has a name (Customers);
- One or more columns or fields (Code, Name, etc);
- Zero or more rows (Records);
- Primary (Unique) key (Code);
- The position of a row in a table is not relevant.



Code	Name	Address	Company	City	Credit
0101	John Voight	1280 Riverside Dr	ACME Inc	New York	1000
0102	Mark Stuart	320 Colonnade Av	B & D Ltd	Boston	1200
0110	Phyllis Morris	25 Fifth Avenue	Mackormik	Ottawa	1000
0105	Mark Knut	111 Main Street	CocaCo Inc	Chicago	850

**Customers**



# In the Real World...

**...tables are always associated to each other via some specific relationship condition.**



**Departments are grouped in Divisions**

**Employees work for Departments**

**Departments are managed by Managers**

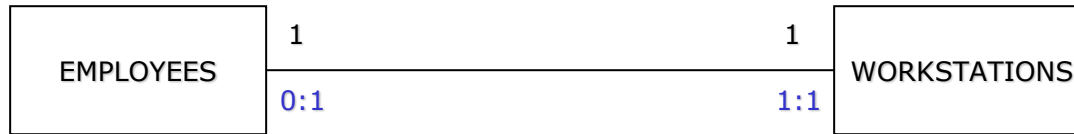
**Employees Live in Cities**

**Departments are located in Buildings**

**Employees Work in Projects**

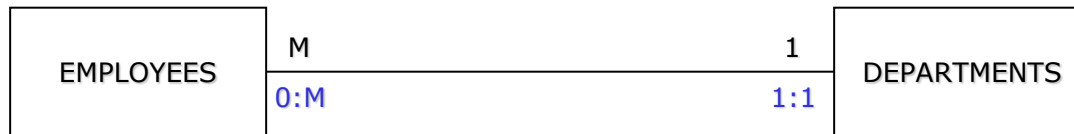
# Relationship Types

One-to-one



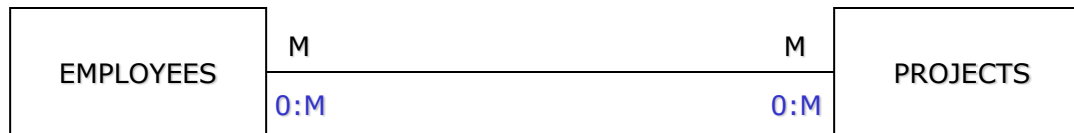
An employee is assigned to one workstation and one workstation may have been assigned to one employee

One-to-many  
or  
Many-to-one



An employee works for one department and in one department may work zero or more employees

Many-to-many



An employee works on one or more projects and one project may have zero or more employees working on it



# Relating Two Tables

## One-to-Many Relationships

**Employees**

ENum	Name	Salary	DeptCode	CityCode	HireDate	Monthly Salary
1	Joseph Travolta	168,000.00	MKT	NYC	1998/10/12	14,000.00
2	Mark Hirudsa	62,001.00	R&D	OTT	1998/12/12	5,166.75
3	Frank Copolla	38,001.00	R&D	OTT	1997/12/08	3,166.75
4	Mary Stuart II	41,501.00	MKT	TOR	1999/02/04	3,458.42
5	Nicholas Nick	26,802.01	R&D	TOR	1997/03/04	2,233.50
10	Carl Santana		R&D	TOR	1998/11/04	
7	Mark Chapman	102,001.00	SAL	OTT	1998/04/21	8,500.08
11	Marshal Winduck	91,802.01	MKT	NWJ	1998/07/17	7,650.17
12	Larry King	53,802.01	R&D	NYC	1998/09/16	4,483.50
13	Alex Sander		R&D	OTT	1997/09/12	
14	Lucy Markham	33,301.00	MKT	OTT	1998/11/09	2,775.08
15	Karin Lalonde	18,602.01	R&D	OTT	1996/12/07	1,550.17
100	Elton Joseph	130,000.00	SAL	LDN	2000/02/24	10,833.33

A row in table **Employees** is related to a row in table **Departments** when the value of **Employees** in **DeptCode** matches one value of the column **DeptCode** in **Departments**.

