|  |  |
| --- | --- |
| **МИНИСТЕРСТВО НА ОБРАЗОВАНИЕТО И НАУКАТА** | |
|  | ПРОФЕСИОНАЛНА ГИМНАЗИЯ ПО МЕХАНОТЕХНИКА “ПРОФЕСОР ЦВЕТАН ЛАЗАРОВ”  ***гр.Пловдив, ул. “Братя Бъкстон” №71 А, тел 032/671 143, 671 141, факс 032/ 671 143;***  e-mail:[***pgmtplovdiv@abv.bg***](mailto:pgmtplovdiv@abv.bg)  ***www. pgmtplovdiv.info*** |

**ДИПЛОМЕН ПРОЕКТ**

**ЗА ПРИДОБИВАНЕ НА ТРЕТА СТЕПЕН НА ПРОФЕСИОНАЛНА КВАЛИФИКАЦИЯ – ЧАСТ ПО ТЕОРИЯ НА ПРОФЕСИЯТА**

**по професия код 481030 „Приложен програмист“**

**специалност код 4810301 „Приложно програмиране“**

**Тема:**

**…………………………………………………………………………………………………..**

**Ученик**: …………Мария..Иванова..Арабска…………………………………

*(име, презиме, фамилия)*

**Клас:** XII “A”,

**Учебна година:** 2022/ 2023

**e-mail:** ……………………………….

**Ръководител-консултант**: ……..………………………..

*(име, фамилия)*

**гр. Пловдив**

**2022 год.**

**СЪдържание:**

1. Въведение

**1.1.** Въведение в предметната област на темата

**1.2.** Цел на дипломния проект

**1.3.** Структура на дипломния проект и очаквани резултати

1. Основна част

**2.1.** Описание на целта на дипломния проект и поставените задачи

**2.2.** Описание и анализ на известни решения по поставената тема

**2.3.** Описание на същинската разработка на дипломния проект

1. Заключение
2. Списък на използваната литература
3. Приложения

**1. Въведение**

**1.1. Въведение в предметната област на темата**

В днешно време, всеки втори ученик има проблеми с усвояването на учебния материал. Било то, защото в учебниците информацията е прекалено обемиста и част от нея даже излишна или начинът по който тя е представена. Сложните и дълги термини не помагат много също. За този проблем, някой вече е измислил решение. Всички са чували за сайта Уча.се, някои даже са го използвали. Този сайт е много полезен и удобен, но не всеки може да си го позволи поради таксата за абонамент. Затова нашата дипломна работа е с цел да подпомогне всички ученици, които изпитват затруднения с учебния материал и по- специално за подготовката за държавния изпит в 12 клас. В него ще намерите най-важната информация за произведенията и техните автори, както и самите произведения събрани на едно място, заедно с тестове и упражнения, с които да проверите знанията си.

**1.2. Цел на дипломния проект**

Главната цел на проекта е да представи компактно информацията, свързана с подготовката на зрелостниците за Държавния зрелостен изпит по Български език и литература.

**1.3. Структура на дипломния проект и очаквани резултати**

Дипломният проект представлява сайт от пет раздела- Начало, Автори, Тестове, Обратна връзка и За нас, в които са структурирани необходимите материали за подготовката по литература и произведенията от учебната програмата за 11 и 12 клас. Вътре са налични плановете на всичките двадесет и седем произведения, които се изучават през тези две учебни години. Разделени са на девет теми. В уебсайта има раздел, посветен на авторите на произведенията и тестове от Държавни зрелостни изпити през изминали години, с цел възможност за упражнение и по- добро усвояване на учебния материал по Държавните образователни изисквания.

