



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО”

Факультет прикладної математики
Кафедра програмного забезпечення комп’ютерних систем

Лабораторна робота № 2

з дисципліни “Бази даних”

тема “Створення додатку бази даних, орієнтованого на взаємодію з СУБД
PostgreSQL”

Виконав(ла)

студент(ка) 2 курсу

групи КП-91

Лопаткіна Катерина Олегівна

(прізвище, ім'я, по батькові)

варіант Школа

(предмети, вчителі, оцінки, учні,
група)

Перевірів

“ ____ ” “ _____ ” 20__ р.

викладач

Петрашенко Андрій Васильович

(прізвище, ім'я, по батькові)

Київ 2020

Мета

Метою роботи є здобуття вмінь програмування прикладних додатків баз даних PostgreSQL.

Загальні вимоги до завдання

Загальне завдання роботи полягає у наступному:

1. Реалізувати функції внесення, редагування та видалення даних у таблицях бази даних, створених у лабораторній роботі №1, засобами консольного інтерфейсу.
2. Передбачити автоматичне пакетне генерування «рандомізованих» даних у базі.
3. Забезпечити реалізацію пошуку за декількома атрибутами з двох та більше сутностей одночасно: для числових атрибутів – у рамках діапазону, для рядкових – як шаблон функції LIKE оператора SELECT SQL, для логічного типу – значення True/False, для дат – у рамках діапазону дат.
4. Програмний код виконати згідно шаблону MVC (модель-подання-контролер).

Репозиторій :

<https://github.com/katerynaLopatkina/Database/tree/main/lab2>

Пункт 1

Student menu:

1. Show all students;
2. Display student by id
3. Add student;
4. Update student
5. Delete student;
6. Generate students;
7. Add subject to student;
0. Exit

2

Enter student id : 5

There is no such student.

Student menu:

1. Show all students;
2. Display student by id
3. Add student;
4. Update student
5. Delete student;
6. Generate students;
7. Add subject to student;
0. Exit

3

Enter student first name : Katya

Enter student last name : Lopatkina

Enter student sex : Female

Enter group id : 8

Enter year : 2002

Enter month : 6

Enter day : 15

Your provided foreign ids are incorrect

Student menu:

Student menu:

1. Show all students;
2. Display student by id
3. Add student;
4. Update student
5. Delete student;
6. Generate students;
7. Add subject to student;
0. Exit

4

Enter student id : 2

We don't have any student in our student list. Please insert it first!

```
Student menu:
  1. Show all students;
  2. Display student by id
  3. Add student;
  4. Update student
  5. Delete student;
  6. Generate students;
  7. Add subject to student;
  0. Exit
```

3

```
Enter student first name : Katya
Enter student last name : Lopatkina
Enter student sex : Female
Enter group id : 2
Enter year : 2
Enter month : 1221
\Enter day : 454
```

Your provided value is not a date

Enter year :

```
Student menu:
  1. Show all students;
  2. Display student by id
  3. Add student;
  4. Update student
  5. Delete student;
  6. Generate students;
  7. Add subject to student;
  0. Exit
```

45

There is no such menu option

```
Student menu:
  1. Show all students;
  2. Display student by id
  3. Add student;
  4. Update student
  5. Delete student;
  6. Generate students;
  7. Add subject to student;
  0. Exit
```

3

Enter student first name :

Your provided value is empty or wrong - first name

```
2. Display teacher by id
3. Add teacher;
4. Update teacher
5. Delete teacher;
6. Generate teachers;
7. Add subject to teacher;
0. Exit
```

3

```
Enter teacher first name : sdfsd
Enter teacher last name : sdfsd
Enter teacher phone number : sdfsd
```

Your provided value is empty or wrong - phone number

```
Student menu:
  1. Show all students;
  2. Display student by id
  3. Add student;
  4. Update student
  5. Delete student;
  6. Generate students;
  7. Add subject to student;
  0. Exit
```

