

ПРАКТИКАЛЫҚ ЖҰМЫС №5. СҰРАНЫСТАРДЫ ҚҰРУ. ШАРТ БОЙЫНША ЖАЗБАЛАРДЫ ТАҢДАУ

5.1. Практикалық жұмыстың мақсаты

Реляциялық МҚБЖ-нде қолданылатын мәліметтерді таңдау операторының қызметін үйрену. 'SQL Server Managment Studio' ортасында SELECT операторымен жұмыс жасауға машықтану.

5.2. Кіріс мәліметтер

Жеке тапсырма бойынша мәліметтер және алдыңғы практикалық жұмыс нәтижелері.

5.3. Қолданылатын программалар

'SQL Server Managment Studio' программасы.

5.4. Теориялық мәлімет

SELECT операторы **DML** жиынына кіреді. **SELECT** операторының толық жазылуы

```
SELECT [DISTINCT | ALL]  
{* | <өлшем> [, <өлшем> ...]}  
[INTO :айнымалы [, :айнымалы ...]]  
FROM <tableref> [, <tableref> ...]  
[WHERE <іздеу шарты>]  
[GROUP BY Өріс [, Өріс ...]]  
[HAVING <іздеу шарты>]  
[UNION [ALL] <select_expr>]  
[ORDER BY <сұрыптау тізімі>];
```

<өлшем> = {Өріс | :Айнымалы | <тұрақты>
| <өрнек> | <функция>
| udf ([<өлшем> [, <өлшем> ...]])
| **NULL** | **USER**} [**AS** Псевдоним]

<константа> = Сан | 'Жол'

<өрнек> = бір ғана мән қайтаратын SQL өрнек

<функция> =
COUNT (* | [**ALL**] < өлшем> | **DISTINCT** < өлшем >)
| **SUM** ([**ALL**] < өлшем > | **DISTINCT** < өлшем >)
| **AVG** ([**ALL**] < өлшем > | **DISTINCT** < өлшем >)
| **MAX** ([**ALL**] < өлшем > | **DISTINCT** < өлшем >)
| **MIN** ([**ALL**] < өлшем > | **DISTINCT** < өлшем >)
| **CAST**(<өлшем > **AS** <мәлімет
типi>) | **UPPER** (<өлшем >)

| **GEN_ID** (Генератор_Атауы, < өлшем >)

<tableref> = {<joined_table> | table | view
| procedure[(<өлшем > [, < өлшем > ...])]}
[Псевдоним]

<joined_table> = <tableref> <join_type> **JOIN** <tableref>
ON <условие поиска> | (<joined_table>)

<join_type> = [**INNER**] | {**LEFT** | **RIGHT** | **FULL** } [**OUTER**]

<іздеу шарты> =
< өлшем > <салыстыру операторлары>
{< өлшем > | (<select_one>)}
| < өлшем > [**NOT**] **BETWEEN** < өлшем > **AND** < өлшем >
| < өлшем> [**NOT**] **LIKE** < өлшем >
| < өлшем > [**NOT**] **IN**
(<өлшем> [, < өлшем > ...] | <select_list>)
| < өлшем > **IS** [**NOT**] **NULL**
| < өлшем > {>= | <= } < өлшем >
| < өлшем > [**NOT**] {= | < | > } < өлшем >
| {**ALL** | **SOME** | **ANY**} (<select_list>)
| **EXISTS** (<select_expr>)
| **SINGULAR** (<select_expr>)
| < өлшем > [**NOT**] **CONTAINING** < өлшем >
| < өлшем > [**NOT**] **STARTING** [**WITH**] < өлшем >
| (<іздеу шарты>)
| **NOT** <іздеу шарты>
| <іздеу шарты>**OR** <іздеу шарты> |
<іздеу шарты>**AND** <іздеу шарты>

<салыстыру операторлары> =
{= | < | > | <= | >= | != | < | != | <> | !=}

<select_one> = бір өрісті таңдап, бір ғана мән қайтаратын SELECT операторы

<select_list> = бір өрісті таңдап, нөл немесе бірнеше мән қайтаратын SELECT операторы

<select_expr> = бірнеше өлшемді таңдап, бірнеше мән қайтаратын SELECT операторы

<сұрыптау тізімі> =

{Өріс | Номер}
 [ASC | DESC]
 [, <сұрыптау тізімі> ...]

