

Lehrplan

## **Berufliche Informatik**

Gymnasiale Oberstufe

Berufsbezogene Fachrichtungen:

Gesundheit und Soziales

Wirtschaft

Hauptphase

Ministerium für Bildung und Kultur

Trierer Str. 33  
66111 Saarbrücken

Saarbrücken, Juli 2015

Hinweis:

Der Lehrplan ist online verfügbar unter  
[www.saarland.de/lehrplaene.htm](http://www.saarland.de/lehrplaene.htm)

## Einleitende Hinweise

Dem vorliegenden Lehrplan im Fach Berufliche Informatik der zweijährigen Hauptphase an den gymnasialen Oberstufen der berufsbezogenen Fachrichtungen Gesundheit und Soziales sowie Wirtschaft liegen die Verordnung – Schul- und Prüfungsordnung – über die gymnasiale Oberstufe und die Abiturprüfung im Saarland (GOS-VO) vom 02.07.2007, geändert durch die Verordnung vom 26.03.2010, sowie die Vereinbarung zur Gestaltung der gymnasialen Oberstufe in der Sekundarstufe II – Beschluss der Kultusministerkonferenz – vom 07.07.1972 i. d. F. vom 06.06.2013 zu Grunde.

Die Einheitlichen Prüfungsanforderungen in der Abiturprüfung Berufliche Informatik (Beschluss der Kultusministerkonferenz vom 1.6.1979 i.d.F. vom 10.05.2007) sind im Lehrplan umgesetzt und werden im Hinblick auf die Abiturprüfung in den Allgemeinen Prüfungsanforderungen für das Abitur im Fach Berufliche Informatik (APA Berufliche Informatik, Juli 2015) konkretisiert.

Mit dem Eintritt in die gymnasiale Oberstufe wachsen neben den inhaltlichen und methodischen Anforderungen auch die Anforderungen an die Selbstständigkeit des Lernens und Arbeitens, an die Verantwortung für die Gestaltung des eigenen Bildungsganges sowie an die Fähigkeit und Bereitschaft zur Verständigung und Zusammenarbeit in wechselnden Lerngruppen mit unterschiedlichen Lebens- und Lernerfahrungen.

Bei der Umsetzung des Lehrplans werden grundlegende Konzepte der Beruflichen Informatik sowie informatische Methoden entlang von Projekten zur Erstellung eines Webauftritts vermittelt. Dabei bearbeiten die Lernenden reale berufstypische Aufgabenstellungen.

Die Zielsetzung des Projektansatzes ist, dass über die Vermittlung von Fachinhalten wichtige Kompetenzen vermittelt werden. Schülerinnen und Schüler lernen komplexe Aufgabensituationen kennen, entwickeln die Fähigkeit zur Selbstorganisation, erwerben soziale Kompetenzen und lernen Methoden zur Bewältigung von Handlungssituationen kennen.

Auf nachstehende formale Vorgaben wird verwiesen:

- Die Lernziele sind mit Blick auf einen stringenten Umfang des Lehrplans als Groblernziele formuliert.
- Die Zeitrictwerte sind als vorgeschlagene zeitliche Empfehlung zu verstehen. Sie sind stets als Jahresstunden ausgewiesen, um Vergleiche mit Schulformen anderer Bundesländer zu ermöglichen.
- Bei den ausgewiesenen Stundenanteilen sind die Zeiten für Wiederholungen, Leistungsüberprüfungen usw. enthalten, die mit rd. einem Drittel angesetzt sind.

