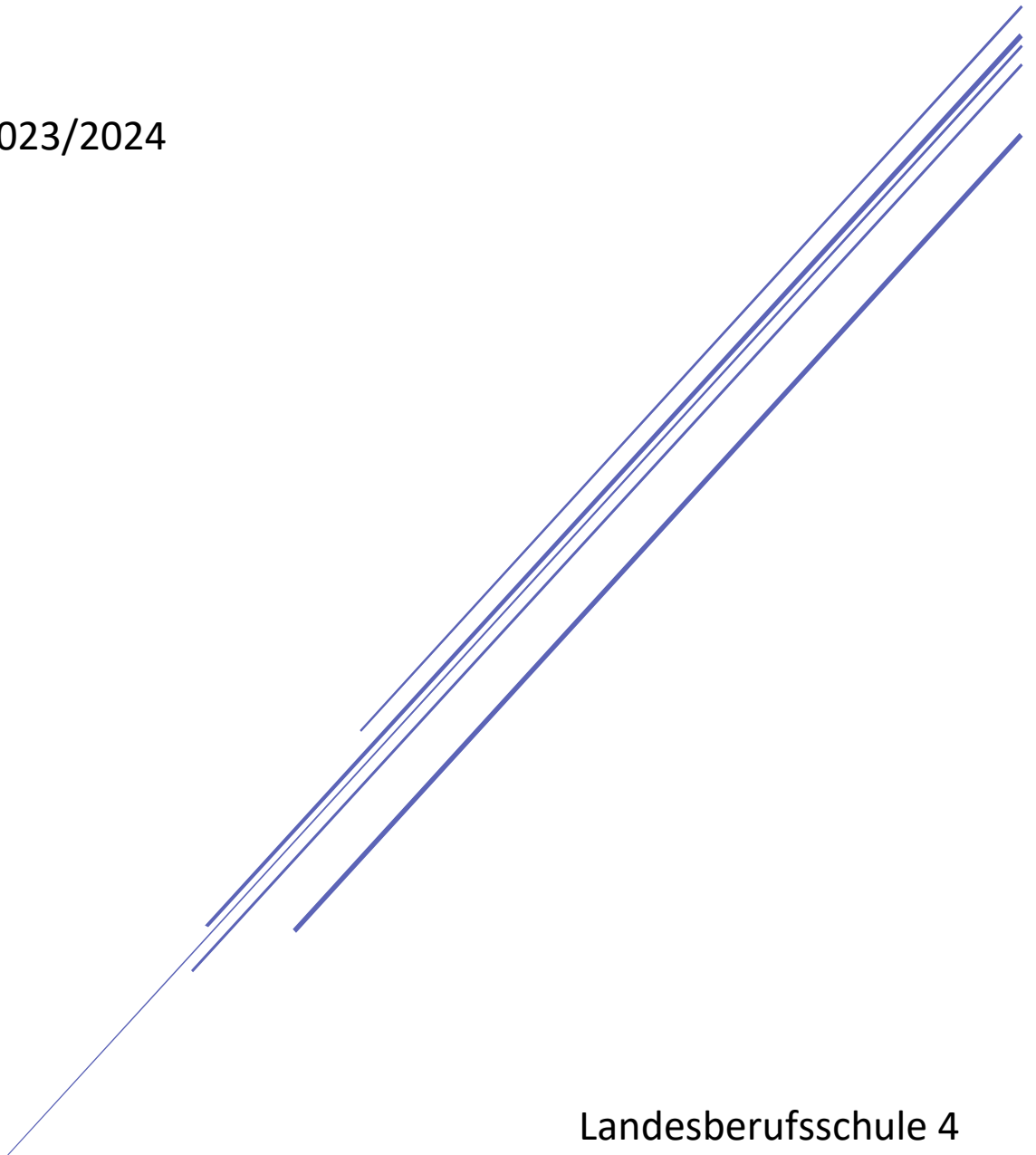


VORBEREITUNG LAP

Jahr: 2023/2024



Landesberufsschule 4

Inhalt

1	Praktische Prüfung.....	- 3 -
1.1	Mögliche Tools	- 3 -
2	Aufgabenstellungen.....	- 3 -
2.1	Teil 1 Datenbankdesign.....	- 3 -
2.1.1	Anforderung	- 3 -
2.1.2	Übung.....	- 3 -
2.1.3	SQL-Datenbankerzeugung	- 4 -
2.1.4	Testen und Dokumentieren	- 5 -
2.2	Teil 2 Datenbank mit Benutzeroberfläche	- 5 -
2.2.1	Allgemeine Anforderungen.....	- 5 -
2.2.2	mögliche Vorgehensweise	- 6 -
2.3	Oberfläche für Suchergebnisse.....	- 6 -
2.3.1	Grundgerüst des Projektes	- 6 -
2.3.2	Erstellen der Suche	- 8 -
2.3.3	Einfügen eines Patienten	- 8 -
3	Ergebnis.....	- 8 -
4	Zusätzliche Ressourcen	- 9 -
4.1	Datenbank und PHP.....	- 9 -
4.2	CSS und HTML	- 9 -
4.3	PHP-Allgemein.....	- 9 -

1 Praktische Prüfung

Als Übung für die praktische Prüfung werden 2 Datenbanken erstellt.