5.1. кестеде параметрлер сипатталған.

| Параметр | Сипаттамасы |
|---|---|
| DISTINCT ALL | DISTINCT – қайталанатын мәндерді алып тастайды. ALL (келісім бойынша) – барлық мәнді шығарады. |
| {* <өлшем> [, <өлшем> ...]} | (*) -барлық өрістер, өлшем-өріс атаулары |
| INTO :Айнымалы [, :Айнымалы ...] | Триггерлер мен процедураларда қолданылады. Біреуден артық жол қайтарылмайды. Айнымалылар тізім көрсетіледі. |
| FROM <tableref> [, <tableref> ...] | Кесте атаулары |
| table | Кесте аты |
| view | Қаралым атауы |
| procedure | SELECT операторында қолданылатын процедура атауы |
| Псевдоним | Қысқаша атауы |
| join_type | Бірігу түрін көрсетеді: сыртқы немесе ішкі |
| WHERE <шарт> | Шарт көрсетіледі |
| GROUP BY Баған[, Баған ...] | Сұраныс нәтижесін топтарға бөледі, яғни топтау операторы |
| HAVING <іздеу шарты> | GROUP BY операторымен бірге қолданылады. Топтау шартын береді. |
| UNION [ALL] | Бірнеше сұранысты біріктіреді. Барлық сұраныстың бағандар саны бірдей болу керек. |

| | |
|--|---|
| ORDER BY <список сортировки> | Сұрыптау орындалады. ASC-өсу ретімен, DESC-кему ретімен сұрыптайды. |
|--|---|

Кесте 5.1 **SELECT** операторының параметрлерінің сипаттамасы

Мысал:

-- қызметкерлердің тізімін шығару:

SELECT * FROM Employee;

Ішкі бірігудің қысқартылып жазылуы(стандарт SQL-92):

SELECT Өріс аты [, Өріс аты ...] | *

FROM <tableref_left> [**INNER**] **JOIN** <tableref_right>

[**ON** <условие поиска>]

[**WHERE** <условие поиска>];

Сыртқы бірігудің қысқартылып жазылуы:

SELECT Өріс аты [, Өріс аты ...] | *

FROM <tableref_left>

{**LEFT** | **RIGHT** | **FULL**} [**OUTER**] **JOIN**

<tableref_right>

[**ON** <бірігу шарты>]

[**WHERE** <іздеу шарты>];

Іштестірілген сұраныстардың қысқартылып жазылуы:

SELECT [**DISTINCT**] Өріс аты [, Өріс аты ...]

FROM <tableref> [, <tableref> ...]

WHERE

{expression {[**NOT**] **IN** | <салыстыру операторы>}

| [**NOT**] **EXISTS**

}

(**SELECT** [**DISTINCT**] Өріс аты [, Өріс аты ...]

FROM <tableref> [, <tableref> ...]

WHERE <іздеу шарты>

)

;

5.5. Тапсырма

1. ‘University.mdf’ мәліметтер қоры негізінде **SELECT** операторының жазылу синтаксисін үйрену.