1







--- STUDENTS LIST ---

There is no items in this table - student

Пункт 2

```
1 SELECT * FROM public.student
2 ORDER BY id ASC
```



Data Output

	 id [PK] integer	 first_name character varying (100)	 last_name character varying (100)	 sex character varying (100)	 id_group integer	 date_of_birth date
1	81	Dima	Lahno	Female	1185	2003-03-06
2	82	Katya	Lopatkina	Female	547	2007-02-14
3	83	Dima	Lahno	Male	848	2007-04-22
4	84	Dima	Kulish	Male	717	2011-10-31
5	85	Olya	Shupik	Male	956	2002-05-19
6	86	Olya	Kulish	Female	362	2015-07-28
7	87	Katya	Kulish	Female	525	2002-11-05
8	88	Andrey	Lahno	Female	937	2001-02-07
9	89	Dima	Kulish	Female	759	2011-04-04
10	90	Dima	Lahno	Male	802	2005-05-31
11	91	Dima	Kulish	Female	724	2011-05-12
12	92	Andrey	Lahno	Male	705	2018-01-15
13	93	Katya	Lopatkina	Male	777	2004-11-10
14	94	Andrey	Lopatkina	Male	1129	2004-02-05
15	95	Andrey	Lahno	Male	442	2006-08-20

Query History **Data Output** Explain Messages Notifications

```
1 SELECT * FROM public."group"
2 ORDER BY id ASC
```

Data Output

	 id [PK] integer	 name character varying (100)
993	1182	WA-70
994	1183	XD-98
995	1184	SV-31
996	1185	TD-66
997	1186	SY-75
998	1187	WN-76
999	1188	OR-39
1000	1189	HA-30

Query History **Data Output** Explain Messages

```

1 SELECT * FROM public.subject
2 ORDER BY id ASC

```

Data Output

	id [PK] integer	name character varying (100)	time integer
991	1103	Philosophy	83
992	1104	Web programing	98
993	1105	Philosophy	69
994	1106	Java	91
995	1107	Java	28
996	1108	Ukrainian language	33
997	1109	English	48
998	1110	Ukrainian language	11
999	1111	Database	62
1000	1112	Oop	23

Query History **Data Output** Explain Messages < >

```

1 SELECT * FROM public.teacher_subject
2 ORDER BY id ASC

```

Data Output

	id [PK] integer	id_teacher integer	id_subject integer
996	1065	282	659
997	1066	954	749
998	1067	615	180
999	1068	936	115

Query History **Data Output** Explain Messag < >

```

1 SELECT * FROM public.teacher
2 ORDER BY id ASC

```

Data Output

	id [PK] integer	first_name character varying (100)	last_name character varying (100)	phone_number character varying (13)
100991	101103	Pavel	Malyutin	380294912408
100992	101104	Dmytro	Nazarenko	380614611862
100993	101105	Dmytro	Legeza	380522757686
100994	101106	Oleg	Melnik	380372146664
100995	101107	Pavel	Legeza	380304973559
100996	101108	Dmytro	Nazarenko	380632047148
100997	101109	Pavel	Legeza	380467696861
100998	101110	Dmytro	Malyutin	380418622625
100999	101111	Oleg	Nazarenko	380711311371
101000	101112	Pavel	Malyutin	380277225854

Query History **Data Output** Explain Messages Notifications < >

```

1 SELECT * FROM public.student_subject
2 ORDER BY id ASC

```

Data Output

	id [PK] integer	id_student integer	id_subject integer
997	1273	333	619
998	1274	267	255
999	1275	988	899
1000	1276	643	656

Query History **Data Output** Explain Messages Nc

```

1 SELECT * FROM public.subject_group
2 ORDER BY id ASC

```

Data Output

	id [PK] integer	id_subject integer	id_group integer
997	1145	673	654
998	1146	352	1157
999	1147	736	764
1000	1148	850	737

Query History **Data Output** Explain Messag <

Копії SQL запитів:

Генерація випадкових груп

```
sql = "INSERT into \"group\" (name) " \
      "SELECT chr(trunc(65 + random() * 25)::int) " \
      "|| chr(trunc(65 + random() * 25)::int) " \
      "|| '-' " \
      "|| (random() * 100)::int as name " \
      "from GENERATE_SERIES(1, {}) as seq " \
      "RETURNING *".format(count)
```