**2. Основна част**

**2.1. Описание на целта на дипломния проект и поставените задачи**

Проекта трябва да включва в себе си възможно най-много полезна за младежите информация, за да бъде подготовката за Държавния им зрелостен изпит по Български език и литература възможно най-добра. За целта сме събрали информация за всички автори и произведения включени в програмата за 11 и 12 клас. Задачите, които са ни необходими да изпълним, за да се стигне до крайния вариант и реализирането на идеята за сайта са следните:

* Разработка на ER диаграма
* Създаване на база данни и SQL скрипт, който да създава схемата на таблиците, в които ще се съдържа цялата ни информация за споменатите по- горе неща.
* Създаване на SQL скриптове за вкарване на информацията вътре в самите таблици
* Създаване на екраните съдържащи се в сайта като HTML и CSS.
* Добавяне на PHP за по-голямо удобство, гъвкавост и яснота на кода.

**2.2. Описание и анализ на известни решения по поставената тема**

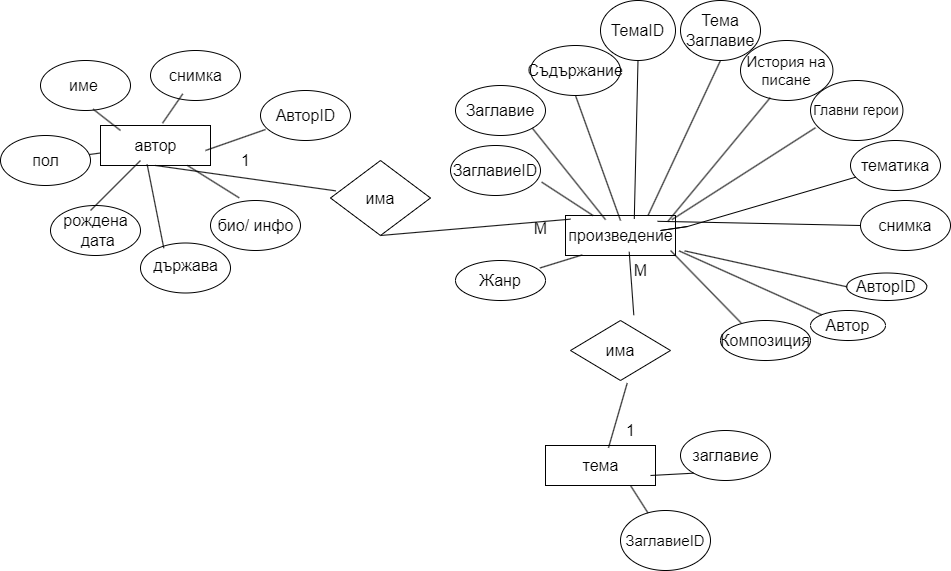
**2.3. Описание на същинската разработка на дипломния проект**

Първата ни задача е да създадем ER диаграма, в помощ за създаването на нашата база данни.

ER – диаграмите се използват за моделиране и проектиране на релационни бази данни. Те отразява обектите, техните свойства и взаимодействията (връзки) между тях. Основните компоненти на ER-диаграмите са:

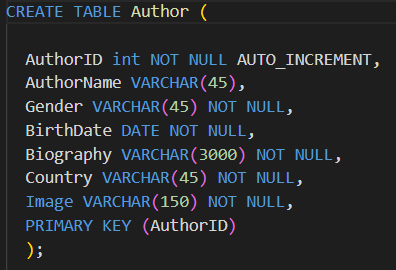
* Обект(Entity) –представлява всеки един самостоятелен обект от предметната област- отбелязва се с правоъгълник. Например човек, студент, кола, къща…в нашия случай обекти ще са темите, произведенията и авторите.
* Характеризиращ обект – обект, който не може да съществува извън рамките на предметната област. В нашата диаграма такъв обект е „автора“, характеризиращ обект в една база данни за произведенията.
* Атрибути – представляват специфичните свойства на обектите. Отбелязват се с елипса. Например атрибутите на обекта автор ще са име, пол, година на раждане и биография.
* • Връзки– Представят се като ромбове, в средата на които се пише името на връзката а в двата края до релациите, които се свързват се поставят 1,M за да се покаже типа на връзката.

Диаграмата, която ще създадем, за да визуализираме нашата идея за създаване на таблици, и връзките между тях ще съдържа три обекта- автор, произведение и тема.