Saarbrücken, Juli 2015

## Übersicht über die Lerngebiete

Lfd. Nr.	Lerngebiet	Zeitrichtwert * (in Unterrichtsstunden)
	<b>Hauptphase (Halbjahre 1 und 2)</b>	
1	Gestalten von statischen Webseiten mit HTML und CSS	30
2	Strukturierung von Webauftritten	20
3	Veröffentlichung von Webinhalten	10
4	Einsatz von Content-Management-Systemen	10
5	Rechtliche und gesellschaftliche Aspekte im Internet	10
Summe		80
	<b>Hauptphase (Halbjahre 3 und 4)</b>	
6	Grundlagen der Programmierung von Webanwendungen mit einer clientseitigen Skriptsprache	38
7	Objektorientiertes Programmieren von Webanwendungen mit einer clientseitigen Skriptsprache	20
8	Erstellen und Verwalten relationaler Datenbanken für Webauftritte	24
9	Webbasiertes Auswerten von relationalen Datenbanken	38
10	Web-Projekte	40
Summe		160

\* Zeitrichtwert i. S. eines Vorschlags

<b>Lerngebiet 1: Gestalten von statischen Webseiten mit HTML und CSS</b>		Zeitrichtwert : 30 Stunden
Lernziele	Lerninhalte	Hinweise zum Unterricht
Die Lernenden		
1.1 kennzeichnen das WWW als System von Webseiten	<ul style="list-style-type: none"> <li>- HTML</li> <li>- Client-Server-Prinzip</li> <li>- Hypertext-Transfer-Protokoll</li> <li>- Hyperlinks</li> <li>- Homepage, Website</li> </ul>	erste Seite: index.html
1.2 beschreiben Ziele eines Internetauftritts	<ul style="list-style-type: none"> <li>- Zielgruppe</li> <li>- Informationsbereitstellung</li> </ul>	Wen möchte ich ansprechen? Was möchte ich kommunizieren?
1.3 beurteilen Internetseiten	<ul style="list-style-type: none"> <li>- inhaltliche Kriterien</li> <li>- gestalterische Kriterien</li> <li>- rechtliche Kriterien</li> <li>- Benutzerfreundlichkeit</li> </ul>	Korrektheit, Aktualität, klare Gestaltung, Corporate Design, Einhaltung rechtlicher Bestimmungen, schnelle Übertragung, benutzerfreundliche Navigationsstruktur
1.4 erläutern Werkzeuge zum Aufbau einer Webpräsenz	<ul style="list-style-type: none"> <li>- Browser</li> <li>- Editor</li> <li>- Grafikprogramm</li> </ul>	Open-Source-Programme
1.5 erstellen HTML-Seiten	<ul style="list-style-type: none"> <li>- Struktur einer HTML-Seite</li> <li>- offene und schließende TAGs</li> <li>- Kommentare</li> <li>- Webordner</li> </ul>	Aktueller HTML-Standard ist zu beachten (WW3), striktes HTML. Ein Webordner ermöglicht einen konsistenten Upload.
1.6 unterscheiden Blockelemente und Inline-Elemente	<ul style="list-style-type: none"> <li>- Beziehung zwischen Inline-Elementen und Blockelementen</li>   <li>- Auswirkung von Blockelementen auf Textfluss</li> </ul>	Inline-Elemente müssen in Blockelementen stehen. Nur Blockelemente können andere Blockelemente enthalten. Absatzschaltung

