МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРЬКОВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКІ

Звіт

З лабораторної роботи № 1

з дисципліни: «Проектування високонавантажених систем зберігання даних»

за темою «Проектування високонавантажених баз даних на платформі СУБД MYSQL з використанням таблиць типу MyISAM & InnoDB»

Виконали: Перевірив:

ст. гр. ІТКНу-17-1 Коваленко А. І.

Гочаренко В.О.

2018

* 1. Предметная область: ИС «Вступительная кампания» для университета.
  2. Университет - высшее учебное заведение, где готовятся специалисты по фундаментальным и многим прикладным наукам. Как правило, осуществляет и научно-исследовательскую работу. Многие современные университеты действуют как учебно-научно-практические комплексы. Университеты объединяют в своём составе несколько факультетов, на которых представлена совокупность различных дисциплин, составляющих основы научного знания. **Перечень нормативных документов** регламентирующих деятельность предприятия:
     1. Внутреннее положение университета.
     2. Правила приема в университет.
     3. Законы Украины в сфере образования.
  3. Состав ПО:
     1. Клиент – reactjs + scss
     2. Сервер – nodejs + разные фрейморвки для реализации рестфул приложения.
     3. БД – mysql на движке ...
  4. Основным процессом для автоматизации является функция «Формароивания рейтиногового списка». Т.е. когда конкурс подходит к концу то адмниистратор (или же по schedule-ру) будте происходит закрытие конкурса и формирование списков поступивших в университет.
  5. С точки зрения пользователя основной процесс для атовтмазации – это подача заявки для участие в конкурсе. Т.е. по нажатию на кнопку «Подать заявку» подсчитуются балы абитуриента осчитуется его средний бал и он добавляется в рейтинговый список.
     + 1. Admin – формирование рейтингово списка для определенной специальности, удаление, добалвение редактирование пользователей, специальностей, просмотр рейтинговых списков (непосредственно самого конкурса) по специальности.
       2. User – регистрация, авотризация, заполенение необходимой информации о себе (балы ЗНО, номер сертификата, город, нуждается ли в общежитии ...), зарегистрироваться для участия в конкурсе.
       3. Anonymous – просмотр рейтинговых списков по специальностям.

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Пользователь** | **Статус пользователя** | **Доступ к данным (наименования таблиц)** |
| **1** | **Незарегистрированный пользователь** | **Нет** | **Имеет доступ по чтению к таблицам: faculty, requestStatment (enrollee and other…)** |
| **2** | **Абитурент** | **Enrollee** | **Имеет доступ по записи и чтению к таблице : enrollee, request, enrolleeRequest, requestStatement.**  **Имеет доступ по чтению к таблицам: faculty, subject, facultySubject, statement, requestStatement.** |
| **3** | **Администратор системы** | **Admin** | **Имеет доступ по записи и чтению для всех таблиц БД** |
| **4** | **Администратор факультета** | **FacultyAdmin** | **Имеет доступ по записи и чтению для таблиц: faculty, facultySubject, statement.** |
| **5** | **Администратор абитуриентов** | **enrolleeAdmin** | **Имеет доступ по записи и чтению для таблиц: enrollee.** |
| **6** | **Администратор предметов** | **subjectAdmin** | **Имеет доступ по чтению и записи для таблиц: subject, subjectType** |

* 1. Сортировка специальностей по название, количеству участников, кол-во бюджетных мест, общему кол-ву мест, сортировка абитуриентов по фио, по среднему балу, по количеству заявок. Искать у абитуриентов в поле «Дополнительная информация» их хобби. К примеру, абитуреинты будут вписывать в это поле свои хобби и увлечения (музыка, спорт, баскетбол, футбол...) а мы будем искать по этому полю
  2. Таблица с перечнем основных особенностей .

