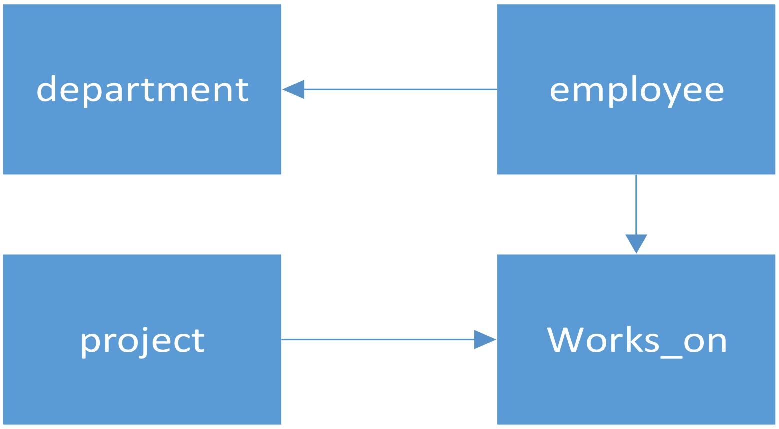
Readings:

- ACID Property: https://en.wikipedia.org/wiki/ACID_(computer_science)
- Concurrency control https://en.wikipedia.org/wiki/Concurrency_control
- chapter 13: "Concurrency Control" (Petkovic)

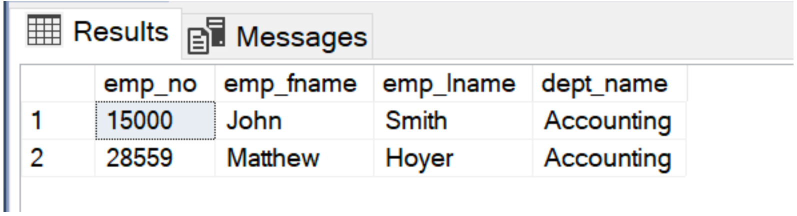
Use the sample database created in previous lecture to answer the following three questions.



1) Create a stored procedure GetEmployeeInfo which takes @dept_no as an input parameter, and outputs a result set which includes the following fields: emp_no, employee full name, department name. Provide a screenshot of output results using 'd1' as input parameter.

```
create procedure GetEmployeeInfo  @d1 char(4)
as
select e.emp_no, e.emp_fname, e.emp_lname, d.dept_name
from employee e join department d on e.dept_no=d.dept_no
where e.dept_no = @d1;

exec GetEmployeeInfo  @d1 = 'D1';
```



	emp_no	emp_fname	emp_lname	dept_name
1	15000	John	Smith	Accounting
2	28559	Matthew	Hoyer	Accounting

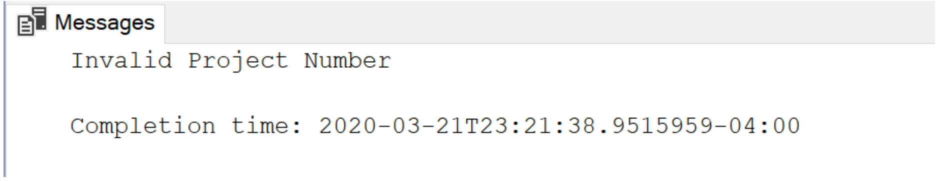
2) Create a stored procedure IncreaseBudgetAmount which takes @project_no and @new_budget as input parameters and returns @message as an output parameter. The stored procedure must perform the following business rules:

   a. If @project_no is not found, it returns the message "Invalid Project Number"
   b. If @new_ budget is greater than the current budget amount, it must update the project budget and return the message "budget amount increased"
   c. If @new_ budget is less than or equal to the current budget, it does nothing and return the message "New budget must be greater than the current budget"

Provide sample execution commands for all three business cases, along with a screen shot of results for each.

```
create procedure IncreaseBudgetAmount @project_no char(4), @new_budget float
as
begin
        if not exists (select * from project where project_no = @project_no)
        begin
                print 'Invalid Project Number'
        end
        else
        begin
                if @new_budget > (select budget from project where project_no =
@project_no)
                begin
                        update project set budget = @new_budget where project_no =
@project_no
                        print 'budget amount increased'
                end
                else
                begin
                        print 'New budget must be greater than the current budget'
                end
        end
end
```

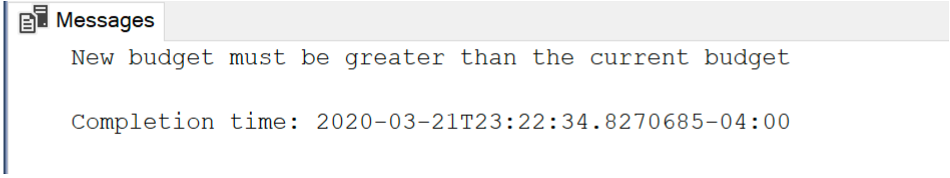
```
exec IncreaseBudgetAmount @project_no = 'p8', @new_budget = 10000.00;
```



```
Invalid Project Number

Completion time: 2020-03-21T23:21:38.9515959-04:00
```