**Lerngebiet 1:**

Zeitrichtwert : 30 Stunden

**Gestalten von statischen Webseiten mit HTML und CSS**

Lernziele	Lerninhalte	Hinweise zum Unterricht
Die Lernenden		
1.6 strukturieren Texte über Blockelemente einheitlich	<ul style="list-style-type: none"> <li>- Struktur eines Artikels</li> <li>- Kopfbereich</li> <li>- Überschriften</li> <li>- Abschnitte und Absätze</li> <li>- Randbemerkungen</li> <li>- Fußbereich</li> </ul>	<pre>&lt;article&gt; &lt;header&gt; &lt;h1&gt; &lt;h2&gt; ... &lt;section&gt;, &lt;p&gt; &lt;aside&gt; &lt;footer&gt;</pre>
1.7 zeichnen Texte über Inline-Elemente nach Bedeutung aus	<ul style="list-style-type: none"> <li>- Betonungen und Hervorhebungen</li> <li>- Definitionen und Beispiele</li> <li>- Zitate</li> <li>- Abkürzungen</li> </ul>	<pre>&lt;strong&gt;, &lt;em&gt; &lt;mark&gt; &lt;dfn&gt;, &lt;samp&gt; &lt;q cite="Quelle"&gt; &lt;abbr&gt;</pre>
1.8 bereiten Grafiken mit einem Bildbearbeitungsprogramm für einen Webauftritt auf	<ul style="list-style-type: none"> <li>- Farbdarstellung von RGB-Grafiken</li> <li>- Grafikformate und Bildqualität</li> <li>- Bildgröße</li> </ul>	Hexadezimale Farbdefinition, Farbmanipulationen, Transparenz, Farbtiefe, Bildgröße ändern und Bilder zuschneiden
1.9 binden Grafiken in HTML-Seiten barrierefrei ein	<ul style="list-style-type: none"> <li>- Eigenschaften des &lt;img&gt;-TAGs</li> <li>- Barrierefreiheit</li> </ul>	alleinstehend, inline  Screen-Reader lesen den Inhalt des Attributs alt.
1.10 erläutern den Aufbau und die korrekte Schreibweise von TAGs	<ul style="list-style-type: none"> <li>- Attribute und Attributwerte</li> <li>- Kleinschreibung, Anführungszeichen, Leerzeichen</li> </ul>	
1.11 binden Strukturelemente in HTML-Seiten ein	<ul style="list-style-type: none"> <li>- interne und externe Hyperlinks</li> <li>- Listen und Aufzählungen</li> <li>- Tabellen</li> </ul>	Sprungmarken, target-Attribut

**Lerngebiet 1:** Zeitrichtwert : 30 Stunden  
**Gestalten von statischen Webseiten mit HTML und CSS**

Lernziele	Lerninhalte	Hinweise zum Unterricht
Die Lernenden		
1.12 erläutern die Vorteile der Formatierung von Webseiten über Cascading Style Sheets (CSS)	<ul style="list-style-type: none"> <li>- Trennung zwischen Inhalt/Struktur und Formatierung</li> <li>- einheitliches Layout</li> <li>- Arbeitersparnis</li> </ul>	CSS ermöglicht verteiltes Arbeiten an einer Website.
1.13 beschreiben die Möglichkeiten der Einbindung von CSS	<ul style="list-style-type: none"> <li>- externe CSS-Datei</li> <li>- eingebettet am Anfang der HTML-Datei</li> <li>- direkt im Quellcode (inline)</li> </ul>	Bei der Entwicklung von Websites sind externe CSS-Dateien vorzuziehen. <span>-TAG, inline-Styles überlagern externe Styles.
1.13 erläutern den Aufbau und die Schreibweise von CSS- Regeln	<ul style="list-style-type: none"> <li>- Selektor</li> <li>- Attribute</li> <li>- Attributwerte</li> </ul>	Der Selektor entspricht dem zu formatierenden TAG im HTML-Code.
1.14 wenden CSS-Regeln zur Formatierung an	<ul style="list-style-type: none"> <li>- Farbgestaltung</li> <li>- Schriftformatierung</li> <li>- Ausrichtung und Positionierung</li> <li>- Formatierung von Strukturelementen</li> <li>- Formatierung über eigene Stilklassen</li> </ul>	Farbbezeichnung oder Hexadezimalcode Schriftfamilien Box-Modell und Float  Grafiken, Tabellen, Hyperlinks, Listen etc.  Klassenselektoren
1.14 setzen Maßeinheiten zur Formatierung ein	<ul style="list-style-type: none"> <li>- absolut</li> <li>- relativ</li> </ul>	Im Hinblick auf unterschiedliche Ausgabegeräte sind relative Größen vorzuziehen.
1.15 erläutern die CSS-Vererbung und wenden fortgeschrittene CSS-Regeln an	<ul style="list-style-type: none"> <li>- Vererbungsregeln</li> <li>- Verschachtelung über Kind- und Elternelement</li> <li>- Pseudoklassen</li> </ul>	HTML-Dokumentenbaum  Zustand bei Hyperlinks Internetrecherche