Всички те имат по няколко атрибута. Обектът автор съдържа атрибутите АвторID, име на автора, пол, дата на раждане, държава, биография и снимка. Обектът прави връзка 1:М или едно към много с друг обект, а имено обектът произведение. Той има 13 атрибута- произведениеID, произедение, авторID, автор, темаID, тема или категория, съдържание, история на писане, жанр, композиция, главни герои, тематика и снимка. Този обект има още една връзка. Тя сочи към обекта теми. Той съдържа само два атрибута и това са тема или категорияID и тема/категория. Връзката между тях е М:1 или много към едно, защото много произведения може да се съдържат в една тема, но те не могат да се повтарят в другите теми и това е причината връзката да е такава. Обектите автор и тема могат да бъдат представени също и като характеризиращи обекти на обекта произведение. До тук скицата ни изглежда така:

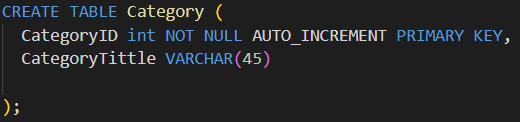
Сега след като първата ни задачата е готова ние се залавяме със следващата- да създадем база данни и SQL скрипт, който да създава схемата на таблиците, в които ще се съдържа цялата ни информация.

За целта ние ще използваме безплатния и свободен софтуерен пакет XAMPP, който е базиран на локален хост сървър, включващ софтуерни пакети и предлагащ няколко функционалности. С него може да се тестват модификации преди да се пуснат в интернет. MySQL базата данни формира важен компонент на XAMPP Stack, като една от най-значимите функции, предлагани от нея. Благодарение на XAMPP може да използваме phpMyAdmin и MySQL, за да създадем нашата база данни и онагледим нашата ER диаграма.

Първо си създаваме база данни, в която ще добавим нашите обекти, които ще бъдат представени във формата на таблици следователно атрибутите им ще са подтаблици. Започваме създаването им с таблица за автора. На снимката виждаме код, който е SQL заявка. Тя се използва за създаване на таблицата "Author". Таблицата има следните колони:

* AuthorID: целочислен тип данни, който е зададен като основен ключ на колоната и автоматично нараства за всеки нов запис, добавен към таблицата.
* AuthorName: променлив тип символни данни с максимална дължина от 45 знака. Колоната се използва за съхраняване името на автора.
* Gender: променлив тип символни данни със зададена от нас максимална дължина- 45 знака. Тази колона съхранява пола на автора.
* BirthDate: тип данни - дата, която съхранява датата на раждане на автора.
* Biography: променлив тип символни данни с максимална дължина от 3000 знака. Тази колона съхранява кратка биография за автора.
* Country: променлив тип символни данни с максимална дължина от 45 знака. Тя съхранява страната на произход на автора.
* Image: променлив тип символни данни с максимална дължина от 150 знака. Нейната функция е да съхранява пътя до файл с изображение на автора.

Важно да отбележим е, че всички колони имат зададено "NOT NULL" ограничение, което означава, че те задължително трябва да имат стойност за всеки запис, добавен към таблицата.

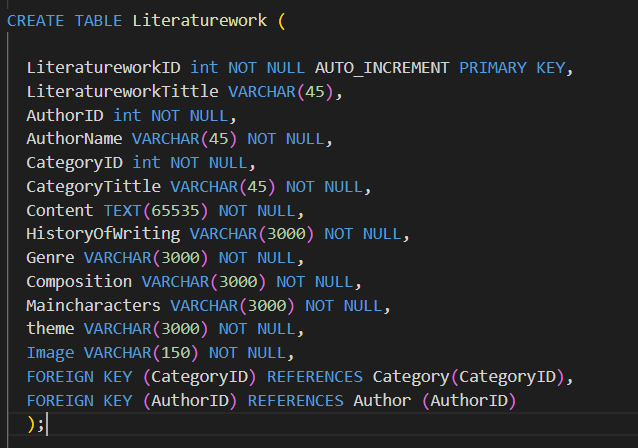
Следващият обект, който ще превърнем в таблица е този за темите. Ние ще си го кръстим category.