Entwickeln Sie innerhalb einer LAMPP oder XAMPP Umgebung eine Patientenverwaltung und eine einfache Filmdatenbank.

Als Entwicklungsumgebung verwenden Sie Freeware oder Open-Source Programme die selbstständig installiert werden müssen.

1.1 Mögliche Tools

- [Visual Studio Code als Entwicklungsumgebung / IntelliJ Toolbox](#)
- [Xampp](#)
- [PHP-IntelliSense für VSC](#)
 - [Konfiguration von VSC](#)
- IntelliJ-IDEA
- [MySQL-Workbench](#)
- [Bootstrap](#)

2 Aufgabenstellungen

2.1 Teil 1 Datenbankdesign

2.1.1 Anforderung

Zu der Übung muss ein Protokoll mit dem ER-Modell, einem Skript für die Daten und dem Datenbanktest erstellt werden (Dauer ca. 2,5 Stunden).

2.1.2 Übung

Erstellen Sie eine Datenbank für eine Patientenverwaltung. Die Datenbank heißt Arztpraxis

2.1.2.1 allgemeine Anforderung

- das ER-Diagramm muss in der Normalform 3 sein (begründen)
- tragen Sie in das Modell die Funktionalität der Beziehungen (Kardinalität) ein (1:1, 1:n)
- fügen Sie bitte für die jeweilige Entität auch passende Attribute hinzu

2.1.2.2 Anforderungen an das ER-Modell

Das Modell wird durch folgende Angaben beschrieben:

- eine Arztpraxis betreut mehrere Patienten
- ein Patient hat maximal eine bestimmte Sozialversicherung, eine Sozialversicherung kann mehrere Patienten haben
- jeder versicherte Patient hat eine eindeutige Sozialversicherungsnummer
- ein Patient kann mehrere Befunde haben, jeder Befund ist immer genau einem Patienten zugeordnet
- in einem Befund können ein oder mehrere Medikamente verordnet werden
- in dieser Medikamentenverordnung wird auch die Dosierung angegeben
- dasselbe Medikament kann mit mehreren Befunden verordnet werden
- jeder Termin mit einem Patienten muss einem Terminverlauf zugeordnet sein. Bei jedem Befund muss nachvollziehbar sein, bei welchem Termin dieser Befund erstellt wurde

2.1.3 SQL-Datenbankerzeugung

Erstellen Sie nun aus Ihrem ER-Diagramm eine SQL-Datenbank mit Beispieldaten.

Arbeitsanweisung:

- benennen Sie die Datenbank Patientenverwaltung
- generieren Sie aus dem ERM eine Datenbankschema

- geben Sie dem Datenbankverwaltungssystem bekannt, dass Sie die Datenbank Patientenverwaltung verwenden
- achten Sie auf eine saubere Formatierung innerhalb der SQL-Datei.

Beachten Sie bitte, dass pro Tabelle mindestens 3 Datensätze enthalten sind.

2.1.4 Testen und Dokumentieren

Testen Sie nun ausführlich Ihre SQL-Datenbank und geben Sie geeignete Datensätze ein, protokollieren Sie Ihre Vorgangsweise beim Testen der Datenbank.

Erstellen Sie eine Abfrage, welche die Anzahl der Patienten in einer Arztpraxis zeigen.

Erstellen Sie eine Abfrage, die die Anzahl der Befunde pro Patienten mit Namen anzeigen, es soll mindestens 1 Befund vorhanden sein.

2.2 Teil 2 Datenbank mit Benutzeroberfläche

Verwenden Sie die bereitgestellte Datenbank und erstellen Sie mit PHP, HTML5 eine einfache Webseite für Abfragen und Inserts der Datenbank. Realisieren Sie zumindest ein Insert für den Patienten mit Adresse (Stadt, plz), das Land kann vorgegeben werden.

2.2.1 Allgemeine Anforderungen

- einfache Webseite (Startseite)
- Navigation
- Seite für das Einfügen von Datensätzen
- Seite für Abfragen der Datenbank
- Prepared Statements
- wichtige Kommentare

2.2.2 mögliche Vorgehensweise

Verwenden Sie die vorgegebene Datenbank und passen Sie (wenn notwendig) Attribute und Tabellen an. Dokumentieren Sie die Arbeitsschritte und begründen Sie die Vorgangsweise sowie die Auswahl der Tools.

