

ΑΣΚΗΣΗ ΑΞΙΟΛΟΓΗΣΗΣ ΕΠΑΡΚΕΙΑΣ ΜΗΧΑΝΙΚΩΝ ΛΟΓΙΣΜΙΚΟΥ

DB v3

Εισαγωγή

Σκοπός της εργασίας είναι να φτιαχτεί μια σχεσιακή βάση δεδομένων στην οποία το αντίστοιχο μοντέλο δεδομένων θα επιτρέπει τη διαχείριση των παρακάτω οντοτήτων:

- Ορισμός κατηγορίας χαρακτηριστικών (attributes)
- Ορισμός Εργαζομένων (employees)
- Ανάθεση Χαρακτηριστικών σε Εργαζομένους

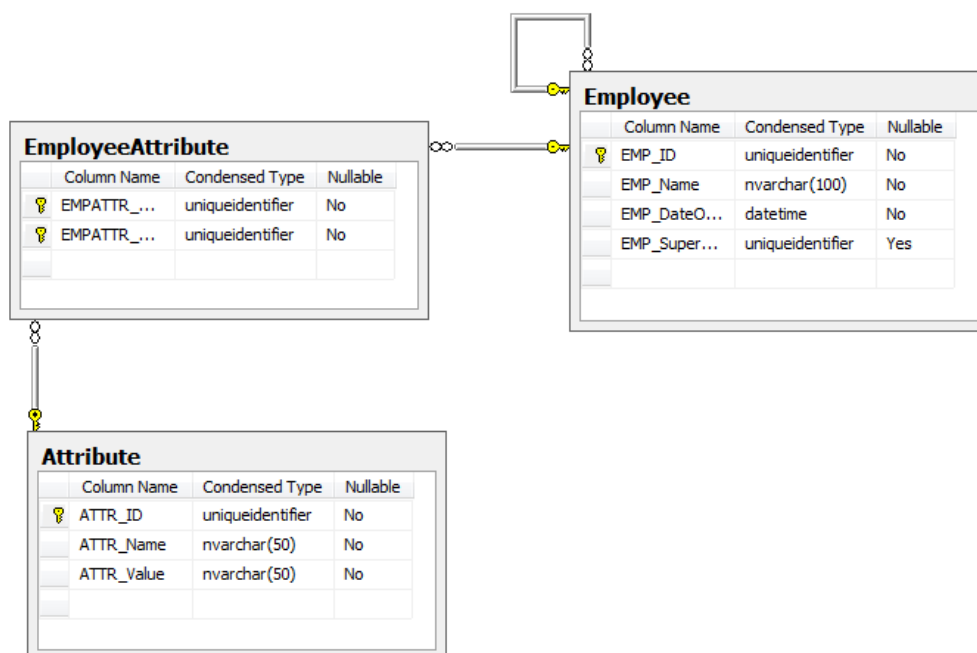
Διαδικασία εκτέλεσης / Αξιολόγησης

Κατά τη φάση της αξιολόγησης εξετάζονται μόνο πλήρεις υλοποιήσεις. Βαρύτητα έχει ο χρόνος υλοποίησης, αλλά μεγαλύτερη σημασία έχει η ποιότητά της, οπότε συνιστάται να μην προσπαθήσετε να την στείλετε όσο πιο γρήγορα μπορείτε έχοντας ως μόνο κριτήριο το χρόνο.