Foreign Key

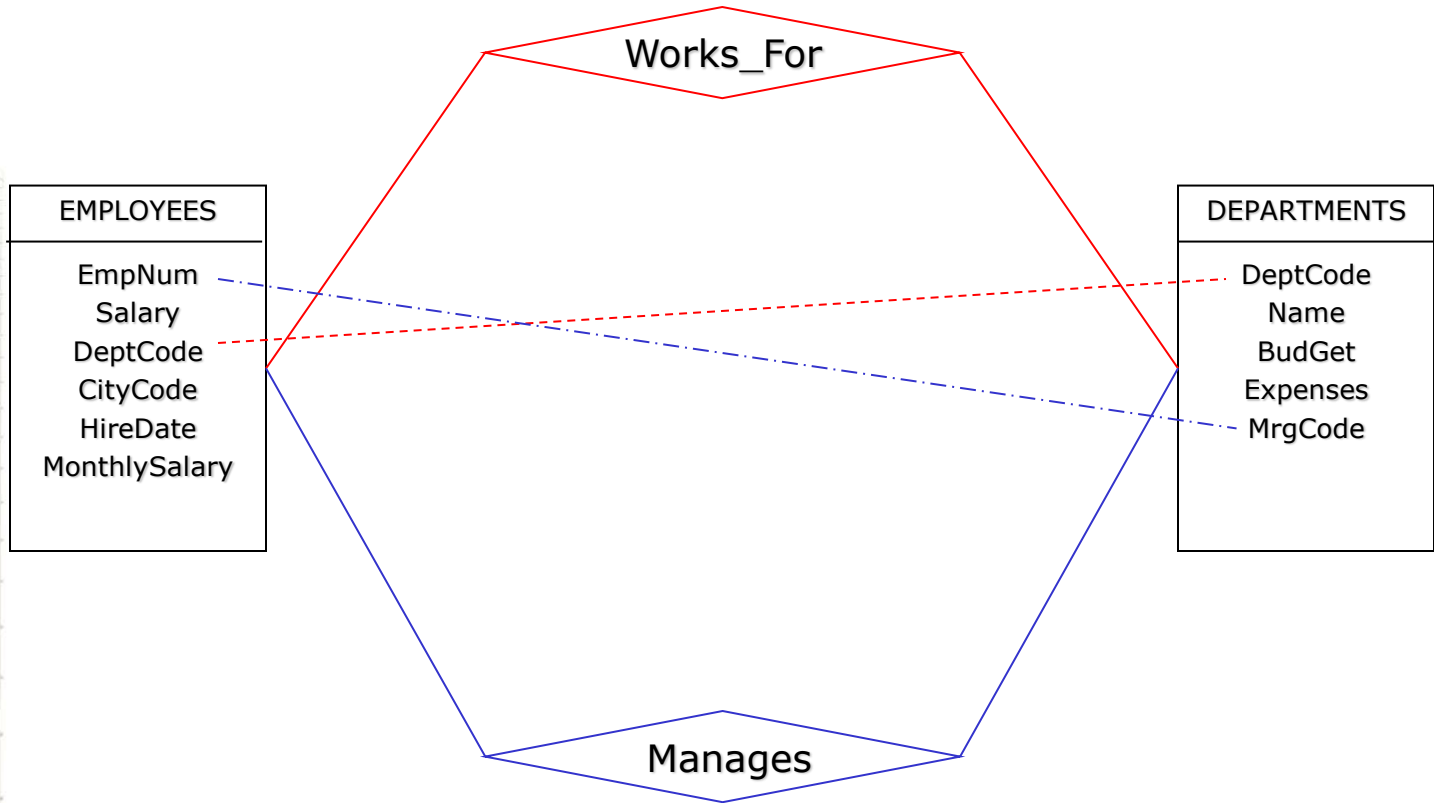
Primary Key

**Departments**

DeptCode	Department Name	MgrCode	Budget
SE	Sales Engineers	2	100000.00
MKT	Marketing	11	180000.00
PRD	Production	0	58000.00
ACC	Accounting	0	80000.00
SAL	Sales	0	120000.00
R&D	Research and Development	10	155000.00

# Relating Two Tables

## One-to-Many Relationships







# Relating Two Tables

## One-to-One Relationships

**This relationship type may occur in Nature but its appearance in a database may be a result of a simplification or a modelling error.**

**For all purposes, One-to-One relationships are treated as One-to-Many.**

**One example is a Country with its capital name.**

# Joining Tables

The relationship `Work_In` is represented by the expression  
`Departments.DeptCode = Employees.DeptCode`