**Lerngebiet 2:**

Zeitrichtwert : 20 Stunden

**Strukturierung von Webauftritten**

Lernziele	Lerninhalte	Hinweise zum Unterricht
Die Lernenden		
2.1 entwickeln Inhaltsbereiche einer Webseite	<ul style="list-style-type: none"> <li>- Kopf</li> <li>- Inhalt</li> <li>- Menü</li> <li>- Fuß</li> </ul>	Einteilung in Container über das <div>-TAG
2.2 gestalten die Webseitenstruktur über CSS	<ul style="list-style-type: none"> <li>- Höhe und Breite der Container</li> <li>- Abstände zwischen den Containern</li> <li>- Ausrichtung und Fluss</li> </ul>	Box-Modell und Float
2.3 erstellen Menüs und formatieren diese mit CSS	<ul style="list-style-type: none"> <li>- horizontal oder vertikal</li> <li>- Hover-Effekt</li> <li>- Dropdown-Effekt</li> </ul>	
2.4 binden Objekte in die Webstruktur ein	<ul style="list-style-type: none"> <li>- Einbetten von Webseiten in Container</li> <li>- Einbinden von Videos</li> </ul>	<object data>-TAG
2.5 verlinken strukturierte Webseiten	<ul style="list-style-type: none"> <li>- interne und externe Links</li> <li>- Hyperlinks bei Containern</li> </ul>	target = "top"
2.6 erläutern die Vorteile der Einbettung von Webinhalten	<ul style="list-style-type: none"> <li>- geringer Pflegeaufwand</li> <li>- Vermeidung von redundanter Datenhaltung</li> </ul>	

<b>Lerngebiet 3: Veröffentlichung von Webinhalten</b>		Zeitrichtwert : 10 Stunden
Lernziele	Lerninhalte	Hinweise zum Unterricht
Die Lernenden		
3.1 testen Websites und validieren HTML-Code	<ul style="list-style-type: none"> <li>- Hyperlinks</li> <li>- Browserkompatibilität</li> <li>- Auflösungen</li> <li>- Validierungstools</li> </ul>	
3.2 erläutern ein Client-Server Szenario im Internet	<ul style="list-style-type: none"> <li>- Client-Devices</li> <li>- Web-Server</li> <li>- Webspace</li> </ul>	PC, Tablet, Smartphone lokaler Webserver oder Provider
3.2 erläutern den Aufbau einer URL	<ul style="list-style-type: none"> <li>- Protokoll</li> <li>- Domäne</li> <li>- Dateiname</li> </ul>	Toplevel-Domain, DENIC
3.3 beschreiben die Datenkommunikation im Internet	<ul style="list-style-type: none"> <li>- Schichtenmodel TCP/IP</li> <li>- Protokolle der Schichten</li> <li>- Geräte zur Datenübermittlung</li> <li>- Übertragungsmedien</li> </ul>	IP, TCP, HTTP, FTP, SMTP, IMAP, SSL Netzwerkkarte (MAC), Repeater, Switch, Router Kabel, drahtlose Medien
3.4 speichern Webauftritte auf einen Webserver	<ul style="list-style-type: none"> <li>- FTP-Server</li> <li>- FTP-Client</li> <li>- Ports</li> </ul>	Upload
3.5 unterscheiden Internetdienste	<ul style="list-style-type: none"> <li>- WWW</li> <li>- E-Mail</li> <li>- Dateiverwaltung</li> <li>- Forum und Chat</li> <li>- Telefonie</li> <li>- Fernsehen</li> </ul>	Voice over IP Video-on-Demand