|  |  |  |
| --- | --- | --- |
| № | Элементы интерфейса | Бизнес-функции высоконагруженной ИС |
| 1 | Таблица с факультетами,  Кнопка «Сортировать по названию факультета», Кнопка «Сортировать по общему количеству мест», Кнопка «Сортировать по бюджетному количеству мест». | Используются для просмотра таблицы доступных факультетов, на которые можно подать заявки. Нажатие каждой из кнопок отсортирует таблицу с факультетами по одному атрибуту. |
| 2 | Таблица с факультетами, Кнопка «Подать заявку» | Используются для формирования заявки абитуриента для участия в «конкурсе». Абитуриент нажимает кнопку подать заявку, после чего формируется заявка на факультет, который был выбран, после чего рейтинговый список обновляется. |
| 3 | Текстовое поле «Имя», «Фамилия», выпадающий список «Город»…  Кнопка «Сохранить» | Используются для заполнения и сохранения информации про абитуриента. |
| 4 | На странице информации про факультет кнопка «Сформировать ведомость». | Используется для формирования ведомости администратором, после чего новые заявки перестанут приниматься, всем, кто принял участие в конкурсе будут высланы списки и информация о тех кто прошел конкурс, и о тех кто его не прошел. |
| 5 | Таблица с заявками факультета | Используется для просмотра всех, кто принимает участие в конкурсе. |
| 6 | Текстовое поле «Описание факультета»,  Кнопка «Поиск» | Используются администратором для поиска факультета по определенному описанию, которое было введено в поле «Описание факультета» |
| 7 | Текстовое поле «Хобби абитуриента», Кнопка «поиск» | Используются для полнотекстового поиска по полю enrollee.additional. |

|  |  |  |  |
| --- | --- | --- | --- |
| № | Описание | MyISAM | InnoDB |
| 1 | Поддержка транзакций | - | + |
| 2 | Поддержка связей по внешним ключам | - | + |
| 3 | Поддержка ссылочной целостности связей для инструкций update, delete, insert | - | + |
| 4 | Блокировка на уровне таблиц\записней\нет блокировки | Таблиц | Записей |
| 5 | Одновременные запросы к разным записям одной таблицы – медленнее\быстрее | Медленнее | Быстрее |
| 6 | При смешанной нагрузке на запросы SELECT, UPDATE, DELETE, INSERT – медленнее\быстрее | Медленнее | Быстрее |
| 7 | Однотипные операции Insert – медленнее\быстрее | Быстрее | Медленнее |
| 8 | Однотипные операции SELECCT – медленнее\быстрее | Быстрее | Медленнее |
| 9 | Запрос COUNT(\*) – медленнее\быстрее | Быстрее | Медленнее |
| 10 | Взаимная блокировка операция (DeadLock) – возможна\невозможна | Не возникают | Возможно |
| 11 | Поддержка полнотекстовокго поиска | + | - (+ MySQL 5.8.4) |
| 12 | Поддержка индексирования полей в запросах | + | + |
| 13 | Возможные типы индексов | PK –(?)  FK –  UNIQUE +  INDEX +  FULLTEXT + | PK +  FK +  UNIQUE+ INDEX+  FULLTEXT- (+ MySQL 5.6) |
| 14 | Возможность бинарного копирования таблиц | + | - |
| 15 | Размер таблицы БД | Меньше() | Больше() |
| 16 | Возможность восстановления в случае сбоя | - | + |
| 17 | СУБД создаст по умолчанию для таблиц | .MYD(MYData), .MYI(MyINdex) |  |

**MyISAM sql query.**

**CREATE SCHEMA IF NOT EXISTS `admissionsBoardInno` DEFAULT CHARACTER SET utf8 COLLATE utf8\_unicode\_ci;;**

**USE `admissionsBoardInno`;**

**-- create db**

**DROP DATABASE IF EXISTS `admissionsboard`;**

**CREATE DATABASE**

**IF NOT EXISTS `admissionsBoard`**

**DEFAULT CHARACTER SET utf8**

**COLLATE utf8\_unicode\_ci;**

**USE admissionsBoard;**

**CREATE TABLE IF NOT EXISTS `enrolee` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`firstName` VARCHAR(50) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NULL,**

**`lastName` VARCHAR(50) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NULL,**

**`login` VARCHAR(30) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NULL,**

**`password` VARCHAR(25) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NULL,**

**`email` VARCHAR(120) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NULL,**

**`additional` TEXT(500) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NULL,**

**`registrationDate` DATE NOT NULL,**

**`cityId` INT UNSIGNED NOT NULL,**

**PRIMARY KEY (`id`),**

**INDEX `enrolleeEmailIndex` (`email` ASC)**

**)**

**ENGINE = MyISAM**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**CREATE TABLE IF NOT EXISTS `subject` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`name` VARCHAR(50) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NOT NULL,**

**`description` TEXT(500) CHARACTER SET `utf8` COLLATE `utf8\_unicode\_ci` NOT NULL,**

**`popularity` FLOAT(4,2) NOT NULL,**

**`anyDate` DATE NULL,**

**`subjectTypeId` INT UNSIGNED NOT NULL,**

**PRIMARY KEY(`id`),**

**INDEX `subjectNameIndex` (`name` ASC)**

**)**

**ENGINE = MyISAM**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**CREATE TABLE IF NOT EXISTS `faculty` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`name` VARCHAR(50) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NULL,**