2.3 Oberfläche für Suchergebnisse und Inserts

Die Darstellung der Ergebnisse soll mittels HTML5 (responsiv) und die Auswertung der Datenbank mit PHP erfolgen.

2.3.1 Grundgerüst des Projektes

Erstellen Sie ein Grundgerüst für HTML. Es kann Bootstrap für ein schnelles Design verwendet werden. Ich verwende ein Gerüst mit einzelnen PHP-Seiten mit header, footer und nav.

Die nachstehenden Quellcodeausschnitte sind von der Webseite [Start with Bootstrap](#)

Erstellen Sie eine header.php im Projektverzeichnis und kopieren Sie den Quellcode in die Datei. Das stellt den Head und den Anfang vom body bereit.

```
<!doctype html>
<html lang="de">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Bootstrap demo</title>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.0/jquery.min.js"></script>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-9ndCyUalbzAi2FUVXJi0CjmCapSmO7SnpJef0486qhl-
nuZ2cdeRhO02iuK6FUUVM" crossorigin="anonymous">
  </head>
  <body>
```

In die footer.php den nachstehenden Code einfügen

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"
integrity="sha384-geWF76RCwLtnZ8qwWowPQN-
guL3RmwHVBC9FhGdlKrxdiJJigb/j/68SIy3Te4Bkz" crossorigin="anonymous"></script>
</body>
</html>
```

Für die nav.php der nachstehende Quellcode.

```
<nav class="navbar navbar-expand-lg bg-body-tertiary">
  <div class="container-fluid">
    <a class="navbar-brand" href="index.php">Startseite</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarNav">
      <ul class="navbar-nav">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" href="#">Suche</a>
        </li>
      </ul>
    </div>
  </div>
</nav>
```

Damit sind die Teile für die index.php fertig.

Als nächstes erstellen Sie eine Datei für die Datenbankverbindung (connect_database.php) mit nachstehenden Quellcode. Passen sie die Parameter an.

```
<?php
//Connect with mySQLi
$hostname="localhost";
$username="root";
$password="";
$databse="handy";
$conn = mysqli_connect($hostname, $username, $password, $databse) or die("Connection failed: " . mysqli_connect_error());
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}
?>
```

Oder mit PDO

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$databse = "myDatabase";

try {
```

```
$conn = new PDO("mysql:host=$servername;dbname=$database", $username, $password);  
// set the PDO error mode to exception  
$conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);  
//bei Bedarf anzeigen lassen  
//echo "Connected successfully";  
} catch(PDOException $e) {  
    echo "Connection failed: " . $e->getMessage();  
}  
?>
```

In der index.php werden jetzt die php-Seiten eingebunden.

```
<?php  
include('header.php');  
?>  
<?php include_once('nav.php') ?>  
<article id="welcome">  
    <p> <h1>Willkommen auf der Testseite</h1></p>  
</article>  
<?php include('footer.php') ?>
```

Legen Sie zwei Seiten für die Suche und das Einfügen der Information na. Kopieren Sie die dazu die index.php.

2.3.2 Erstellen der Suche

In der Suchseite wird ein Formular erstellt. Das Ergebnis wird auf derselben Seite ausgegeben. Siehe Quellcode search.php

2.3.3 Einfügen eines Patienten mit Adresse

In dieser Datei (insert.php) wird ein Formular zur Verfügung gestellt und der Datensatz in die Datenbank eingegeben (siehe insert.php).

3 Ergebnis

Für die Aufgabe stehen ca. 3 Stunden bereit und als Ergebnis soll,

- ein ER-Diagramm mit Angabe der Kardinalität (Reverse-Engineering)
- die Dokumentation der Arbeitsschritte

- minimale Dokumentation des Quellcodes
- Funktionalität
- minimale CSS-Formatierungen für ein lesbares Design oder eben Bootstrap

vorliegen. Das Ergebnis muss im Anschluss dem Prüfer erklärt werden können!

Alles Gute!

4 Zusätzliche Ressourcen

In diesem Abschnitt finden Sie weitere Links zu Ressourcen und Code-Schnipseln für PHP, HTML und CSS.

4.1 Datenbank und PHP

[Vergleich PDO und MySQLi](#) , [PDO](#)

4.2 CSS und HTML

[W3Schools CSS](#), [W3Schools HTML-Semantik](#)

[CSS-Aufbau, Bootstrap](#)

4.3 PHP-Allgemein

[PHP-Formulare](#)

[Tabelle und sortieren](#)

[PHP-Einsteiger](#)