<b>Lerngebiet 4: Einsatz von Content-Management-Systemen</b>		Zeitrichtwert : 10 Stunden
Lernziele	Lerninhalte	Hinweise zum Unterricht
Die Lernenden		
4.1 installieren ein CMS-System	<ul style="list-style-type: none"> <li>- Webserver</li> <li>- Datenbankanbindung</li> <li>- Administratorkonto</li> </ul>	z. B. Joomla, WordPress
4.2 beschreiben Komponenten eines CMS	<ul style="list-style-type: none"> <li>- Frontend</li> <li>- Backend</li> <li>- Template</li> </ul>	CSS ermöglicht verteiltes Arbeiten an einer Website.
4.3 richten einen Webauftritt ein und gliedern diesen	<ul style="list-style-type: none"> <li>- Inhaltskomponenten</li> <li>- Aufbau von Webseiten</li> <li>- Editor</li> <li>- Menüverwaltung</li> <li>- Medien</li> </ul>	z. B. Artikel, Block, Bildgalerie
4.4 administrieren ein CMS	<ul style="list-style-type: none"> <li>- Ordnerstruktur</li> <li>- Benutzer und Benutzergruppen</li> <li>- Zugriffsrechte</li> <li>- geschützte Bereiche</li> <li>- Archivierung und Sicherung</li> <li>- Module</li> </ul>	
4.5 installieren Templates und passen diese an	<ul style="list-style-type: none"> <li>- Planung der Formate</li> <li>- Menüs</li> <li>- Bilder/Grafikformate</li> <li>- Farben</li> </ul>	Änderungen in der CSS-Datei des Templates

<b>Lerngebiet 5: Rechtliche und gesellschaftliche Aspekte im Internet</b>		Zeitrichtwert : 10 Stunden
Lernziele	Lerninhalte	Hinweise zum Unterricht
Die Lernenden		
5.1 beurteilen eine Veröffentlichung nach dem Urheberrecht	<ul style="list-style-type: none"> <li>- Nutzungsrechte des Urhebers</li> <li>- Tatbestand der Veröffentlichung</li> <li>- Recht am eigenen Bild</li> <li>- freie Bildquellen</li> <li>- richtiges Zitieren</li> <li>- Verstöße und Strafen</li> </ul>	<p>Urheberrechtsgesetz (UrhG), Schutzdauer, Panoramafreiheit, gemeinfreie Werke</p> <p>Kunsturhebergesetz (KuG) Creative Commons</p>
5.2 beschreiben wichtige Bestimmungen des Telemediengesetzes und wenden diese an	<ul style="list-style-type: none"> <li>- Impressumspflicht</li> <li>- Verwendung personenbezogener Daten</li> </ul>	Impressumgenerator
5.3 berücksichtigen bei der Webseitengestaltung die Barrierefreiheit	<ul style="list-style-type: none"> <li>- Seitentitel</li> <li>- Überschriften</li> <li>- Trennung von Layout und Inhalt</li> <li>- Navigation</li> </ul>	Online-Tools zur Überprüfung
5.4 erläutern Techniken zur Datensicherheit bei der Datenübertragung	<ul style="list-style-type: none"> <li>- sichere Passwörter</li> <li>- Verschlüsselung</li> <li>- Virensoftware und Firewall</li> <li>- Tunneling</li> </ul>	<p>HTTPS, Verschlüsselungstools, z. B. GPG4win</p> <p>VPN</p>
5.5 beschreiben gesellschaftliche Gefahren der Internetnutzung	<ul style="list-style-type: none"> <li>- Probleme in sozialen Netzwerken</li> <li>- Internetsucht</li> <li>- Internetkriminalität</li> </ul>	<p>Cyber-Mobbing</p> <p>Computerspiele Störerhaftung</p>

**Lerngebiet 6:** Zeitrichtwert: 38 Stunden  
**Grundlagen der Programmierung von Webanwendungen mit einer clientseitigen Skriptsprache**

