



Базы данных и экспертные системы

НИЯУ МИФИ, Кафедра финансового мониторинга. Лабораторный практикум. В.Ю. Радыгин

Лабораторная работа 7

Лабораторная работа 7 рассчитана на работу в классе в течение 90 минут. Результатом работы является SQL-файл, содержащий решение всех задач на языке SQL или PL/SQL. Номера задач указываются в виде комментария.

Решения необходимо отправить преподавателю в виде файла, прикрепленного к сообщению на home.mephi.ru. В самом сообщении должен быть указан номер варианта. Решения, отправленные позже времени окончания занятия проверяться не будут. В таком случае оценка автоматически считается равной 0, а лабораторная работа – не сданной.

Лабораторная работа оценивается максимум в 15 баллов. Минимальная положительная оценка – 9 баллов.

Вариант 1

Задание 1

- Создайте две таблицы: `operations` и `operations_log`. Первая таблица должна содержать поля идентификатор (`id`), номер счёта (`account_number`), название операции (`operation_name`), сумма операции (`operation_sum`). Поле `id` – это первичный ключ. Поле `operation_sum` – это действительное число. Вторая таблица должна содержать поля: идентификатор операции (`operation_id`), номер счёта (`account_number`), дата операции (`operation_date`), тип операции (`operation_type`). Поле `operation_date` должно позволять сохранять дату и время. Поле `operation_type` может содержать только два значения '-', '+'. Заполните таблицы `operations` и `operations_log` несколькими записями (не менее 5 на таблицу).

Задание 2

- Напишите процедуру или функцию `statement_of_acount`, которая на основе трёх параметров: даты начала периода, даты окончания периода и номера счёта печатала бы на экран данные (тип операции, сумма, дата операции) трёх самых крупных операций с положительной суммой, трёх самых крупных операций с отрицательной суммой и данные об общем числе операций и среднем значении суммы.

Задание 3

- Напишите процедуру или функцию `account_operation`, которая на основе трёх параметров: номера счёта, идентификатора и суммы вносит изменения в таблицу `operations`. В таблицу `operations` должна добавляться запись, где поля идентификатор, номер счёта и сумма по операции должны заполняться переданными параметрами. Поле название операции заполняется фразой «внесение денег на счёт». Если сумма отрицательная или ноль, ничего вноситься не должно.

Вариант 2

Задание 1

- Создайте две таблицы: `people` и `people_log`. Первая таблица должна содержать поля идентификатор (`id`), имя (`first_name`), фамилия (`last_name`), дата рождения (`birthday`), текущее состояние счёта (`amount`). Поле `id` – это первичный ключ. Поле `amount` – это действительное число. Вторая таблица должна содержать поля: идентификатор человека (`person_id`), дата операции (`operation_date`), сумма (`sum`). Поле `person_id` должно быть ограничено, как внешний ключ, ссылающийся на поле `id` таблицы `people`. Поле `operation_date` должно позволить сохранять дату и время. Поле `sum` – это действительное число. Заполните таблицы `people` и `people_log` несколькими записями (не менее 5 на таблицу).

Задание 2

- Напишите процедуру или функцию `statement_of_acount`, которая на основе двух параметров: даты начала периода и даты окончания периода, печатала бы на экран данные по каждой операции из таблицы `people_log` за указанный период (`operation_date` и `sum`). В конце должна печататься информация об общем числе операций и суммах положительных и отрицательных операций.

Задание 3

- Напишите процедуру или функцию `account_operation`, которая на основе двух параметров: идентификатора человека и суммы вносит изменения в таблицы `people` и `people_log`. В таблице `people` у человека с указанным в качестве параметра идентификатором состояние счёта должно измениться на указанную в качестве параметра сумму. В таблицу `people_log` должна быть добавлена запись с указанным в качестве параметра идентификатором человека, текущей датой и временем и указанной в качестве параметра суммой.

Вариант 3

Задание 1

- Создайте две таблицы: `operations` и `operations_log`. Первая таблица должна содержать поля идентификатор (`id`), номер счёта (`account_number`), номер документа (`document_number`), название операции (`operation_name`), код операции (`operation_code`) сумма по операции (`operation_sum`). Поле `id` – это первичный ключ. Поле `operation_sum` – это действительное число. Вторая таблица должна содержать поля: идентификатор операции (`operation_id`), номер счёта (`account_number`) дата операции (`operation_date`), тип операции (`operation_type`). Поле `operation_date` должно позволять сохранять дату и время. Поле `operation_type` может содержать только два значения '-', '+'. Заполните таблицы `operations` и `operations_log` несколькими записями (не менее 5 на таблицу).

Задание 2

- Напишите процедуру или функцию `statement_of_acount`, которая на основе трёх параметров: даты начала периода, даты окончания периода и номера счёта печатала бы на экран данные по каждой операции из таблиц `operations` и `operations_log` за указанный период (тип операции, сумма, дата операции). В конце должна печататься информация об общем числе операций и две суммы: все положительные и все отрицательные операции.

Задание 3

- Напишите процедуру или функцию `account_operation`, которая на основе трёх параметров: номера счёта, идентификатора и суммы вносит изменения в таблицы `operations` и `operations_log`. В таблицу `operations` должна добавляться запись, где поля идентификатор, номер счёта и сумма по операции должны заполняться переданными параметрами. Поля номер документа, код операции и название операции заполняются следующим образом: если сумма операции > 0 , то это ('12345', '1', 'внесение денег на счёт') иначе это ('12346', '2', 'снятие денег со счёта').

Вариант 4

Задание 1

- Создайте две таблицы: `people` и `people_log`. Первая таблица должна содержать поля идентификатор (`id`), имя (`first_name`), фамилия (`last_name`), дата рождения (`birthday`), текущее состояние счёта (`amount`). Поле `id` – это первичный ключ. Поле `amount` – это действительное число. Вторая таблица должна содержать поля: идентификатор человека (`person_id`), дата операции (`operation_date`), сумма (`sum`). Поле `person_id` должно быть ограничено, как внешний ключ, ссылающийся на поле `id` таблицы `people`. Поле `operation_date` должно позволить сохранять дату и время. Поле `sum` – это действительное число. Заполните таблицы `people` и `people_log` несколькими записями (не менее 5 на таблицу).

Задание 2

- Напишите процедуру или функцию `statement_of_acount`, которая на основе двух параметров: даты начала периода и даты окончания периода печатала бы на экран данные (`operation_date` и `sum`) трёх самых крупных операций с положительной суммой, трёх самых крупных операций с отрицательной суммой и данные об общем числе операций и среднем значении суммы.

Задание 3

- Напишите процедуру или функцию `account_operation`, которая на основе двух параметров: идентификатора человека и суммы вносит изменения в таблицы `people` и `people_log`. В таблице `people` у человека с указанным в качестве параметра идентификатором состояние счёта должно измениться на указанную в качестве параметра сумму. В таблицу `people_log` должна быть добавлена запись с указанным в качестве параметра идентификатором человека, текущей датой и временем и указанной в качестве параметра суммой.