- Εφόσον παραλάβετε την άσκηση θα πρέπει να προσδιορίσετε τον χρόνο που θα χρειαστείτε ώστε να την εκτελέσετε
- Εφόσον επιβεβαιωθεί η διάρκεια, προχωράτε στην εκτέλεση της άσκησης
- Εφόσον ολοκληρώσετε την άσκηση μέσα στο χρόνο που αρχικά εκτιμήσατε ότι απαιτείται, στέλνετε τα αποτελέσματα μέσω email και αναμένετε επιβεβαίωση λήψης
- Εφόσον ο χρόνος και η ποιότητα εκτέλεσης καλύπτουν τις ελάχιστες απαιτήσεις της CITE, θα κληθείτε σε συνέντευξη

Μοντέλο Δεδομένων

Το παρακάτω σχήμα δείχνει τις οντότητες και τις αντίστοιχες συσχετίσεις τους



Το αντίστοιχο script για τη δημιουργία του μοντέλου δεδομένων σε T-SQL είναι το παρακάτω:

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Employee](
    [EMP_ID] [uniqueidentifier] NOT NULL,
    [EMP_Name] [nvarchar](100) NOT NULL,
    [EMP_DateOfHire] [datetime] NOT NULL,
    [EMP_Supervisor] [uniqueidentifier] NULL,
    CONSTRAINT [PK_Employee] PRIMARY KEY CLUSTERED
(
    [EMP_ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Attribute] *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Attribute](
    [ATTR_ID] [uniqueidentifier] NOT NULL,
    [ATTR_Name] [nvarchar](50) NOT NULL,
    [ATTR_Value] [nvarchar](50) NOT NULL,
    CONSTRAINT [PK_Attribute] PRIMARY KEY CLUSTERED
(
    [ATTR_ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[EmployeeAttribute](
    [EMPATTR_EmployeeID] [uniqueidentifier] NOT NULL,
    [EMPATTR_AttributeID] [uniqueidentifier] NOT NULL,
    CONSTRAINT [PK_EmployeeAttribute] PRIMARY KEY CLUSTERED
(
    [EMPATTR_EmployeeID] ASC,
    [EMPATTR_AttributeID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS
= ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO

ALTER TABLE [dbo].[Employee] WITH CHECK ADD CONSTRAINT [FK_Employee_Employee] FOREIGN
KEY([EMP_Supervisor])
REFERENCES [dbo].[Employee] ([EMP_ID])
GO
ALTER TABLE [dbo].[Employee] CHECK CONSTRAINT [FK_Employee_Employee]
GO

ALTER TABLE [dbo].[EmployeeAttribute] WITH CHECK ADD CONSTRAINT
[FK_EmployeeAttribute_Attribute] FOREIGN KEY([EMPATTR_AttributeID])
REFERENCES [dbo].[Attribute] ([ATTR_ID])
GO
ALTER TABLE [dbo].[EmployeeAttribute] CHECK CONSTRAINT [FK_EmployeeAttribute_Attribute]
GO

ALTER TABLE [dbo].[EmployeeAttribute] WITH CHECK ADD CONSTRAINT
[FK_EmployeeAttribute_Employee] FOREIGN KEY([EMPATTR_EmployeeID])
REFERENCES [dbo].[Employee] ([EMP_ID])
GO
ALTER TABLE [dbo].[EmployeeAttribute] CHECK CONSTRAINT [FK_EmployeeAttribute_Employee]
GO
```

Σαν seed αρχικών δεδομένων, μπορείτε να χρησιμοποιήσετε το παρακάτω script:

```
insert into Employee values('82D58D49-72A2-42B0-A250-471E5C10D7D9', 'Greg', GETUTCDATE(),
null)
insert into Employee values('8CEE7A83-A9EB-4170-B7E8-5D4F0440C074', 'Oleg', GETUTCDATE(),
'82D58D49-72A2-42B0-A250-471E5C10D7D9')
insert into Employee values('561E2D88-A747-460F-99E1-CFB1D3D8CA5C', 'Pete', GETUTCDATE(),
'8CEE7A83-A9EB-4170-B7E8-5D4F0440C074')
insert into Employee values('28106345-435B-4215-AECF-7C226C071E11', 'Paul', GETUTCDATE(),
'82D58D49-72A2-42B0-A250-471E5C10D7D9')
insert into Employee values('7012F5C7-33AD-4839-A092-4FA6E1448A5D', 'Aura', GETUTCDATE(),
'82D58D49-72A2-42B0-A250-471E5C10D7D9')
insert into Employee values('2E3074E7-8FFB-4C5F-83AE-962812F93D08', 'Phil', GETUTCDATE(),
'82D58D49-72A2-42B0-A250-471E5C10D7D9')