Lernziele	Lerninhalte	Hinweise zum Unterricht
Die Lernenden		
6.1 erstellen Formulare zur Dialogsteuerung	<ul style="list-style-type: none"> <li>- Beschriftung</li> <li>- Eingabefeld</li> <li>- Optionsgruppe</li> <li>- Kontrollkästchen</li> <li>- Texteingabebereich</li> <li>- Schaltflächen</li> </ul>	Eigenschaften, Methoden
6.2 beschreiben den Aufbau einer Skript-Funktion	<ul style="list-style-type: none"> <li>- Funktionsname</li> <li>- Anweisungen (Statements)</li> <li>- Kommentare</li> </ul>	
6.3 binden ein Skript in HTML-Code ein	<ul style="list-style-type: none"> <li>- Skript im HTML Code</li> <li>- externes Skript</li> </ul>	Vorgabe kleiner Beispielanwendungen
6.4 geben wesentliche Merkmale eines Algorithmus an	<ul style="list-style-type: none"> <li>- Problemlösungsvorschrift</li> <li>- Ausführbarkeit</li> <li>- Eindeutigkeit</li> <li>- Endlichkeit</li> <li>- Allgemeingültigkeit</li> </ul>	JavaScript als Programmiersprache
6.5 wenden Datentypen an	<ul style="list-style-type: none"> <li>- Datentypen String Boolean Number Array Object</li> </ul>	JavaScript als schwach typisierte Sprache begreifen
6.6 unterscheiden Variablen und Konstanten	<ul style="list-style-type: none"> <li>- Deklaration von Variablen</li> <li>- Gültigkeit von Variablen lokal global</li> <li>- Konstanten</li> </ul>	Konventionen bei der Namensgebung
6.7 führen Rechen- operationen durch	<ul style="list-style-type: none"> <li>- Wertzuweisung</li> <li>- Operatoren arithmetische Operatoren Boolesche Operatoren Vergleichsoperatoren String-Operatoren Verkettungsoperatoren</li> </ul>	

**Lerngebiet 6:** Zeitrichtwert: 38 Stunden  
**Grundlagen der Programmierung von Webanwendungen mit einer clientseitigen Skriptsprache**

Lernziele	Lerninhalte	Hinweise zum Unterricht
Die Lernenden		
6.8 wenden Funktionen zur Schaffung abgeschlossener Prozeduren an	<ul style="list-style-type: none"> <li>- Parameterübergabe</li> <li>- Rückgabewert</li> </ul>	Event-Handler
6.9 begründen die Umwandlung von Datentypen	<ul style="list-style-type: none"> <li>- Umwandlung einer Zeichenkette in eine Zahl  <code>parseFloat()</code>  <code>parseInt()</code>  <code>Number()</code></li> <li>- Umwandlung einer Zahl in eine Zeichenkette  <code>toString()</code></li> </ul>	
6.10 modellieren Kontrollstrukturen und setzen sie in Programmcode um	<ul style="list-style-type: none"> <li>- Programmablaufplan</li> <li>- Sequenz</li> <li>- Auswahl (Selektion)</li> <li>- Wiederholung (Repetition)</li> </ul>	Modellierungstool  zweiseitige, mehrseitige Auswahl kopfgesteuerte Schleife Fußgesteuerte Schleife Zählschleife
6.11 deklarieren Arrays und wenden diese an	<ul style="list-style-type: none"> <li>- statische und dynamische Arrays</li> <li>- ein- und mehrdimensionale Arrays</li> <li>- Zugriffsmöglichkeiten auf Arrays</li> </ul>	
6.12 unterscheiden verschiedene Fehlertypen und realisieren Maßnahmen zur Fehlerbehandlung	<ul style="list-style-type: none"> <li>- Syntaxfehler</li> <li>- logische Fehler</li> <li>- Laufzeitfehler</li> </ul>	<code>window.onerror</code> <code>try ... catch</code> z. B. <code>firebug</code>