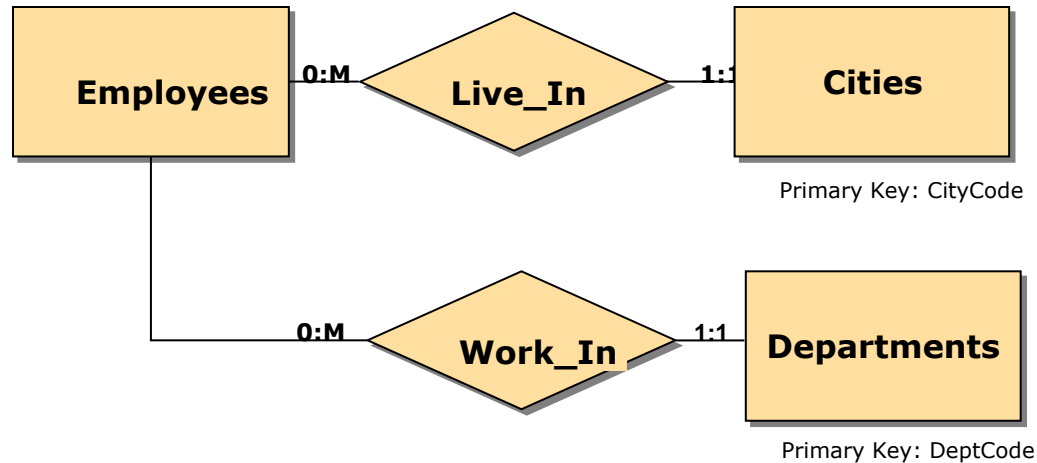
**In Zim, to list all employees working in all departments:**

```
LIST ALL Employees WORK_IN Departments
```

**in SQL:**

```
SELECT * FROM Employees, Departments WHERE \  
Departments.DeptCode = Employees.DeptCode
```

# Using the Relationships



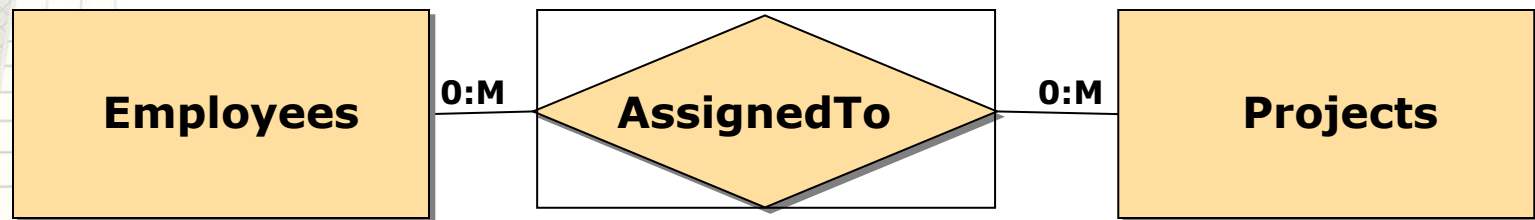
```
List all Employees Live_In Cities \
      Work_In Departments      \
Where Cities.Population > 1500000
```

# Many-to-Many Relationships

These relationships are a combination of tables and a data relationship or a relationship with fields (that is, another table).

The relationship condition is:

```
Employees.EmpNum = AssignedTo.EmpNum and \  
AssignedTo.ProjCode = Projects.ProjCode
```