Генерація випадкових предметів

```
sql = "INSERT into subject (name, time) " \
      "SELECT " \
      "(case (random() * 8)::int " \
      "when 0 then 'Oop' " \
      "when 1 then 'Database' " \
      "when 2 then 'Physics' " \
      "when 3 then 'Philosophy' " \
      "when 4 then 'Web programing' " \
      "when 5 then 'Java' " \
      "when 6 then 'Math' " \
      "when 7 then 'English' " \
      "when 8 then 'Ukrainian language' " \
      "end) as name, " \
      "(random() * 100) as time " \
      "from GENERATE_SERIES(1, {}) as seq " \
      "RETURNING *".format(count)
```

Генерація випадкових студентів

```
sql = "INSERT into student (first_name, last_name, sex, id_group, date_of_birth) " \
      "SELECT " \
      "(case (random() * 3)::int " \
      "when 0 then 'Katya' " \
      "when 1 then 'Dima' " \
      "when 2 then 'Andrey' " \
      "when 3 then 'Olya' " \
      "end) as first_name, " \
      "(case (random() * 3)::int " \
      "when 0 then 'Lopatkina' " \
      "when 1 then 'Lahno' " \
      "when 2 then 'Kulish' " \
      "when 3 then 'Shupik' " \
      "end) as last_name, " \
      "(case (random() * 1)::int " \
      "when 0 then 'Female' " \
      "when 1 then 'Male' end) " \
      "as sex, " \
      "get_group_id() as id_group, " \
      "((NOW() + (random() * (NOW() - INTERVAL '20 years' - NOW())))::date) as date_of_birth " \
      "from GENERATE_SERIES(1, {}) as seq " \
      "RETURNING *".format(count)
```

Генерація випадкових вчителів

```
sql = "INSERT into teacher (first_name, last_name, phone_number) " \
      "SELECT " \
      "(case (random() * 3)::int " \
      "when 0 then 'Oleg' " \
      "when 1 then 'Dmytro' " \
      "when 2 then 'Pavel' " \
      "when 3 then 'Olya' " \
      "end) as first_name, " \
      "(case (random() * 3)::int " \
      "when 0 then 'Melnik' " \
      "when 1 then 'Malyutin' " \
      "when 2 then 'Nazarenko' " \
      "when 3 then 'Legeza' " \
      "end) as last_name, " \
      "('380' " \
      "|| (RANDOM() * 9)::INT " \
      "|| (RANDOM() * 9)::INT " \
      "|| (RANDOM() * 9)::INT " \
      "|| (RANDOM() * 9)::INT " \
      "|| (RANDOM() * 9)::INT " \
      "|| (RANDOM() * 9)::INT " \
      "|| (RANDOM() * 9)::INT " \
      "|| (RANDOM() * 9)::INT " \
      "|| (RANDOM() * 9)::INT) as phone_number " \
      "from GENERATE_SERIES(1, {}) as seq " \
      "RETURNING *".format(count)
```


Генерація випадкових зв'язків вчителів і предметів

```
sql = "insert into teacher_subject (id_teacher, id_subject) " \
      "select get_teacher_id(), get_subject_id() " \
      "FROM generate_series(1, '{}') ON CONFLICT DO NOTHING RETURNING *".format(count)
```

Генерація випадкових зв'язків учнів і предметів

```
sql = "insert into student_subject (id_student, id_subject) " \
      "select get_student_id(), get_subject_id() " \
      "FROM generate_series(1, '{}') ON CONFLICT DO NOTHING RETURNING *".format(count)
```


Генерація випадкових зв'язків учнів і предметів

```
sql = "insert into subject_group (id_subject, id_group) " \
      "select get_subject_id(), get_group_id() " \
      "FROM generate_series(1, '{}') ON CONFLICT DO NOTHING RETURNING *".format(count)
```

Для генерації зв'язків між таблицями потрібно було отримувати їх ідентифікатори, і для цього було створено додаткові функції в базі даних.