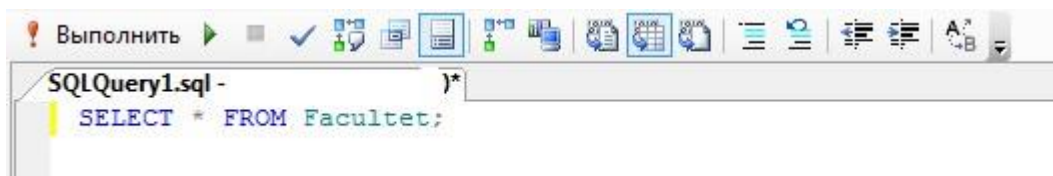
2. **Мәліметтер қорының 27 сұранысы құрылып, дәптерге кодымен бірге түсіндірме сипаттамасы жазылуы керек!!!**

5.6. Жұмысты орындау реті

Ескерту. Университет мәліметтер қорының барлық кестелері құрылып, мәліметтермен толтырылған болу керек

sql-сұраныстарды орындау

1. Сұраныстарды жазу үшін ‘Создать запрос’ батырмасы арқылы сұранысты жазу терезесін ашыңыз.



Сурет. 1. Сұранысты жазу терезесі

2.  [Выполнить] . батырмасының көмегімен сұраныс орындауға жіберіледі.

Қарапайым сұраныстарды құру

Салыстыру операторлары

Сұраныс 1. Профессорлардың фамилияларын шығару.

```
SELECT NAME_TEACHER AS 'Профессорлар тізімі'  
FROM TEACHER  
WHERE DOLGNOST = 'профессор';
```

 батырмасы арқылы орындауға жіберіңіз.

Сұранысты өзгертіп көріңіз:

```
WHERE LOWER(DOLGNOST) = 'профессор'  
WHERE UPPER(DOLGNOST) = 'ПРОФЕССОР'
```

Сұраныс 2. 30000тг -ден жоғары стипендия алатын студенттерді шығару.

```
SELECT STUDNAME, STUFDNAME  
FROM STUDENT WHERE STIPEND>30000;
```

Сұраныс 3. 01.01.2022 кейін алынған оқытушылардың фамилиясын және қызметін шығару.

```
SELECT NAME_TEACHER AS 'Фамилия', DOLGNOST AS 'Қызметі'  
FROM TEACHER  
WHERE DATA_HIRE > '1/01/2022';
```

Сұраныс 4. Алфавит бойынша Саменова фамилиясынан кейін орналасқан оқытушылардың фамилиясын және қызметін шығару.

```
SELECT NAME_TEACHER, DOLGNOST  
FROM TEACHER  
WHERE UPPER(NAME_TEACHER) > 'Саменова';
```

Сұраныс 5(Өз бетінше). Үстемақысы 2,5-нан төмен болатын оқытушылардың фамилиясын шығару.

**Логикалық операторлар. AND, OR және NOT операторлары
AND операторын қолдану**

Сұраныс 6. Ақтөбе қаласында тұратын және стипендиясы 40000тг-ден жоғары болатын студенттерді шығару

```
SELECT SUTFNAME  
FROM STUDENT  
WHERE CITY = 'Актобе' AND STIPEND >40000;
```

Сұраныс 7(Өз бетінше). 4500-ден жоғары және қызметі профессор болатындарды шығару.

Сұраныс 8(Өз бетінше). Реттік номері 2 және стипендиясы 10000-50000тг аралығында болатын студенттердің фамилияларын шығару.

OR(немесе) логикалық операторын қолдану

Сұраныс 9. 1 немесе 8 корпусқа орналасқан кафедраларды шығару.

```
SELECT NAME_KAFEDRU, NUM_KORPUSA  
FROM KAFEDRA  
WHERE NUM_KORPUSA =1 OR NUM_KORPUSA =8;
```

NOT(емес) логикалық операторын қолдану

Сұраныс 10. Физика -математика факультетінен басқа факультеттердің тізімін шығару.

```
SELECT NAME_FACULTETA  
FROM FACULTET  
WHERE NOT LOWER(NAME_FACULTETA) = 'физика-математика';
```

Логикалық операторларды бірге қолдану

Сұраныс 11. Ставкасы 550-ден немесе үстемақысы 60-тан жоғары болатын ассистенттердің фамилиясын, қызметін, ставкасын және үстемақысын шығару

```
SELECT NAME_TEACHER, DOLGNOST, Salary, Rise  
FROM TEACHER
```

**WHERE LOWER(DOLGNOST) ='ассистент' AND
(Salary < 550 OR Rise > 60);**

Өрнектерді қолдану

Сұраныс 12. Жалақысы(ставка плюс үстемақы) 350000-нан жоғары болатын оқытушылардың тізімін шығару.