Primary Key: EmpNum

EmpNum  
ProjCode

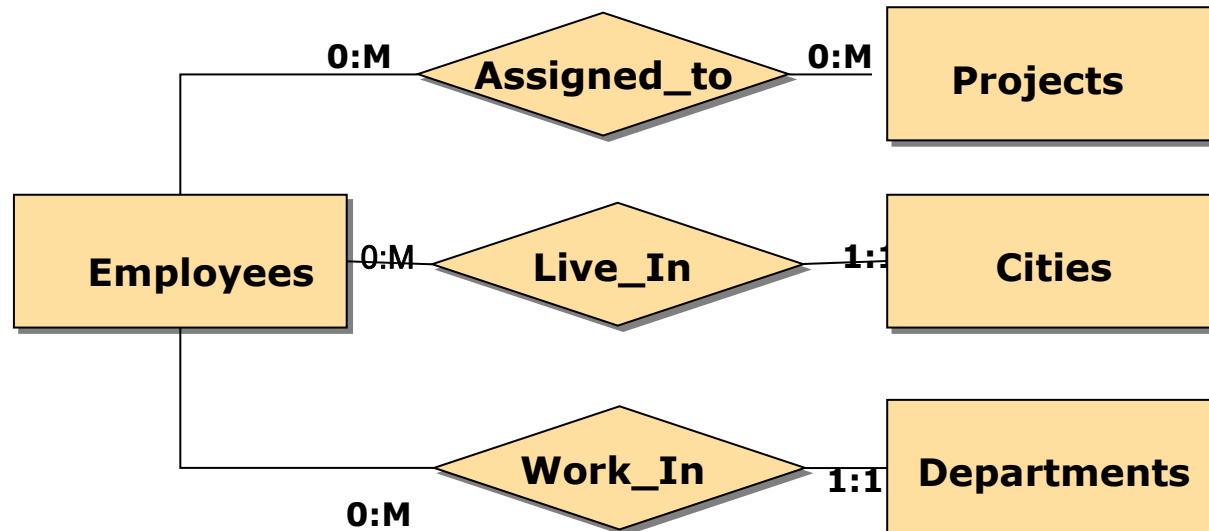
Primary Key: ProjCode

The Primary Key in a Data Relationship is the concatenation of both Foreign Keys (EmpNum and ProjCode) plus some other fields like a date, etc.

# Tests

List the number, name, salaries and project name of all employees that work in one or more projects

List the employee name, city name, project name and department name for all employees working in one or more projects