 **get_group_id()**

General	Definition	Code	Options	Parameters	Security
<pre>1 DECLARE 2 group_id int; 3 BEGIN 4 SELECT "group".id INTO group_id FROM "group" 5 ORDER BY RANDOM() LIMIT 1; 6 RETURN group_id; 7 END;</pre>					

 **get_student_id()**

General	Definition	Code	Options	Parameters	Security
<pre>1 DECLARE 2 student_id int; 3 BEGIN 4 SELECT student.id INTO student_id FROM student 5 ORDER BY RANDOM() LIMIT 1; 6 RETURN student_id; 7 END;</pre>					

get_subject_id()

General Definition Code Options Parameters Security S

```
1 DECLARE
2   subject_id int;
3 BEGIN
4   SELECT subject.id INTO subject_id FROM subject
5   ORDER BY RANDOM() LIMIT 1;
6   RETURN subject_id;
7 END;
```

get_teacher_id()

General Definition Code Options Parameters Security :

```
1 DECLARE
2   teacher_id int;
3 BEGIN
4   SELECT teacher.id INTO teacher_id FROM teacher
5   ORDER BY RANDOM() LIMIT 1;
6   RETURN teacher_id;
7 END;
```

Teacher menu:

1. Show all teachers;
2. Display teacher by id
3. Add teacher;
4. Update teacher
5. Delete teacher;
6. Generate teachers;
7. Add subject to teacher;
0. Exit

Enter teacher Amount of generated data : 100000

```
* 99994. id - 101106, first name - Oleg, last name - Melnik, phone number - 380372146664
* 99995. id - 101107, first name - Pavel, last name - Legeza, phone number - 380304973559
* 99996. id - 101108, first name - Dmytro, last name - Nazarenko, phone number - 380632047148
* 99997. id - 101109, first name - Pavel, last name - Legeza, phone number - 380467696861
* 99998. id - 101110, first name - Dmytro, last name - Malyutin, phone number - 380418622625
* 99999. id - 101111, first name - Oleg, last name - Nazarenko, phone number - 380711311371
* 100000. id - 101112, first name - Pavel, last name - Malyutin, phone number - 380277225854
```