**Lerngebiet 7:** Zeitrichtwert: 20 Stunden  
**Objektorientiertes Programmieren von Webanwendungen mit einer clientseitigen Skriptsprache**

Lernziele	Lerninhalte	Hinweise zum Unterricht
Die Lernenden		
7.1 erläutern den Denkansatz der objektorientierten Programmierung	<ul style="list-style-type: none"> <li>- Abbildung der Zusammenhänge der realen Welt in Objekte</li> <li>- Objekten liegt ein Bauplan zugrunde</li> </ul>	
7.2 modellieren Ausschnitte der Wirklichkeit objektorientiert	<ul style="list-style-type: none"> <li>- Klassendiagramm</li> <li>- Objektdiagramm</li> <li>- Beziehungen</li> </ul>	UML-Tools z. B. Vererbung
7.3 codieren Klassen	<ul style="list-style-type: none"> <li>- Konstruktor</li> <li>- Attribute</li> <li>- Methoden</li> </ul>	In JavaScript erfolgt die Klassenbildung über Konstruktoren.
7.4 instanzieren Objekte einer Klasse und rufen Eigenschaften und Methoden auf	<ul style="list-style-type: none"> <li>- Aufruf des Konstruktors</li> <li>- Parameterübergabe</li> <li>- Zugriff auf Attribute</li> <li>- Rückgabewerte von Methoden</li> </ul>	
7.5 erläutern das Konzept der Kapselung und setzen es in Programmcode um	<ul style="list-style-type: none"> <li>- kontrollierter Zugriff</li> <li>- Sichtbarkeit von Attributen</li> <li>- Setter und Getter</li> </ul>	Zustand von Objekten wird über Schnittstellen verändert.
7.6 beschreiben den Mechanismus der Vererbung und setzen ihn in Programmcodes um	<ul style="list-style-type: none"> <li>- Vererbung von Attributen und Methoden</li> <li>- Überschreiben von Attributen und Methoden</li> </ul>	In JavaScript erfolgt die Vererbung über Prototyping.

**Lerngebiet 8:**

Zeitrichtwert: 24 Stunden

**Erstellen und Verwalten relationaler Datenbanken für Webauftritte**

Lernziele	Lerninhalte	Hinweise zum Unterricht
Die Lernenden		
8.1 unterscheiden Datenmodelle und physische Datenbank	<ul style="list-style-type: none"> <li>- konzeptuelles Datenmodell</li> <li>- logisches Datenmodell</li> <li>- physische Speicherung</li> </ul>	ERM/ER-D relationales Datenmodell SQL-Datenbank
8.2 modellieren Datenstrukturen konzeptuell	<ul style="list-style-type: none"> <li>- Entitytypen</li> <li>- Attribute</li> <li>- Schlüsselattribute</li> <li>- Beziehungen</li> <li>- Kardinalitäten</li> </ul>	
8.3 setzen das konzeptuelle Datenmodell in ein relationales Datenmodell um	<ul style="list-style-type: none"> <li>- Tabellen</li> <li>- Assoziationen</li> <li>- Datentypen</li> </ul>	
8.4 erläutern Datenbankanomalien und korrigieren fehlerhafte Datenstrukturen	<ul style="list-style-type: none"> <li>- Löschanomalie</li> <li>- Einfügeanomalie</li> <li>- Aktualisierungsanomalie</li> <li>- Normalisierungsprozess</li> </ul>	Inkonsistenz, Redundanz  1., 2. und 3. Normalform
8.5 erzeugen aus dem relationalen Datenmodell eine Datenbank für einen Webauftritt	<ul style="list-style-type: none"> <li>- Verbindung zum Webserver</li> <li>- SQL-Datenbank</li> <li>- Benutzerverwaltung</li> </ul>	Zugriffsrechte
8.6 verwalten Datenbanken mit einer Datenbanksprache	<ul style="list-style-type: none"> <li>- Datendefinition</li> <li>- Datenmanipulation</li> <li>- Datenabfrage</li> <li>- Datenzugriffsrechte</li> </ul>	SQL, z. B. CREATE TABLE, DROP VIEW z. B. INSERT, UPDATE SELECT, ORDER, auch Abfragen über mehrere Tabellen: Joins  z. B. GRANT, REVOKE