**`totalCount` SMALLINT UNSIGNED NOT NULL,**

**`budgetCount` SMALLINT UNSIGNED NOT NULL,**

**`creationDate` DATE NOT NULL,**

**`popularity` FLOAT(4,2) NULL,**

**`description` TEXT(500) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NOT NULL,**

**PRIMARY KEY (`id`),**

**INDEX `facultyNameIndex`(`name` ASC)**

**)**

**ENGINE = MyISAM**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**CREATE TABLE IF NOT EXISTS `statement` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`finishedDate` DATE NOT NULL,**

**`popularity` FLOAT(4,2) NULL,**

**`description` TEXT(500) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NOT NULL,**

**`someName` VARCHAR(50) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NOT NULL,**

**PRIMARY KEY (`id`),**

**INDEX `finishedDate` (`finishedDate` ASC)**

**)**

**ENGINE = MyISAM**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**CREATE TABLE IF NOT EXISTS `city` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`name` VARCHAR(255) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NOT NULL,**

**`area` FLOAT(10,2) NOT NULL,**

**`description` TEXT(500) NOT NULL,**

**`creationDate` DATE NOT NULL,**

**PRIMARY KEY `id`(`id`),**

**INDEX `cityName`(`name` ASC)**

**)**

**ENGINE = MyISAM**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**CREATE TABLE IF NOT EXISTS `subjecttype` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`type` VARCHAR(255) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NULL,**

**PRIMARY KEY (`id`)**

**)**

**ENGINE = MyISAM**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**CREATE TABLE IF NOT EXISTS `request` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`enrolleeId` INT UNSIGNED NOT NULL,**

**`facultyId` INT UNSIGNED NOT NULL,**

**`averageScore` FLOAT(4,2) NOT NULL,**

**PRIMARY KEY (`id`, `enrolleeId`, `facultyId`),**

**INDEX `requestEnrolleeIndex`(`enrolleeId` ASC),**

**INDEX `requestFacultyIndex`(`facultyId` ASC)**

**)**

**ENGINE = MyISAM**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**CREATE TABLE IF NOT EXISTS `statementRequest` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`requestId` INT UNSIGNED NOT NULL,**

**`statementId` INT UNSIGNED NOT NULL,**

**PRIMARY KEY (`id`, `requestId`, `statementId`),**

**INDEX `statementRequestIndex` (`requestId` ASC)**

**)**

**ENGINE = MyISAM**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**CREATE TABLE IF NOT EXISTS `enrolleeSubject` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`enrolleeId` INT UNSIGNED NOT NULL,**

**`subjectId` INT UNSIGNED NOT NULL,**

**`point` FLOAT(4,2) NOT NULL,**

**PRIMARY KEY (`id`, `enrolleeId`, `subjectId`),**

**INDEX `enrolleeSubjectIndex`(`enrolleeId` ASC)**

**)**

**ENGINE = MyISAM**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**CREATE TABLE IF NOT EXISTS `facultySubject` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`coefficient` FLOAT NOT NULL,**

**`facultyId` INT UNSIGNED NOT NULL,**

**`subjectId` INT UNSIGNED NOT NULL,**

**PRIMARY KEY (`id`),**

**INDEX `facultySubjectFacultyIdIndex` (`facultyId` ASC)**

**)**

**ENGINE = MyISAM**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**LOAD DATA INFILE 'enrolee.txt' INTO TABLE enrolee**