**SELECT NAME_TEACHER AS 'Оқытушы фамилиясы',
Salary + Rise AS 'Жалаңысы'
FROM TEACHER
WHERE Salary + Rise > 3500;**

Сұраныс 13. Жалақысының жартысы үстемақысының 5 еселенген мәнінен асатын оқытушылардың фамилиясын шығарыңыз.

**SELECT NAME_TEACHER
FROM TEACHER
WHERE (Salary + Rise) / 2 > 5 * Rise;**

Арнайы операторларды қолдану

Арнайы операторлар тізімі:

- Жиынға тиістілік;
- Диапазонға тиістілік;
- Шаблонға сәйкестік;
- Арнайы өрнектерге сәйкестігіні тексеру;
- Белгісіз мән.

Жиынға тиістілік

IN операторы жиынға тиістілікті тексереді. Жазылуы:

өріс_атауы [NOT] IN (мәндер_тізімі)

Сұраныс 14. 1, 3, 12 корпусарында орналасқан кафедралардың атауылары мен корпус номерлерін шығару.

**SELECT Name_Kafedru, NUM_KORPUSA
FROM KAFEDRA
WHERE NUM_KORPUSA IN ('1', '3', '12');**

Терістеуді қолдану

NOT IN

Сұраныс 15. 1, 3 немесе 12 болмайтын кафедралардың атаулары мен номерін шығару.

**SELECT Name_Kafedru AS 'Кафедра атауы',
NUM_KORPUSA AS 'Корпус'
FROM KAFEDRA**

WHERE NUM_KORPUSA NOT IN ('1', '3', '12');

Өрнекті қолдану

Сұраныс 16. Жалақысы (ставка + үстемақы) 800, 900, 1000, 1100 немесе 1200 болатын оқытушыларды шығару.

**SELECT NAME_TEACHER AS 'Фамилия',
Salary + Rise AS 'Жалақы'
FROM TEACHER
WHERE Salary + Rise IN (1150, 2400, 3150, 4300);**

Сұраныс 17.

**SELECT NAME_TEACHER, Salary, Salary + Rise
FROM TEACHER
WHERE Salary + Rise IN (Salary + 100, Salary + 200, Salary + 300, Salary + 400,
Salary + 500);**

Диапазонға тиістілікті тексеру

Жиынға тиістілікті тексерудің тағы біреуі BETWEEN операторы. Жазылуы:

өріс_атауы [NOT] BETWEEN минимум AND максимум

Сұраныс 18. Ставкасы 1000-2000 аралығында болатын оқытушыларды шығару.

**SELECT NAME_TEACHER
FROM TEACHER
WHERE Salary BETWEEN 1000 AND 2000;**

Жолдық мәндерді қолдану

Сұраныс 19. Фамилиясы 'З' мен 'М' аралығында болатын оқытушыларды шығару.

**SELECT NAME_TEACHER
FROM TEACHER
WHERE UPPER(NAME_TEACHER) BETWEEN 'Z' AND 'M';**

Терістеуді қолдану

Жазылуы:

NOT (өріс_атауы BETWEEN минимум AND максимум)

Сұраныс 20. 1 мен 3 аралығында болмайтын кафедралардың атауы мен корпус номерін шығарыңыз.