Пункт 3

```
Enter student first name : Katya
Enter student sex : Female
Enter start date
Enter year : 2010
Enter month : 5
Enter day : 5
Enter end date
Enter year : 2015
Enter month : 5
Enter day : 5
--- STUDENTS LIST ---
* 1. id - 139, first name - Katya, last name - Kulish, sex - Female, date of birth - 2014-09-24, group name - RS-90, group id - 1132
* 2. id - 341, first name - Katya, last name - Kulish, sex - Female, date of birth - 2014-11-02, group name - SP-72, group id - 1168
* 3. id - 378, first name - Katya, last name - Shupik, sex - Female, date of birth - 2013-10-19, group name - RI-18, group id - 1025
* 4. id - 381, first name - Katya, last name - Lahno, sex - Female, date of birth - 2010-05-15, group name - NV-98, group id - 797
* 5. id - 385, first name - Katya, last name - Shupik, sex - Female, date of birth - 2013-08-17, group name - HV-33, group id - 1066
* 6. id - 471, first name - Katya, last name - Kulish, sex - Female, date of birth - 2014-03-03, group name - XR-86, group id - 249
* 7. id - 518, first name - Katya, last name - Kulish, sex - Female, date of birth - 2014-09-12, group name - AD-8, group id - 854
* 8. id - 673, first name - Katya, last name - Kulish, sex - Female, date of birth - 2012-11-14, group name - PY-84, group id - 351
* 9. id - 689, first name - Katya, last name - Shupik, sex - Female, date of birth - 2013-01-25, group name - IG-15, group id - 330
* 10. id - 722, first name - Katya, last name - Kulish, sex - Female, date of birth - 2010-08-13, group name - YB-68, group id - 352
* 11. id - 780, first name - Katya, last name - Lopatkina, sex - Female, date of birth - 2011-05-16, group name - LA-63, group id - 722
* 12. id - 798, first name - Katya, last name - Shupik, sex - Female, date of birth - 2012-09-01, group name - MB-69, group id - 678
* 13. id - 804, first name - Katya, last name - Lahno, sex - Female, date of birth - 2014-10-29, group name - KH-84, group id - 1019
* 14. id - 866, first name - Katya, last name - Kulish, sex - Female, date of birth - 2012-07-29, group name - JG-18, group id - 1020
* 15. id - 884, first name - Katya, last name - Shupik, sex - Female, date of birth - 2014-12-01, group name - FD-83, group id - 350
* 16. id - 904, first name - Katya, last name - Lopatkina, sex - Female, date of birth - 2013-09-09, group name - UF-54, group id - 883
* 17. id - 930, first name - Katya, last name - Kulish, sex - Female, date of birth - 2015-05-01, group name - TX-56, group id - 1013
* 18. id - 991, first name - Katya, last name - Lopatkina, sex - Female, date of birth - 2015-02-28, group name - EV-1, group id - 1177
Search duration - 0.0020051002502441406
```

```
sql = "SELECT s.id, first_name, last_name, sex, date_of_birth, gr.name as group_name, gr.id as group_id " \
      "from student as s " \
      "inner join \"group\" as gr ON gr.id = s.id_group " \
      "where first_name like '%{}%' " \
      "and sex like '{}'" \
      "and date_of_birth >= '{}'" \
      "and date_of_birth <= '{}'" \
      "order by s.id ASC".format(scrub(first_name), scrub(sex), star_date, end_date)
```

```
Search menu:
1. Search students
2. Search teachers;
3. Search students by teacher and group;
0. Exit
2
Enter teacher first name : Dmytro
Enter teacher Subject name : Oop
Enter subject max credits : 50
Enter subject min credits : 23
--- TEACHERS LIST ---
* 1. id - 319, first name - Dmytro, last name - Melnik, phone number - 380836156357, subject name - Oop, subject credits - 41, subject id - 305
* 2. id - 1027, first name - Dmytro, last name - Legeza, phone number - 380716065094, subject name - Oop, subject credits - 45, subject id - 1065
* 3. id - 237, first name - Dmytro, last name - Malyutin, phone number - 380564282776, subject name - Oop, subject credits - 41, subject id - 1073
* 4. id - 454, first name - Dmytro, last name - Malyutin, phone number - 380944167220, subject name - Oop, subject credits - 41, subject id - 1073
Search duration - 0.003957986831665039
```

```
sql = "SELECT teacher.id, first_name, last_name, phone_number, " \
      "s.name as subject_name, s.time as credits, s.id as id_subject " \
      "from teacher " \
      "inner join teacher_subject as ts ON teacher.id = ts.id_teacher " \
      "inner join subject as s ON s.id = ts.id_subject " \
      "where first_name like '%{}%' " \
      "and s.name like '%{}%' " \
      "and s.time >= '{}'" \
      "and s.time <= '{}'" \
      "order by s.id ASC" \
      .format(scrub(first_name), scrub(subject_name), min_credits, max_credits)
```

```

Enter teacher first name : Olya
Enter teacher Group name : YI-79
Enter start date
Enter year : 2002
Enter month : 5
Enter day : 5
Enter end date
Enter year : 2020
Enter month : 5
Enter day : 5
--- STUDENTS LIST WITH TEACHER, GROUP and SUBJECT ---
* 1. id - 264, student first name - Andrey, student last name - Lahno, student date of birth - 2010-11-21group name - YI-79, group id - 996, s
* 2. id - 407, student first name - Olya, student last name - Shupik, student date of birth - 2005-10-16group name - YI-79, group id - 996, su
Search duration - 0.0030307769775390625

```

```

sql = "SELECT DISTINCT s.id as id_student, s.first_name as first_name_s, s.last_name as last_name_s," \
      " s.date_of_birth, gr.name as group_name," \
      " gr.id as id_group, sb.id as id_subject, sb.name as sb_name, teach.id as id_teacher," \
      " teach.first_name as first_name_t," \
      " teach.last_name as last_name_t " \
      "from " \
      "student as s " \
      "inner join \"group\" as gr on s.id_group = gr.id " \
      "inner join subject_group as sg on sg.id_group = gr.id " \
      "inner join subject as sb on sg.id_subject = sb.id " \
      "inner join teacher_subject as ts on ts.id_subject = sb.id " \
      "inner join teacher as teach on teach.id = ts.id_teacher " \
      "where teach.first_name like '%{}%' " \
      "and gr.name like '{}{}' " \
      "and s.date_of_birth >= '{}{}' " \
      "and s.date_of_birth <= '{}{}' " \
      "order by s.id asc" \
      .format(scrub(first_name), group_name, start_date, end_date)