**FIELDS TERMINATED BY '::'**

**(firstName, lastName, login, password, email, additional, cityId, registrationDate);**

**LOAD DATA INFILE 'city.txt' INTO TABLE City**

**FIELDS TERMINATED BY '::'**

**(name, area, description, creationDate);**

**LOAD DATA INFILE 'faculty.txt' INTO TABLE faculty**

**FIELDS TERMINATED BY '::'**

**(name, totalCount, budgetCount, creationDate, popularity, description);**

**LOAD DATA INFILE 'statements.txt' INTO TABLE statement**

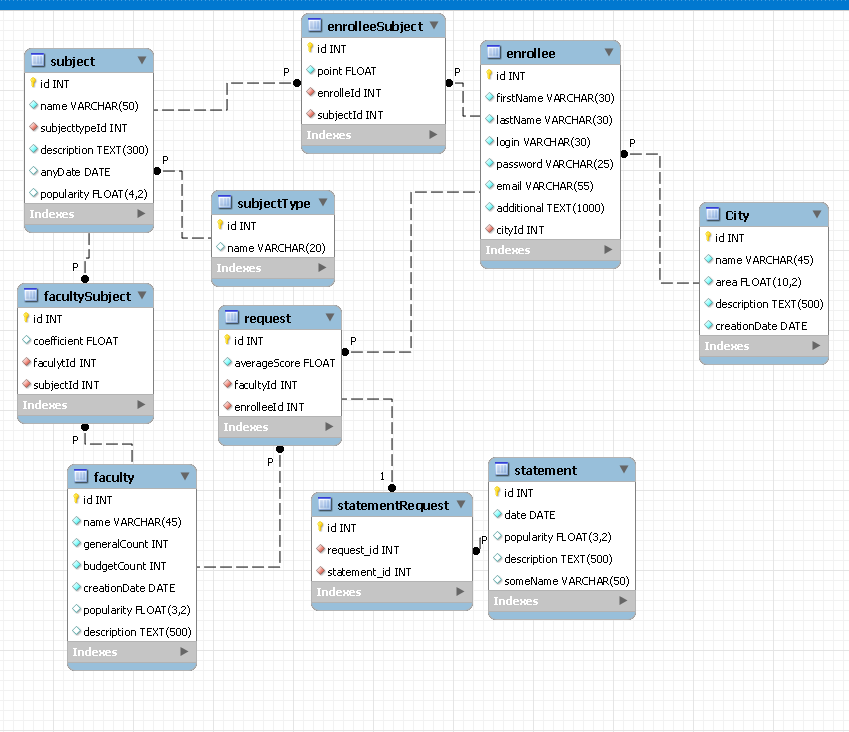
**FIELDS TERMINATED BY '::'**

**(someName, popularity, description, finishedDate);**

**LOAD DATA INFILE 'subjects.txt' INTO TABLE subject**

**FIELDS TERMINATED BY '::'**

**(name, subjecttypeid, description, anyDate, popularity);**



**InnoDB Query**

**CREATE SCHEMA IF NOT EXISTS `admissionsBoardInno` DEFAULT CHARACTER SET utf8 COLLATE utf8\_unicode\_ci;;**

**USE `admissionsBoardInno`;**

**CREATE TABLE IF NOT EXISTS `admissionsBoardInno`.`City` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`name` VARCHAR(45) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NOT NULL,**

**`area` FLOAT(10,2) UNSIGNED NOT NULL,**

**`description` TEXT(500) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NOT NULLk,**

**`creationDate` DATE NOT NULL,**

**PRIMARY KEY (`id`),**

**UNIQUE INDEX `id\_UNIQUE` (`id` ASC),**

**UNIQUE INDEX `name\_UNIQUE` (`name` ASC))**

**ENGINE = InnoDB**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**CREATE TABLE IF NOT EXISTS `admissionsBoardInno`.`enrollee` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`firstName` VARCHAR(30) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NOT NULL,**

**`lastName` VARCHAR(30) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NOT NULL,**

**`login` VARCHAR(30) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NOT NULL,**

**`password` VARCHAR(25) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NOT NULL,**

**`email` VARCHAR(55) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NOT NULL,**

**`additional` TEXT(1000) NOT NULL,**

**`cityId` INT UNSIGNED NOT NULL,**

**PRIMARY KEY (`id`),**

**UNIQUE INDEX `login\_UNIQUE` (`login` ASC),**

**UNIQUE INDEX `email\_UNIQUE` (`email` ASC),**

**UNIQUE INDEX `id\_UNIQUE` (`id` ASC),**

**INDEX `fk\_enrollee\_City1\_idx` (`cityId` ASC),**

**CONSTRAINT `fk\_enrollee\_City1`**

**FOREIGN KEY (`cityId`)**

**REFERENCES `admissionsBoardInno`.`City` (`id`)**

**ON DELETE NO ACTION**

**ON UPDATE CASCADE)**

**ENGINE = InnoDB**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**CREATE TABLE IF NOT EXISTS `admissionsBoardInno`.`subjectType` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`name` VARCHAR(20) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NULL,**

