Employee

Attribute	Data type	Justification
Emp_id	Integer	Numbers only and smallint won't work in the long term.
Emp_fname	Varchar(20)	First names generally aren't longer than 20 characters
Emp_lname	Varchar(20)	Last names generally aren't longer than 20 characters
Emp_bdate	Date	It's a date
Emp_address	Varchar(100)	Includes street address and suburb
Gender	Char(1)	It's a one-character variable (M, F or U)
Dept_id	Integer	Numbers only. Don't know how many departments there are.
Super_id	Integer	Emp_id of superior. Can be null. The boss doesn't have a superior?

Department

Attribute	Data type	Justification
Dept_id	Integer	Numbers only. Don't know how many departments there are.
Dept_name	Varchar(100)	Department names can be long. Department of Environment, Land, Water and Planning has 51 characters.
Mgr_id	Integer	Numbers only. Don't know how many departments there are.
Mgr_start_date	Date	It's a date

Project

Attribute	Data type	Justification
Proj_id	Integer	Numbers only. Don't know how many projects there are are.
Proj_name	Varchar(20)	Project names shouldn't be too long.
Proj_loc	Varchar(100)	Either include street address and suburb or just the city.
Dept_id	Integer	Numbers only. Don't know how many departments there are.

Works_on

Attribute	Data type	Justification
Proj_id	Integer	Numbers only. Don't know how many projects there are.
Emp_id	Integer	Numbers only and smallint won't work in the long term.
Role	Varchar(20)	Name of the role
Comment	Varchar(255)	A brief description on their role and responsibilities. Can be null.
Hours	Smallint	Hours spent on a project shouldn't exceed smallint unless it's a lifetime project.
Salary_factor	Decimal(5,2)	Percentage of additional remuneration for working in their role. Check constraint (0.00% to 100.00%) Se the default to 0.00%

Salary

Attribute	Data type	Justification
Emp_id	Integer	Numbers only and smallint won't work in the long term.
Salary	Decimal(10,2)	Don't think anyone has an eight-figure annual salary.
Start_date	Date	It's a date

Relation	Primary key
Employee	Emp_id
Department	Dept_id
Project	Proj_id
Works_on	Proj_id, emp_id
Salary	Emp_id

Relation	Foreign Key	References
Employee	Dept_id	Department.dept_id
Works_on	Proj_id, emp_id	Project.proj_id, Employee.emp_id
Salary	Emp_id	Employee.emp_id

```sql
CREATE DATABASE test;
USE test;

CREATE TABLE Department
(
dept_id INT unsigned not null,
dept_name VARCHAR(100) not null,
mgr_id INT unsigned not null,
mgr_start_date DATE not null,
PRIMARY KEY(dept_id)
);

CREATE TABLE Employee
(
emp_id INT unsigned not null,
emp_fname VARCHAR(20) not null,
emp_lname VARCHAR(20) not null,
emp_bdate DATE not null,
emp_address VARCHAR(100) not null,
gender CHAR(1) CHECK(gender='M' OR gender='F' OR gender='U'),
dept_id INT unsigned not null,
super_id INT unsigned,
PRIMARY KEY(emp_id),
FOREIGN KEY(dept_id) REFERENCES Department(dept_id)
);

CREATE TABLE Project
(
proj_id INT unsigned not null,
proj_name VARCHAR(20) not null,
proj_loc VARCHAR(100) not null,
dept_id INT unsigned not null,
PRIMARY KEY(proj_id)
);

CREATE TABLE Works_on
(
proj_id INT unsigned not null,
emp_id INT unsigned not null,
role VARCHAR(20) not null,
comment VARCHAR(255),
hours SMALLINT unsigned not null,
salary_factor DECIMAL(5,2) DEFAULT 0.00 CHECK (salary_factor>=0.00 AND
salary_factor<=100.00),
PRIMARY KEY(proj_id, emp_id),
FOREIGN KEY(proj_id) REFERENCES Project(proj_id),
FOREIGN KEY(emp_id) REFERENCES Employee(emp_id)
);
```

```sql
CREATE TABLE Salary
(
emp_id INT unsigned not null,
salary DECIMAL(10,2) unsigned not null,
start_date DATE not null,
PRIMARY KEY(emp_id),
FOREIGN KEY(emp_id) REFERENCES Employee(emp_id)
);
```