**Lerngebiet 9:**  
**Webbasiertes Auswerten von relationalen Datenbanken**

Zeitrichtwert: 38 Stunden

Lernziele	Lerninhalte	Hinweise zum Unterricht
Die Lernenden		
9.1 beschreiben die Entwicklungs-umgebung und die Funktionsweise dynamischer Websites	<ul style="list-style-type: none"> <li>- Webserver mit Datenbank-managementsystem</li> <li>- serverseitige Skriptsprache</li> <li>- Kommunikation zwischen Server und Client</li> </ul>	<p>z. B. Apache, PHP und MySQL  Ein Skript wird auf dem Webserver geparkt und erzeugt eine neue Webseite.</p>
9.2 erstellen serverseitige Skripte und binden Sie in HTML-Code ein	<ul style="list-style-type: none"> <li>- Notation der Einbindung</li> <li>- Anweisungen und Kommentare</li> <li>- Rechenoperationen</li> <li>- Operatoren</li> <li>- Variablen und Arrays</li> <li>- Kontrollstrukturen</li> <li>- Einbinden von Inhalten</li> </ul>	<p>include</p>
9.3 verarbeiten Formulardaten serverseitig	<ul style="list-style-type: none"> <li>- Verschicken von Daten</li> <li>- Auslesen und Verarbeiten von Formulardaten</li> <li>- Überprüfen von Formularvariablen</li> </ul>	<p>Unterschied zwischen post und get</p> <p>clientseitig, z. B. mit Java Script, und serverseitig, z. B. isset()</p>
9.4 arbeiten mit Funktionen	<ul style="list-style-type: none"> <li>- Grundsyntax</li> <li>- Argumente</li> <li>- Rückgabewert</li> </ul>	<p>vordefinierte und selbsterstellte Funktionen</p>
9.5 erstellen Skripte zur Speicherung von Daten und Datenauswertung einer serverseitigen Datenbank über Webseiten	<ul style="list-style-type: none"> <li>- Herstellen der Verbindung</li> <li>- Einbinden des SQL-Statements</li> <li>- Datenspeicherung</li> <li>- Auswertung über Abfragen</li> </ul>	<p>Daten als HTML-Tabelle ausgeben</p>

<b>Lerngebiet 10: Web-Projekte</b>		Zeitrichtwert : 40 Stunden
Lernziele	Lerninhalte	Hinweise zum Unterricht
Die Lernenden		
10.1 planen ein Webprojekt	<ul style="list-style-type: none"> <li>- Ziele</li> <li>- Adressaten</li> <li>- Inhalte</li> <li>- Ressourcen und Werkzeuge</li> </ul>	Lastenheft, Pflichtenheft
10.2 entwerfen ein Webprojekt	<ul style="list-style-type: none"> <li>- Abgrenzung der Inhalte</li> <li>- Layout</li> <li>- Verteilen der Aufgaben</li> </ul>	Teamarbeit
10.3 implementieren ein Webprojekt	<ul style="list-style-type: none"> <li>- statische Inhalte</li> <li>- dynamische Inhalte</li> <li>- Projekttest</li> </ul>	
10.4 erstellen eine Dokumentation	<ul style="list-style-type: none"> <li>- Navigation</li> <li>- Design</li> <li>- Namens- und Syntaxkonventionen</li> <li>- Aufbau der Datenbank</li> </ul>	Navigationsbaum, Screenshots Code-Beispiele  Datenbankmodell