

Version: 03.01

Date: 10-Mar-2016

Developed by:

Verified by:

Endava SQL Discipline

SQL Test

# Revision History

|  |  |  |
| --- | --- | --- |
| **Revision** | **Date of revision** | **Description of modifications** |
| 03.01 | 10-MAR-2016 | The third version of the document. |

# Description

**About the test:**

|  |  |
| --- | --- |
|  |  |
| Applied Level | Basic |
| Number of tasks | 18 |
| Domain | Standard DB |
| Test type | PC |
| Test duration | 2 h |

**Evaluation info:**

|  |  |
| --- | --- |
|  |  |
| Evaluated person |  |
| Evaluator name |  |
| Date of evaluation |  |
| Evaluation result  (passed/failed) |  |

# Test Tasks

## Precondition:

Use the below credentials:

|  |  |
| --- | --- |
| **Server Name** | MDCH-AMWCI-S01 |
|
| **DB** | AMInternship |
|
|

**NOTE: 1. Do not use JOIN on the below queries.**

**2. Execute the Task 1 – Task 7 on AMInternship\_YourNameSurname db;**

**Execute the Task 10 – Task 18 on AMInternship db.**

**3. Execute the Task 10 – Task 18 on AMInternship\_YourNameSurname db(Only for IOS);**

## Task 1

**Insert 20 records in each created table from your database.**

**\*You can use the information from the original tables, db AMInternship.**

**Query:**

|  |
| --- |
| /\*person\*/  Insert into AMInternShip\_AlexPershin.dbo.person  (first\_name, last\_name,date, gender\_id, address\_id,job\_id)  select  first\_name, last\_name,date, gender\_id, address\_id,job\_id  from AMInternship.dbo.person where AMInternship.dbo.person.id<=20  select \* from AMInternShip\_AlexPershin.dbo.person  /\*country\*/  Insert into AMInternShip\_AlexPershin.dbo.country  select  \*  from AMInternShip.dbo.country where AMInternShip.dbo.country.id<=20  /\*job\_title\*/  select \* from AMInternShip\_AlexPershin.dbo.job\_title  Insert into AMInternShip\_AlexPershin.dbo.job\_title  (title)  select  title  from AMInternShip.dbo.job\_title where AMInternShip.dbo.job\_title.id<=20  /\*company\*/  select \* from AMInternShip\_AlexPershin.dbo.company  Insert into AMInternShip\_AlexPershin.dbo.company  (name)  select  name  from AMInternShip.dbo.company where AMInternShip.dbo.company.id<=20  /\*job\*/  select \* from AMInternShip\_AlexPershin.dbo.job  Insert into AMInternShip\_AlexPershin.dbo.job  (jobtitle\_id, company\_id,salary\_id,workplace\_id)  select  jobtitle\_id, company\_id,salary\_id,workplace\_id  from AMInternShip.dbo.job where AMInternShip.dbo.job.id<=20  /\*address\*/  select \* from AMInternShip\_AlexPershin.dbo.address  Insert into AMInternShip\_AlexPershin.dbo.address  (city\_id, street,street\_num,post\_code)  select  city\_id, street,street\_num,post\_code  from AMInternShip.dbo.address where AMInternShip.dbo.address.id<=20  /\*gender\*/  select \* from AMInternShip\_AlexPershin.dbo.gender  Insert into AMInternShip\_AlexPershin.dbo.gender  (name)  select  name  from AMInternShip.dbo.gender where AMInternShip.dbo.gender.id<=20  /\*salary\*/  select \* from AMInternShip\_AlexPershin.dbo.salary  Insert into AMInternShip\_AlexPershin.dbo.salary  (salary,currency)  select  salary,currency  from AMInternShip.dbo.salary where AMInternShip.dbo.salary.id<=20  /\*workplace\*/  select \* from AMInternShip\_AlexPershin.dbo.workplace  Insert into AMInternShip\_AlexPershin.dbo.workplace  (username,mac\_address,ip\_address,email)  select  username,mac\_address,ip\_address,email  from AMInternShip.dbo.workplace where AMInternShip.dbo.workplace.id<=20  /\*city\*/  select \* from AMInternShip\_AlexPershin.dbo.city  Insert into AMInternShip\_AlexPershin.dbo.city  (country\_id, name)  select  country\_id, name  from AMInternShip.dbo.city where AMInternShip.dbo.city.id<=20 |

## Task 2

**Display full details for the persons from [dbo].[Person] table**.

**Query:**

|  |
| --- |
| select \* from person |

## Task 3

**Display only first name and address in an ascending order by first name from [dbo].[Person ].**

**Query:**

|  |
| --- |
| SELECT  first\_name,  (  SELECT street  FROM address  WHERE id = person.address\_id  )  FROM  person  order by first\_name ASC |