**SELECT Name_Kafedru, NUM_KORPUSA
FROM KAFEDRA
WHERE NUM_KORPUSA NOT BETWEEN '1' AND '3';**


```
SELECT Name_Kafedru, NUM_KORPUSA
FROM KAFEDRA
WHERE NOT (NUM_KORPUSA BETWEEN '1' AND '3');
```

Өрнектерді қолдану

Сұраныс 21. 01.01.2000 және 12.12.2001 аралығында жұмысқа қабылданған оқытушылардың тізімін шығару.

```
SELECT NAME_TEACHER, DATA_HIRE
FROM TEACHER
WHERE DATA_HIRE BETWEEN '01/01/2000' AND '12/12/2001';
```

Сұраныс 22. Вывести данные преподавателей, зарплата которых (ставка + надбавка) находится в диапазоне от удвоенной величины надбавки до утроенной надбавки плюс 50.

```
SELECT NAME_TEACHER, Salary + Rise, 2 * Rise, 3 * Rise + 50
FROM TEACHER
WHERE Salary + Rise BETWEEN 2 * Rise AND 3 * Rise + 50;
```

Шаблонға сәйкестікті тексеру

Жазылуы:

өріс_атауы [NOT] LIKE шаблон [ESCAPE символ_пропуска]

Тек жолдық мәндерде ғана қолданылады.

Шаблонды қолдану

% — кез-келген символдар тізбегі;

_ — кез-келген бір символ.

LIKE операторы регистрден тәуелді, сол себепті UPPER() және LOWER() қолданған дұрыс.

Сұраныс 23. Фамилиясы 'М' әрпінен басталатын оқытушыларды табу.

```
SELECT NAME_TEACHER
FROM TEACHER
WHERE UPPER(NAME_TEACHER) LIKE 'M%';
```

Сұраныс 24. Фамилиясының 1-ші әрпі 'М', 4-ші әрпі – 'ы' болатын оқытушыны табу.

```
SELECT NAME_TEACHER
FROM TEACHER
WHERE NAME_TEACHER LIKE 'M__ ы%';
```

Сұраныс 25. ‘анализ’ сөзі кездесетін кафедра атауларын шығару.

```
SELECT Name_Kafedru  
FROM KAFEDRA  
WHERE LOWER(Name_Kafedru) LIKE '%анализ%';
```

Сұраныс 26. Фамилиясы мен қызметі атауында ‘о’ әрпі 5 рет кездесетіндерді табу.

```
SELECT NAME_TEACHER, DOLGNOST  
FROM TEACHER  
WHERE  
LOWER(NAME_TEACHER +DOLGNOST) LIKE '%о%о%о%о%о%о%';
```

Нөлдік емес мәнге теңдігін тексеру

Жазылуы:

өріс_атауы IS [NOT] NULL

Сұраныс 27. Телефон номері немесе идентификациялық коды берілмеген оқытушылардың фамилиясын шығару.

```
SELECT NAME_TEACHER, INDEF_KOD, TEL_TEACHER  
FROM TEACHER  
WHERE INDEF_KOD IS NULL OR TEL_TEACHER IS NULL;
```

№5 практикалық жұмыс тапсырмасы

Өз варианттарыңыз бойынша **Transact-SQL** тілінде **15**сұраныс құрып шығарыңыз:

- Салыстыруды қолданып 3 қарапайым сұраныс құру;
- AND, OR және NOT логикалық операторларын қолданып 3 сұраныс құру;
- Логикалық опертаорларды араластырып қолдануға 1 сұраныс құру;
- Өрнектерді қолдануға 1 сұраныс;
- Жиынға тиістілікке 2 сұраныс;
- Диапазонға тиістілікке 2 сұраныс;
- Шаблонға сәйкестікке 2 сұраныс;
- Нөлдік мәнге тексеруге 1 сұраныс.

Әрбір сұранысқа жазбаша түсінік беріп, **SQL Server Management Studio** МҚБЖ-нде орындап, скриншоттарымен қосып салынған **Прак5_СтудентФамилиясы.docx** файлына сақтап, **univer** жүйесіне жүктеу.