Създаваме таблица "Category" Тази таблица има две колони:

CategoryID: целочислен тип данни, който е зададен като основен ключ на колоната и автоматично нараства за всеки нов запис, добавен в таблицата. И тук както в предишната таблица имаме ограничението “NOT NULL”, следователно тази стойност задължително трябва да бъде попълнена.

* CategoryTittle: променлив тип символни данни с максимална дължина от 45 знака. Тази колона ще съхранява заглавието на темата.

Преминаваме към третия и последен обект- произведение. Създаваме си таблица под името Literaturework.

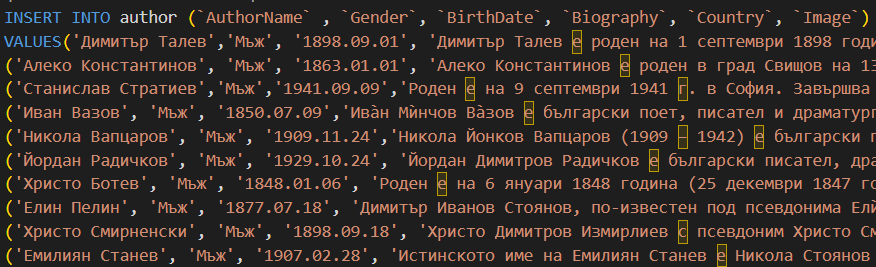


Таблицата има 13 колони:

* LiteratureworkID: целочислен тип данни, който е зададен като основен ключ на колоната и автоматично нараства за всeки нов запис, който ще бъде добавен към таблицата.
* LiteratureworkTittle: променлив тип символни данни с максимална дължина от 45 знака. Задачата на тази колона е да съхранява заглавието на литературното произведение.
* AuthorID: целочислен тип данни, който е свързан с основния ключ на таблицата "Author". Тази колона съхранява идентификатора на автора на литературното произведение.
* AuthorName: променлив тип символни данни с максимална дължина от 45 знака. Тя записва името на автора на литературното произведение.
* CategoryID: целочислен тип данни, който е свързан с основния ключ на таблицата "Category". Тази колона съхранява идентификатора на категорията, към която принадлежи литературното произведение.
* CategoryTittle: променлив тип символни данни с максимална дължина от 45 знака. Тя пази заглавието на категорията, към която принадлежи литературното произведение.
* Content: текстов тип данни с максимална дължина от 65535 символа. Тази колона съхранява съдържанието на литературното произведение. Зададен й е голям брой за максимални символи, поради факта че има много обемисти по съдържание произведения.
* HistoryOfWriting: променлив тип символни данни с максимална дължина от 3000 знака. Колона съхранява информация за творческата историята за създаването на литературното произведение.
* Genre: променлив тип символни данни с максимална дължина от 3000 знака. Тя съхранява информация за жанра на произведението.
* Composition: променлив тип символни данни с максимална дължина от 3000 знака. Колоната е с предназначение да съхранява информация за композицията на литературната творба.
* Maincharacters: променлив тип символни данни с максимална дължина от 3000 знака. Тази колона ще съхранява информация за всички основни герои в литературното произведение.
* Theme: променлив тип символни данни с максимална дължина от 3000 знака. Служи за съхранение на информация за темата, която се обсъжда в произведението.
* Image: променлив тип символни данни с максимална дължина от 150 знака. Колоната съхранява връзка към изображението за литературното произведение.

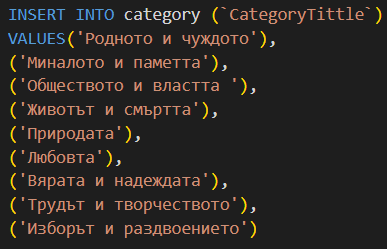
Сега след като сме приключили с важната за проекта задача е време да напълним тези таблици с необходимата за тях информация. Това ще направим чрез скриптове, както до сега, а не ръчно. Причината за това е риска информацията да се загуби и да я няма съхранена никъде, което ще доведе до подновяването на процеса за вкарването на информация.