**PRIMARY KEY (`id`),**

**UNIQUE INDEX `id\_UNIQUE` (`id` ASC))**

**ENGINE = InnoDB**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**CREATE TABLE IF NOT EXISTS `admissionsBoardInno`.`subject` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`name` VARCHAR(50) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NOT NULL,**

**`subjecttypeId` INT UNSIGNED NOT NULL,**

**`description` TEXT(300) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NOT NULL,**

**`anyDate` DATE NULL,**

**`popularity` FLOAT(4,2) NULL,**

**PRIMARY KEY (`id`),**

**UNIQUE INDEX `id\_UNIQUE` (`id` ASC),**

**UNIQUE INDEX `name\_UNIQUE` (`name` ASC),**

**INDEX `fk\_subject\_subjectType1\_idx` (`subjecttypeId` ASC),**

**CONSTRAINT `fk\_subject\_subjectType1`**

**FOREIGN KEY (`subjecttypeId`)**

**REFERENCES `admissionsBoardInno`.`subjectType` (`id`)**

**ON DELETE NO ACTION**

**ON UPDATE CASCADE)**

**ENGINE = InnoDB**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**-- -----------------------------------------------------**

**-- Table `admissionsBoardInno`.`faculty`**

**-- -----------------------------------------------------**

**CREATE TABLE IF NOT EXISTS `admissionsBoardInno`.`faculty` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`name` VARCHAR(45) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NOT NULL,**

**`generalCount` INT UNSIGNED NOT NULL,**

**`budgetCount` INT UNSIGNED NOT NULL,**

**`creationDate` DATE NOT NULL,**

**`popularity` FLOAT(4,2) NULL,**

**`description` TEXT(500) CHARACTER SET 'utf8' COLLATE 'utf8\_unicode\_ci' NULL,**

**PRIMARY KEY (`id`),**

**UNIQUE INDEX `name\_UNIQUE` (`id` ASC),**

**UNIQUE INDEX `name\_UNIQUE` (`name` ASC))**

**ENGINE = InnoDB**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**-- Table `admissionsBoardInno`.`statement`**

**CREATE TABLE IF NOT EXISTS `admissionsBoardInno`.`statement` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`date` DATE NOT NULL,**

**`popularity` FLOAT(4,2) NULL,**

**`description` TEXT(500) NULL,**

**`someName` VARCHAR(50) NULL,**

**PRIMARY KEY (`id`),**

**UNIQUE INDEX `id\_UNIQUE` (`id` ASC))**

**ENGINE = InnoDB**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**CREATE TABLE IF NOT EXISTS `admissionsBoardInno`.`enrolleeSubject` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`point` FLOAT NOT NULL,**

**`enrolleId` INT UNSIGNED NOT NULL,**

**`subjectId` INT UNSIGNED NOT NULL,**

**PRIMARY KEY (`id`),**

**UNIQUE INDEX `id\_UNIQUE` (`id` ASC),**

**INDEX `fk\_enrolleeSubject\_enrollee\_idx` (`enrolleId` ASC),**

**INDEX `fk\_enrolleeSubject\_subject1\_idx` (`subjectId` ASC),**

**CONSTRAINT `fk\_enrolleeSubject\_enrollee`**

**FOREIGN KEY (`enrolleId`)**

**REFERENCES `admissionsBoardInno`.`enrollee` (`id`)**

**ON DELETE NO ACTION**

**ON UPDATE CASCADE,**

**CONSTRAINT `fk\_enrolleeSubject\_subject1`**

**FOREIGN KEY (`subjectId`)**

**REFERENCES `admissionsBoardInno`.`subject` (`id`)**

**ON DELETE NO ACTION**

**ON UPDATE CASCADE)**

**ENGINE = InnoDB**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**-- -----------------------------------------------------**

**-- Table `admissionsBoardInno`.`facultySubject`**

**-- -----------------------------------------------------**

**CREATE TABLE IF NOT EXISTS `admissionsBoardInno`.`facultySubject` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`coefficient` FLOAT NULL,**

**`faculytId` INT UNSIGNED NOT NULL,**

**`subjectId` INT UNSIGNED NOT NULL,**

**PRIMARY KEY (`id`),**

**UNIQUE INDEX `id\_UNIQUE` (`id` ASC),**

**INDEX `fk\_facultySubject\_faculty1\_idx` (`faculytId` ASC),**

**INDEX `fk\_facultySubject\_subject1\_idx` (`subjectId` ASC),**

**CONSTRAINT `fk\_facultySubject\_faculty1`**

**FOREIGN KEY (`faculytId`)**

**REFERENCES `admissionsBoardInno`.`faculty` (`id`)**

**ON DELETE CASCADE**

**ON UPDATE CASCADE,**

**CONSTRAINT `fk\_facultySubject\_subject1`**

**FOREIGN KEY (`subjectId`)**

**REFERENCES `admissionsBoardInno`.`subject` (`id`)**

**ON DELETE NO ACTION**

**ON UPDATE CASCADE)**

**ENGINE = InnoDB**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**-- -----------------------------------------------------**