# Roles

**Roles can be used as synonyms of Entity sets or Relationships**



**List all Employees**

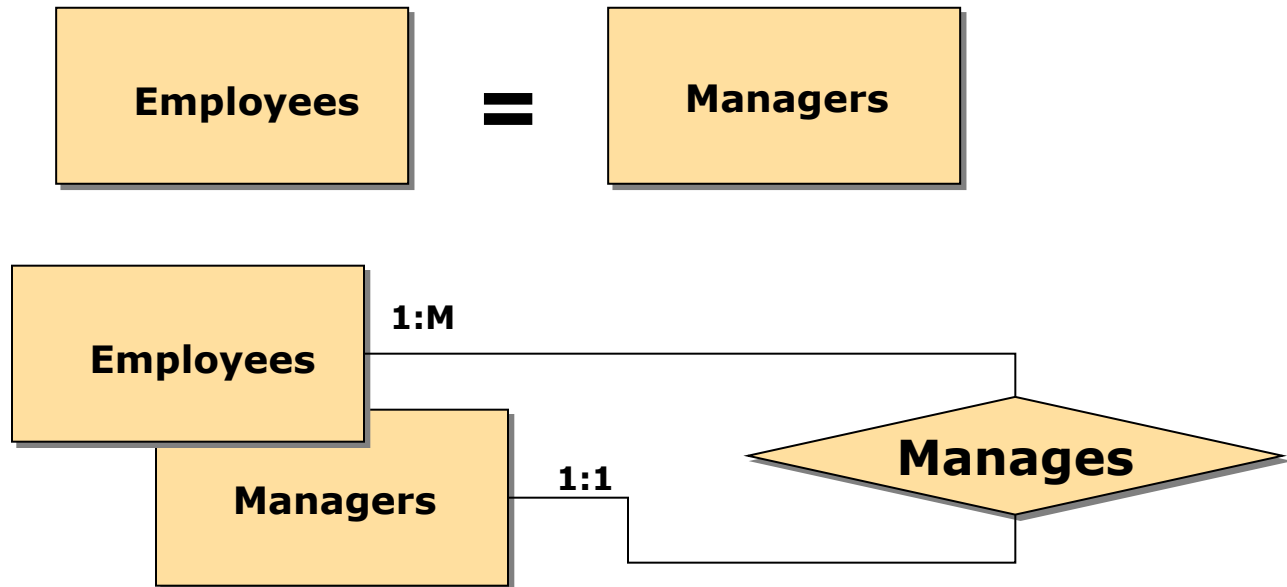
**List all Emps**

**List all Personnel**

EEnum	Name	Salary	DeptCode	CityCode	HireDate	Monthly Salary
1	Joseph Travolta	100000.00	MKT	NYC	19981012	8333.33
2	Mark Hirudsa	62000.00	R&D	OTT	19981212	5166.67
3	Frank Copolla	38000.00	R&D	OTT	19971208	3166.67
4	Mary Stuart	41500.00	MKT	TOR	19990204	3458.33
5	Nicholas Nick	26800.00	R&D	TOR	19970304	2233.33
10	Carl Santana		R&D	TOR	19981104	
10	Mark Chapman	102000.00	SAL	OTT	19980421	8500.00
11	Marshal Winduck	91800.00	MKT	NWJ	19980717	7650.00
12	Larry King	53800.00	R&D	NYC	19980916	4483.33
13	Alex Sander		R&D	OTT	19970912	
14	Lucy Markhan	33300.00	MKT	OTT	19981109	2775.00
15	Karin Lalonde	18600.00	R&D	OTT	19961207	1550.00

# Roles

**Roles can be used as synonyms for Entity Sets or Relationships with fields in reflexive relationships**

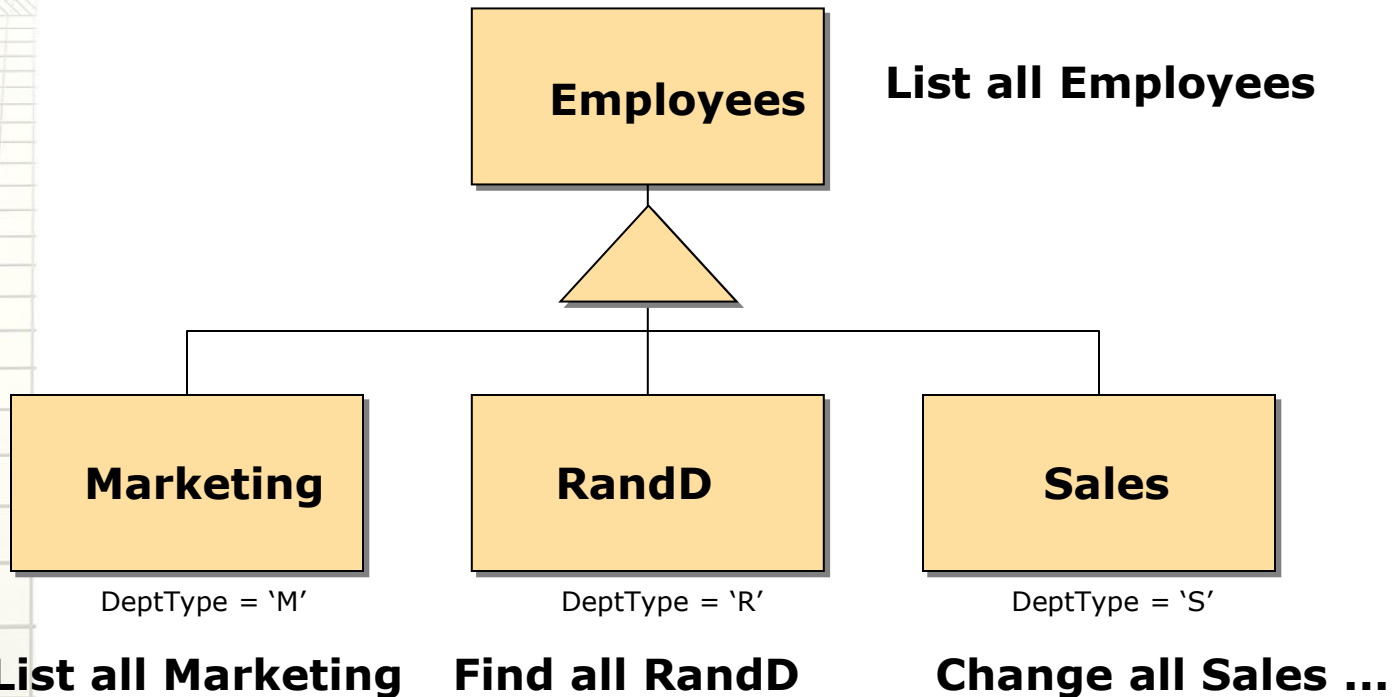


**Managers are Employees but their EmpType are "M" and "E", i. e., Managers is a Role with the Role Condition**

**EmpType = "M"**

# Roles

**By inference, Roles defines subtypes of entity sets or data relationships.**







# ZIM 9.10

## **Entity-Relationship Model (E-R Model)**