Скрипт за добавяне на информация в таблица автори са оформени по следния начин.

Стойностите, които трябва да бъдат добавени, са предоставени като списък от разделени със запетаи стойности, затворени в скоби, съответстващи на атрибутите на таблицата в същия ред, в който са дефинирани. В този случай стойностите, които се въвеждат са:

* **AuthorName:** 'Димитър Талев'
* **Gender:** 'Мъж'
* **BirthDate:** '1898-09-01'
* **Biography:** 'Димитър Талев е роден на 1 септември 1898 г. в гр. Бургас. Български поет, прозаик и драматург. Известен е с прозвището "Кумчо Димитър"...'
* **Country:** 'България'
* **Image:** 'Images/Dimitar\_Talev.jpg'

Командата ще създаде нов ред в таблицата "author" с тези стойности за всеки атрибут. Атрибутът на основния ключ "AuthorID" не е включен в списъка със стойности, защото е зададен да се увеличава автоматично, което означава, че базата данни автоматично ще генерира уникална стойност за него, когато се добавя нов ред. По този начин се добавят всички автори.

Сега преминаваме към другите скриптове за добавяне на информация. Те работят по абсолютно същия начин, но все пак нека ги разгледаме. Започваме с заявките за category.

Стойностите, които трябва да бъдат вмъкнати, се предоставят като списък от разделени със запетаи стойности, затворени в скоби като скрипта по- горе.В този случай стойностите, които се вмъкват, са:

* CategoryID: 1, CategoryTittle: 'Родното и чуждото'
* CategoryID: 2, CategoryTittle: 'Миналото и паметта'
* CategoryID: 3, CategoryTittle: 'Обществото и властта '
* CategoryID: 4, CategoryTittle: 'Животът и смъртта'
* CategoryID: 5, CategoryTittle: 'Природата'
* CategoryID: 6, CategoryTittle: 'Любовта'
* CategoryID: 7, CategoryTittle: 'Вярата и надеждата'
* CategoryID: 8, CategoryTittle: 'Трудът и творчеството'
* CategoryID: 9, CategoryTittle: 'Изборът и раздвоението'

Командата ще създаде нов ред в таблицата "category" с тези стойности за всеки атрибут. Атрибутът "CategoryID" е ключов за таблицат и не е включен в списъка със стойности, защото е настроен да се инкрементира автоматично, което означава, че базата данни ще му генерира стойност.

Останаха последните и най- трудните за вкарване скриптове, защото в тях се съдържа много повече информация.

INSERT INTO literaturework (`LiteratureworkID`,`LiteratureworkTittle`,`AuthorID`,`AuthorName`,`CategoryID`,`CategoryTittle`,`Content`,`HistoryOfWriting`,`Genre`,`Composition`,`Maincharacters`,`Theme`,`Image`)