**-- Table `admissionsBoardInno`.`request`**

**-- -----------------------------------------------------**

**CREATE TABLE IF NOT EXISTS `admissionsBoardInno`.`request` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`averageScore` FLOAT NOT NULL,**

**`facultyId` INT UNSIGNED NOT NULL,**

**`enrolleeId` INT UNSIGNED NOT NULL,**

**PRIMARY KEY (`id`),**

**UNIQUE INDEX `id\_UNIQUE` (`id` ASC),**

**INDEX `fk\_request\_faculty1\_idx` (`facultyId` ASC),**

**INDEX `fk\_request\_enrollee1\_idx` (`enrolleeId` ASC),**

**CONSTRAINT `fk\_request\_faculty1`**

**FOREIGN KEY (`facultyId`)**

**REFERENCES `admissionsBoardInno`.`faculty` (`id`)**

**ON DELETE NO ACTION**

**ON UPDATE CASCADE,**

**CONSTRAINT `fk\_request\_enrollee1`**

**FOREIGN KEY (`enrolleeId`)**

**REFERENCES `admissionsBoardInno`.`enrollee` (`id`)**

**ON DELETE CASCADE**

**ON UPDATE CASCADE)**

**ENGINE = InnoDB**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**-- -----------------------------------------------------**

**-- Table `admissionsBoardInno`.`statementRequest`**

**-- -----------------------------------------------------**

**CREATE TABLE IF NOT EXISTS `admissionsBoardInno`.`statementRequest` (**

**`id` INT UNSIGNED NOT NULL AUTO\_INCREMENT,**

**`request\_id` INT UNSIGNED NOT NULL,**

**`statement\_id` INT UNSIGNED NOT NULL,**

**PRIMARY KEY (`id`),**

**UNIQUE INDEX `id\_UNIQUE` (`id` ASC),**

**INDEX `fk\_statementRequest\_request1\_idx` (`request\_id` ASC),**

**INDEX `fk\_statementRequest\_statement1\_idx` (`statement\_id` ASC),**

**CONSTRAINT `fk\_statementRequest\_request1`**

**FOREIGN KEY (`request\_id`)**

**REFERENCES `admissionsBoardInno`.`request` (`id`)**

**ON DELETE NO ACTION**

**ON UPDATE CASCADE,**

**CONSTRAINT `fk\_statementRequest\_statement1`**

**FOREIGN KEY (`statement\_id`)**

**REFERENCES `admissionsBoardInno`.`statement` (`id`)**

**ON DELETE CASCADE**

**ON UPDATE CASCADE)**

**ENGINE = InnoDB**

**DEFAULT CHARACTER SET = utf8**

**COLLATE = utf8\_unicode\_ci;**

**Контрольные вопросы:**

1. **CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET utf8 COLLATE utf8\_unicode\_ci;**
2. **1.mysql –uroot –p –e “CREATE DATABASE newdb\_name”**

**2. mysqldump –uroot –p dbname >export.sql**

**3. mysql –uroot –p newdb\_name < export.sql**

**4. mysql –uroot –p –e “DROP DATABASE mydb”**

1. **DROP DATABSE `mydb`. Удаляются все файлы которые были созданы mysql, временные таблицы удаляются автоматически когда итсекаех их сессия, если в папке с базой данных лежали файлы, которые не были созданы mysql, то mysql не может удалить эту папку и ее прийдется удалить самостоятельно. Не удаляются привелегии которые были выделины для этой бд.**
   1. **SELECT \* FROM INFORMATION\_SCHEMA.SCHEMATA;**
      1. **ALTER DATABASE `mydb` DEFAULT CHARACTER SET ‘utf8’;**
      2. **ALTER DATABASE `mydb` DEFAULT COLLATE utf8\_unicode\_ci;**
   2. **SHOW TABLE STATUS FROM `mydb`;**
      1. **ALTER TABLE `mydb`.`user` CHARCTER SET utf8 COLLATE utf8\_unicode\_ci;**