```

Пункт 4

main Database / lab2 /

katerynaLopatkina add lab2	
..	
.idea	add lab2
models	add lab2
.DS_Store	add lab2
README.md	add lab2
controller.py	add lab2
main.py	add lab2
model.py	add lab2
mvc_exception.py	add lab2
postgres_backend.py	add lab2
view.py	add lab2

main Database / lab2 / model.py / <> Jump to

katerynaLopatkina add lab2

1 contributor

226 lines (197 sloc) | 8.68 KB

```
1 import postgres_backend
2
3 from models.student import Student
4 from models.subject import Subject
5 from models.grade import Grade
6 from models.group import Group
7 from models.teacher import Teacher
8 from models.teacher_subject import TeacherSubject
9 from models.subject_group import SubjectGroup
10 from models.student_subject import StudentSubject
11
12 GROUP_TABLE = "\"group\""
13 STUDENT_TABLE = "student"
14 TEACHER_TABLE = "teacher"
15 SUBJECT_TABLE = "subject"
16 GRADE_TABLE = "grade"
17 SUBJECT_GROUP_TABLE = "subject_group"
18 TEACHER_SUBJECT_TABLE = "teacher_subject"
19 STUDENT_SUBJECT_TABLE = "student_subject"
20
21
22 def get_value_if_exists(res, key):
23     if key in res:
24         return res[key]
25     else:
26         return None
27
28
29 class Model(object):
30
31     def __init__(self):
32         self._cursor = postgres_backend.connect_db()
33
34     @property
35     def connection(self):
36         return self._cursor
37
```

main Database / lab2 / view.py / <> Jump to

katerynaLopatkina add lab2

1 contributor

352 lines (309 sloc) | 15.1 KB

```
1 from model import GRADE_TABLE
2 from model import GROUP_TABLE
3 from model import SUBJECT_TABLE
4 from model import STUDENT_TABLE
5 from model import TEACHER_TABLE
6 from model import TEACHER_SUBJECT_TABLE
7 from model import STUDENT_SUBJECT_TABLE
8 from model import SUBJECT_GROUP_TABLE
9
10
11 class View(object):
12
13     @staticmethod
14     def show_main_menu():
15         return input("Main menu:\n"
16                     "\t1. Student\n"
17                     "\t2. Teacher;\n"
18                     "\t3. Subject;\n"
19                     "\t4. Group;\n"
20                     "\t5. Subject Group;\n"
21                     "\t6. Teacher Subject;\n"
22                     "\t7. Student Subject;\n"
23                     "\t8. Search menu;\n"
24                     "\t0. Exit\n")
25
26     @staticmethod
27     def show_search_menu():
28         return input("Search menu:\n"
29                     "\t1. Search students\n"
30                     "\t2. Search teachers;\n"
31                     "\t3. Search students by teacher and group;\n"
32                     "\t0. Exit\n")

```


main ▾

Database / lab2 / controller.py / <> Jump to ▾



katerynaLopatkina add lab2

1 contributor

670 lines (570 sloc) | 26.4 KB

```
1  import os
2  import datetime
3  import time
4
5  from model import GROUP_TABLE
6  from model import GRADE_TABLE
7  from model import STUDENT_TABLE
8  from model import SUBJECT_TABLE
9  from model import TEACHER_TABLE
10 from model import TEACHER_SUBJECT_TABLE
11 from model import STUDENT_SUBJECT_TABLE
12 from model import SUBJECT_GROUP_TABLE
13 from model import Model
14 from view import View
15
16
17 class Controller(object):
18     MAIN_M = -1
19     # GRADE_M = 5
20     STUDENT_M = 1
21     TEACHER_M = 2
22     SUBJECT_M = 3
23     GROUP_M = 4
24     SUBJECT_GROUP_M = 5
25     TEACHER_SUBJECT_M = 6
26     STUDENT_SUBJECT_M = 7
27     SEARCH_MENU = 8
28
29     def __init__(self):
30         self.view = View()
31         self.model = Model()
32
33     def start(self):
34         current_menu = self.MAIN_M
35
36         while current_menu == self.MAIN_M:
37             menu_option = self.view.show_main_menu()
38             if menu_option.isdigit():
39
40                 while int(menu_option) == self.GROUP_M:
41                     group_option = self.view.show_group_menu()
42                     if group_option.isdigit():
43                         current_menu = self.handle_common_inner_menu(int(group_option), GROUP_TABLE, self.GROUP_M)
```