insert into Attribute values ('3C86A592-823B-4B83-952F-F437D08F2EA8', 'Height', 'Tall')
insert into Attribute values ('70C311F5-B2B0-4118-A069-3AB9C3AC65E1', 'Height', 'Short')
insert into Attribute values ('82FF24BB-0180-40F9-B68E-15799556A5C2', 'Height', 'Medium')
insert into Attribute values ('EB812BF6-3415-4686-A0B6-38089C87D09D', 'Height', 'Short')

insert into Attribute values ('83382664-DA55-4C6D-8D18-ED79C26332A8', 'Weight', 'Medium')
insert into Attribute values ('F27B9C58-FD9E-4EB1-9B09-E01FF7032CC8', 'Weight', 'Thin')
insert into Attribute values ('4F8EAC6B-8B29-4716-A597-C8CDE3A3996D', 'Weight', 'Heavy')

insert into EmployeeAttribute values ('82D58D49-72A2-42B0-A250-471E5C10D7D9', '3C86A592-823B-
4B83-952F-F437D08F2EA8')
insert into EmployeeAttribute values ('8CEE7A83-A9EB-4170-B7E8-5D4F0440C074', '70C311F5-B2B0-
4118-A069-3AB9C3AC65E1')
insert into EmployeeAttribute values ('561E2D88-A747-460F-99E1-CFB1D3D8CA5C', '82FF24BB-0180-
40F9-B68E-15799556A5C2')
insert into EmployeeAttribute values ('28106345-435B-4215-AECF-7C226C071E11', 'EB812BF6-3415-
4686-A0B6-38089C87D09D')

insert into EmployeeAttribute values ('2E3074E7-8FFB-4C5F-83AE-962812F93D08', '4F8EAC6B-8B29-
4716-A597-C8CDE3A3996D')
insert into EmployeeAttribute values ('8CEE7A83-A9EB-4170-B7E8-5D4F0440C074', 'F27B9C58-FD9E-
4EB1-9B09-E01FF7032CC8')
insert into EmployeeAttribute values ('82D58D49-72A2-42B0-A250-471E5C10D7D9', '83382664-DA55-
4C6D-8D18-ED79C26332A8')
```

Αιτούμενα

Στα πλαίσια της εργασίας, τα παρακάτω tasks πρέπει να ολοκληρωθούν:

- Create a new Database on the database platform of choice. Name it as you prefer.
- Comprehend and modify/execute the schema scripts to create the tables and constraints depending on the database platform selected.
- Comprehend and modify/execute the data scripts to populate the database with initial data.
- Author the following SQL Scripts on the platform of choice (SQL Server or PostgreSQL preferred):
 - Create a script that will add an attribute of type 'Weight' and with a value set to 'Thin' to all employees. If an employee already has a 'Weight' attribute, update that attribute, otherwise create a new one.
 - Create a script that will add an attribute of type 'Height' and with a value set to 'Short' to all employees that are supervisors of anybody else. If an employee already has a 'Height' attribute, update that attribute, otherwise create a new one.
 - Create a script that will expect whoever runs it to provide an employee id. Locate all employees that the provided employee supervises in any depth. Add to these employee an attribute with a type of 'Team' and a value of the name of the originally provided supervising employee.

- Notes
 - All scripts must handle transactions properly, print progress messages and reporting at the end with overview of what happened to how many number of rows etc (not too detailed).
 - Initialization scripts are provided in T-SQL dialect.
 - All scripts (initialization and new ones) must be provided back for evaluation.

Καλή επιτυχία!