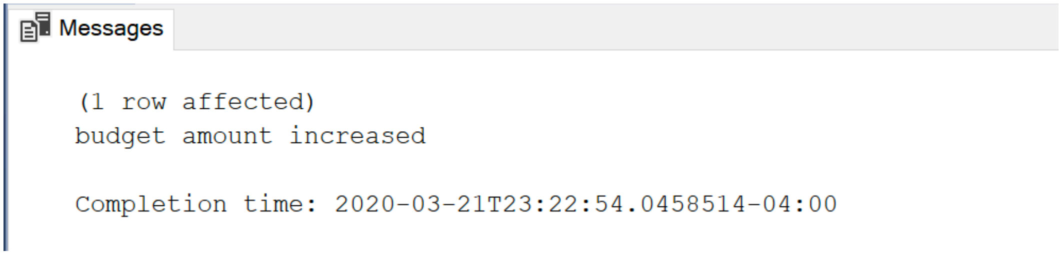
```
exec IncreaseBudgetAmount @project_no = 'p1', @new_budget = 10000.00;
```



```
New budget must be greater than the current budget

Completion time: 2020-03-21T23:22:34.8270685-04:00
```

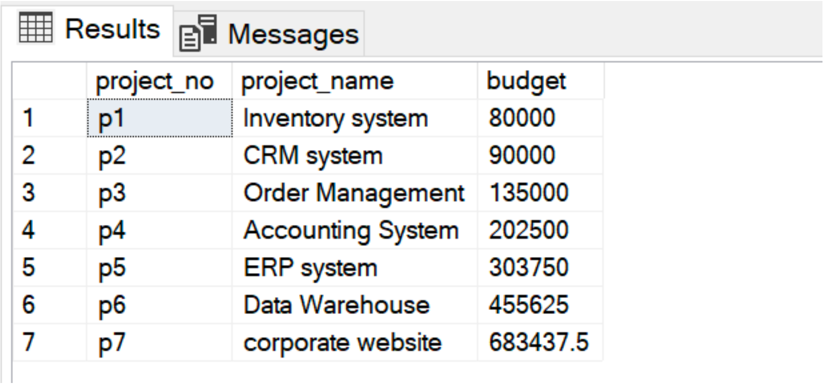
```
exec IncreaseBudgetAmount @project_no = 'p1', @new_budget = 80000.00;
```



```
(1 row affected)
budget amount increased

Completion time: 2020-03-21T23:22:54.0458514-04:00
```

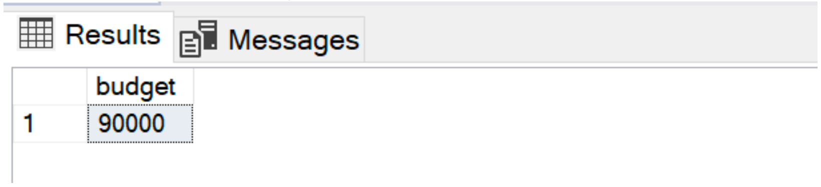
```
select * from project;
```



	project_no	project_name	budget
1	p1	Inventory system	80000
2	p2	CRM system	90000
3	p3	Order Management	135000
4	p4	Accounting System	202500
5	p5	ERP system	303750
6	p6	Data Warehouse	455625
7	p7	corporate website	683437.5

3) Create a User Defined Function GetBudgetAmount which takes @project_name and returns the budget for a given project. If it cannot find the record it returns NULL. Show a SQL example of a function being used to the budget for "CRM system"

```
create function GetBudgetAmount (@project_name varchar(50))
returns table
as return (select budget from project where project_name = @project_name);
select * from GetBudgetAmount('CRM system');
```



	budget
1	90000

4) What is the name of a single logical operation on the data to satisfy ACID property?
Transaction.

5) Which ACID property does the following DDLs satisfy?
   a. CREATE TABLE Customer (CustomerID int PRIMARY KEY, CustomerName varchar(100) NOT NULL)

Consistency, Durability.

6) Which ACID property ensures the integrity of data reads?
Consistency.

7) Failure to write data to non-volatile memory violates which property?**d**
   a. Atomicity
   b. Consistency
   c. Isolation
   d. Durability

8) State the reasons why concurrency control needed?
Concurrency control in Database management systems (DBMS; e.g., Bernstein et al. 1987, Weikum and Vossen 2001), other transactional objects, and related distributed

applications (e.g., Grid computing and Cloud computing) ensures that database transactions are performed concurrently without violating the data integrity of the respective databases. Thus concurrency control is an essential element for correctness in any system where two database transactions or more, executed with time overlap, can access the same data, e.g., virtually in any general-purpose database system.

9) What is the difference between a local transaction and a distributed transaction?

Local transactions are performed on a single database table, but distributed transactions are performed on more than one database tables. b. Local transactions are performed on a single database server, but distributed transactions can be performed across multiple database servers.

10) When should you use the SAVE TRANSACTION statement?

The Database Engine supports optimistic concurrency so that older versions of data rows are saved, and any process that reads the same data uses the row version that was active when it started reading data. For that reason, a process that modifies the data can do so without any limitation, because all other processes that read the same data access the saved versions of the data. The only conflict scenario occurs when two or more write operations use the same data. In that case, the system displays an error so that the client application can handle it.

11) Discuss the difference between row-level and page-level locking.

A row lock is the lowest level of granularity of locking possible in SQL Server. This means one or more specific rows will be locked, and the adjacent rows are still available for locking by concurrent queries. A page lock in SQL Server will lock 8K worth of data even when your query only needs 10 bytes from the page.

12) Can a user explicitly influence the locking behavior of the system?

Yes. User can either locking hints or the LOCK_TIMEOUT option of the SET statement to affect locks.