## Task 4

**Display all the persons ordered ascending by the first name and descending by the last name using the table [dbo].[Person ].**

**Query:**

|  |
| --- |
| select \* from person order by first\_name ASC , last\_name DESC |

## Task 5

**Insert the result set from task 2 into a new table.**

**Query:**

|  |
| --- |
| select \*  into newPerson  from person |

## Task 6

**Update the record 15 from [dbo].[Person ] with the information: first name – Camelia, last name – Agachi, gender, address „Kingsford 8” and the job titile „Environmental Tech”.**

**Query:**

|  |
| --- |
| update person  Set first\_name='Camelia',  last\_name='Aghachi'  address= 'Kingsford 8',  job\_titile ='Environmental Tech'  where id =15 |

## Task 7

**Delete ONLY the updated record from Task 6.**

**Query:**

|  |
| --- |
| delete from person  where first\_name='Camelia' |

## Task 8

**Write down the general syntax for a SELECT statements covering all the options.**

**Answer:**

|  |
| --- |
| <SELECT statement> ::=  [ WITH { [ XMLNAMESPACES ,] [ <common\_table\_expression> [,...n] ] } ]  <query\_expression>  [ ORDER BY { order\_by\_expression | column\_position [ ASC | DESC ] }  [ ,...n ] ]  [ <FOR Clause>]  [ OPTION ( <query\_hint> [ ,...n ] ) ]  <query\_expression> ::=  { <query\_specification> | ( <query\_expression> ) }  [ { UNION [ ALL ] | EXCEPT | INTERSECT }  <query\_specification> | ( <query\_expression> ) [...n ] ]  <query\_specification> ::=  SELECT [ ALL | DISTINCT ]  [TOP ( expression ) [PERCENT] [ WITH TIES ] ]  < select\_list >  [ INTO new\_table ]  [ FROM { <table\_source> } [ ,...n ] ]  [ WHERE <search\_condition> ]  [ <GROUP BY> ]  [ HAVING < search\_condition > ] |

## Task 9

**What does mean the word DISTINCT in SELECT query? Give examples.**

**Answer:**

|  |
| --- |
| The SELECT DISTINCT statement is used to return only distinct (different) values.  Inside a table, a column often contains many duplicate values; and sometimes you only want to list the different (distinct) values. |

## Task 10

**Display first name and last name from the [dbo].[Person] table, where gender is male.**

**Query:**

|  |
| --- |
| select first\_name, last\_name from person  where person.gender\_id in  (select id from gender  where gender.name = 'male') |

## Task 11

**Display all the streets and street numbers from Vienna.**

**Query:**

|  |
| --- |
| select street, street\_num from address  where address.city\_id in  (select id from city  where city.name = 'Vienna') |

## Task 12

**Display all the details of the persons who live in Vienna.**

**Query:**

|  |
| --- |
| select \* from person  where person.address\_id in  (select id from address  where address. city\_id in ( (select id from city  where city.name = 'Vienna'))) |

## Task 13

**Write a query which returns all the data from [dbo].[Job] table, where salary is bigger than 10 and less than 1000 monetary units.**

**Query:**

|  |
| --- |
| select \* from job  where job.salary\_id in  (select id from salary  where salary.salary between 10 and 1000  ) |

## Task 14

**Write the task 13 using the comparison operators.**

**Query:**

|  |
| --- |
| select \* from job  where job.salary\_id in  (select id from salary  where salary.salary > 10 AND salary.salary < 1000 ) |

## Task 15

**Display all the persons who were born in 1986 year.**

**Query:**

|  |
| --- |
| select \* from person where DATEPART(yyyy,date) =1986 |

## Task 16

**Display the persons who were born in 1986 and their first name starts with “L” and the third character is “u”.**

**Query:**

|  |
| --- |
| select \* from person where DATEPART(yyyy,date) =1986 AND first\_name like 'L\_u%' |

## Task 17

**Display the persons that have the salary bigger than 500 monetary units.**

**Query:**

|  |
| --- |
| select \* from person where person.job\_id in ( select id from job  where job.salary\_id in  (select id from salary  where salary.salary >500 )) |

## Task 18

**Display which persons from San Jose have the salary bigger than 500 monetary units.**

**Query:**

|  |
| --- |
| select \* from person  where person.address\_id in  (select id from address  where address.city\_id in ( select id from city where city.name = 'San Jose') )  AND  person.job\_id in  ( select id from job  where job.salary\_id in  (select id from salary  where salary.salary >500 )) |