create table SEAT (

SeatID integer not null,

RoomNumber varchar(30) not null,

SeatType varchar(30) null,

constraint SeatPK primary key(SeatID),

constraint SeatChk1 check(SeatType in ('opent-space','cubicle','enclosed office'))

);

create table COMPUTER (

ComputerID integer not null,

PurchaseDate date not null,

Os\_type varchar(30) not null,

SeatID integer not null,

constraint ComputerPK primary key (ComputerID),

constraint ComputerFK1 foreign key (SeatID)

references SEAT (SeatID)

);

create table SOFTWARE\_APPLICATION (

SoftwareID int not null,

SoftwareName varchar(50) not null,

VersionNumber varchar(50) not null,

Publisher varchar(50) null,

constraint Software\_ApplicationPK primary key(SoftwareID)

);

create table EMPLOYEE (

EmployeeID integer not null,

EmployeeFirstName varchar(50) not null,

EmployeeLastName varchar(50) not null,

EmailAddress varchar(100) not null,

Department varchar(50) not null,

PhoneNumber Integer not null,

constraint EmployeePK primary key(EmployeeID)

);

create table INSTALLED\_SOFTWARE (

Installed\_SoftwareID integer not null AUTO\_INCREMENT,

PurchaseDate Date not null,

Licence varchar(50) not null ,

ComputerID integer not null,

SoftwareID integer not null,

constraint InstalledPK primary key (Installed\_SoftwareID),

constraint InstalledAK1 unique (ComputerID,softwareID),

constraint InstalledFK1 foreign key(ComputerID) references COMPUTER (ComputerID),

constraint InstalledFK2 foreign key(SoftwareID) references software\_application (SoftwareID) );

create table REQUIRED\_SOFTWARE (

Required\_softwareID integer not null Auto\_increment,

SoftwareID int not null ,

EmployeeID int not null,

constraint required\_softwarePK primary key (Required\_softwareID),

constraint required\_softwareAK1 unique (SoftwareID,employeeID),

constraint required\_softwareFk1 foreign key(SoftwareID) references software\_application (SoftwareID),

constraint required\_softwareFk2 foreign key (EmployeeID) references employee (EmployeeID)

);

create table SEAT\_AVAILABILITY (

Seat\_AvailabilityID int not null Auto\_increment,

SeatID integer not null ,

Availability\_date Date not null,

SeatAvailablity\_Status varchar(50) not null,

Reserved\_Stause varchar(50) not null,

EmployeeID Integer null,

constraint Seat\_AvailabilityPK primary key(Seat\_AvailabilityID),

constraint Seat\_AvailabilityAK1 unique(SeatID, Availability\_date),

constraint Seat\_AvailabilityFK1 foreign key (SeatID) references seat (SeatID),

constraint Seat\_AvailabilityFK2 foreign key (EmployeeID) references employee(EmployeeID),

constraint Seat\_AvailabilityCHk1 check(Reserved\_Stause in ('reserved','not reserved')),

constraint Seat\_AvailabilityCHk2 check(SeatAvailablity\_Status in ('Available','not Available'))

);

**Number of seats by type**

select count(\*) as count , SeatType from seat group by SeatType;

**Popular software:**

select SoftwareName as Popular software, VersionNumber as Version, count( required\_software.SoftwareID) as NumberOFemployee

from required\_software,software\_application

where required\_software.SoftwareID= software\_application.SoftwareID

group by required\_software.SoftwareID

order by required\_software.SoftwareID asc;

**Open seat lookup:**

select distinct seat\_availability.SeatID as seatnumber, roomnumber as Room , SeatType as SEATTYPE from seat\_availability, seat,computer,installed\_software, required\_software,software\_application

where seat\_availability.SeatID= seat.SeatID and seat.SeatID= computer.SeatID and computer.ComputerID= installed\_software.ComputerID and installed\_software.SoftwareID= software\_application.SoftwareID

and required\_software.EmployeeID='?' and seat\_availability.SeatAvailablity\_Status like 'Available' and seat\_availability.Availability\_date like '?';