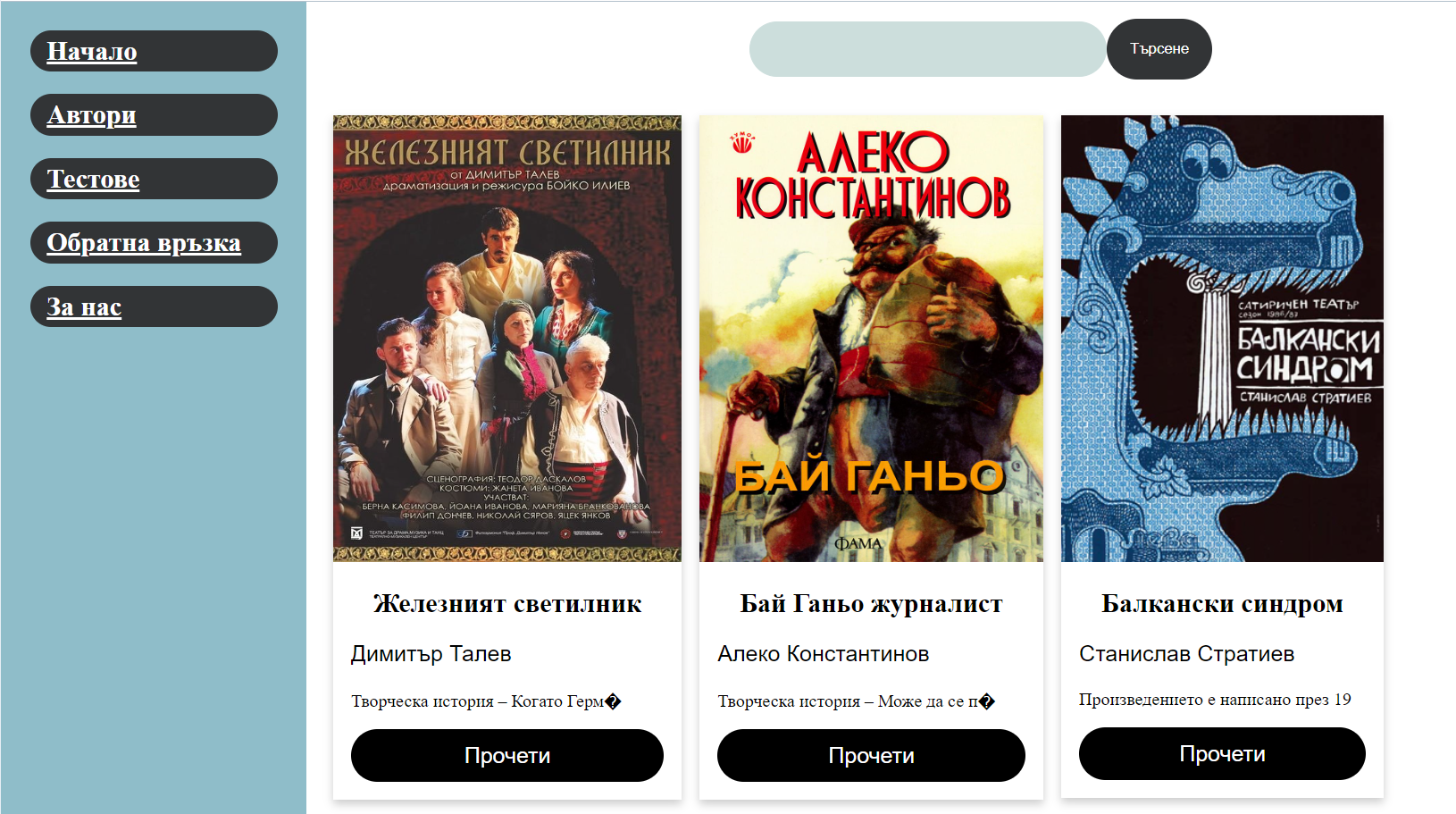
VALUES

Тук стойностите са 13 на брой, но принципа на вкарване на информацията не остава по различен.

И сега след като сме приключили с нашата база данни и сме сложили необходимата информация, вече можем да започнем работа по екраните ни.

Както в началото споменахме, сайтът се състои от пет раздела- Начало, Автори, Тестове, Обратна връзка и За нас. Освен тези раздели(екрани) сайтът разполага с още 3, които се отварят при навигиране до дадено произведение. Това прави нашия сайт осемекранен.

Първият екран, който ще разгледаме е този, който се отравя след изиране на тема от началния екран, а имено избиращия кое произведение да се отвори или както ние сме кръстили файла - ChooseScreen.php.



За да се постигне визията на този екран, той е минал през три езика- HTML, CSS и PHP.

HTML е основният маркиращ език за описание и дизайн на уеб страници. HTML елементите са основната градивна единица на кода, който изграждат уеб страниците. Чрез тях се форматира, графично оформя текста и неговите отделните части в рамките на една уеб страница, като заглавия, цитати, текстови раздели и хипертекстови препратки. Такива ние имаме във всичките ни страници, целият сайт се гради на тях. Характерни HTML елементите за сайта и които се съдържат във всички старици са <head>, <header>, <body> и <nav> заключени едни между други.

**3. Заключение**

**4. Списък на използваната литература**

**5. Приложения**