

## ФИТНЕС

Като се използва БД по избор да се създаде приложение Фитнес.

### - Таблица Клиенти

- Id;
- FirstName;
- LastName;
- SubFrom;
- SubTo;
- SubType;
- AddedAt;
- NumberRenewings;
- Email
- PhoneNumber.

**Описание на полетата:** идентификационен номер, име, фамилия, абонамент от(дата), абонамент до(дата), тип абонамент, добавен на(регистрация на клиенти в БД - дата), брой подновявания(на абонамента), ел. поща и телефон. Може би е добро и един атрибут инструктор - дали ползва или не (true/false).

### - Таблица Служители

- Id
- FirstName
- LastName
- FirstWorkDayDate
- MonthSalary

**Описание на полетата:** идентификационен номер, име, фамилия, дата на постъпване и месечна заплата.

**Забележка:** 3 роли за служител:

- шеф, който има достъп до всичко;
- инструктор, който може да си провери графика, с кои клиенти работи
- касиер, който да регистрира нови клиенти, подновява абонаменти и т.н.

### - Таблица Продукти

- Id
- Name
- Amount
- Barcode
- Group
- LastDeliveryDate
- FutureDeliveryDate

**Описание на полетата:** идентификационен номер, име на продукт, количество, баркод, група (аминокиселини, протеини и пр.), дата на последна доставка и дата на бъдеща доставка.

- **Таблица Абонаменти**
  - Id
  - Name
  - Price
  - Duration(months)

**Описание на полетата:** идентификационен номер, име, цена, продължителност (в месеци).

- **Таблица Доставки**
  - Id
  - Provier
  - Date
  - Price
  - Amount.

**Описание на полетата:** Идентификационен номер, доставчик, дата, цена, количество.

- **Таблица Доставчици**

**Описание на полетата:** ид на доставчик, име на фирмата доставчик, адрес, телефон за връзка, e-mail и подобни.

- **Таблица Работни графици**

**Описание на полетата:** ид, ид на служител, смяна, работно време, дата и т.н.

- **Таблица Баланс**

/Да има ежедневен отчет за приходи и разходи/

**Описание на полетата:** ид, дата, вид на операция (приход или разход), не се сещам за други полета, но със сигурност трябва.

- **Таблица Оборудване**

**Описание на полетата:** ид, име (например дъмбел 5 кг.), брой, дата на доставка, брой ремонти, срок на годност, дата на бъдеща смяна, цена при закупуване. Нещо подобно.

- **Таблица Заплати**

/За всеки месец ще съхраняваме информация кой служител каква заплата е взел/

**Описание на полетата:** ид, ид на служител, месец и година, заплата

### **Справки:**

1. Списък с всички служители и техните заплати по месеци.
2. Месечен отчет с приходите и разходите.
3. Месечен и дневен отчет за посещенията на залата по график.
4. Определяне на служителите с най-висока и най-ниска заплата за последните 5 години.
5. Създаване на стълбчеста диаграма с посещаемостта по клиенти и дати.

### **Изисквания**

Задължително реализацията да включва използване на класове, колекции – list, dictionary, queue, stack и други структури съобразно тяхната необходимост.

### **Изисквания за предаване на курсов проект**

1. **Заглавна страница.** (1 стр) - курсов проект с трите имена, курс, специалност, факултет номер, дисциплина, преподавател (гл.ас.д-р Димитричка Николаева)
2. **Задание на проекта, описание.** (1 стр)
3. **Анотация.** (1 стр. Съдържа кратко описание на основните части и резултати на проекта)
4. **Увод.** (1 стр. Съдържа обосновка на актуалността на зададената тема.
5. **Обзор на съществуващите решения. Изводи. Цел и задачи.**  
(Около 1 стр. Съдържа описание и анализ на известните решения, при което се цитират съответните литературни източници. В подточката "Изводи" се дават в синтезиран вид предимствата и недостатъците на тези решения. В последната подточка се формулира целта, която се преследва с разработването на проекта и задачите, които трябва да бъдат решени, за да се постигне тази цел.)
6. **Проектиране и описание на предлаганото решение.** (от 2 до 10 стр.)
  - 2.1. Изисквания към програмната система.
  - 2.2. Логически модел на програмната система - представя се чрез диаграми на потоците от данни, UML или друг формален апарат за описание модела на системата.
  - 2.3. Архитектура на системата - отразява връзките и взаимодействието между програмните модули и данните.
  - 2.4. Организация на данните - концептуален модел на БД.
  - 2.5. Избор на език и среда за програмиране.
  - 2.6. Реализация на програмната система.
    - 2.6.1. Структура на данните - избор на ключови полета, тип и размер.
    - 2.6.2. Описание на програмните модули - изпълнявана функция, интерфейс между отделните програмни модули, обобщен алгоритъм (блок-схеми или псевдокод).
    - 2.6.3. Структура и организация на потребителския интерфейс.
    - 2.7. Формат на входните документи - формален и логически контрол.
    - 2.8. Формат на извежданите справки.
    - 2.9. Инструкции за работа с програмната система:
      - 2.9.1. Ръководство за потребителя - подробни сведения за начина на използване на системата, основни диалогови форми.
      - 2.9.2. Инструкции и изисквания при инсталиране на системата.
      - 2.9.3. Инструкции за поддържане на системата - указания за поддържане организацията и сигурността на данните и тяхното архивиране.
      - 2.9.4. Изисквания към апаратното осигуряване.
7. **Резултати от тестване на системата. СЪЗДАВАНЕ на поне два Unit Теста**
8. **Използвана литература.**

(Това е списък на цитираната и използвана в записката на проекта литература. Отделните заглавия се описват и подреждат по начина, показан по-долу.)

Боровски, Б. Х., А. Е. Егоров. Цифрови електронни изчислителни машини. София, Техника, 1989. Котров, С., М. Маринова, Я. Славова. Относно оценяване на качеството на обучение на студентите в специалност "Електротехника". Научни трудове на международната конференция "Качество на висшето образование", Варна, 27-28 октомври 2000.

Скворецкий, С. И., Е. И. Муратова. Особенности организации учебного процесса в техническом вузе в условиях профессионально-ориентированной информационной среды. Научные труды конференции ИТО'99, 1999.