

Fach INFORMATIK 2. Biennium (3. Klasse) Wirtschaftsfachoberschule Bruneck – Schwerpunkt: Wirtschaftsinformatik

Zielsetzungen, Kompetenzen am Ende der 5. Klasse

(Kompetent sein heißt, Wissen, Fertigkeiten und Kenntnisse zur Lösung von fremd- bzw. selbstgestellten Aufgaben nutzen zu können)

Der Schüler, die Schülerin kann:

- mit mathematischen und informatischen Werkzeugen ökonomische und soziale Ereignisse analysieren
- betriebliche Informationssysteme unter Berücksichtigung verschiedener Unternehmensformen interpretieren
- betriebliche Organisationsmodelle analysieren und situationsgerechte effiziente informationstechnische Lösungen ermitteln
- die Daten im Betrieb mit Informationssystemen verwalten
- die Vorgehensweisen und Werkzeuge der Planung und Betriebsführung anwenden und die Ergebnisse analysieren
- die geeigneten Vorgehensmodelle des Projektmanagements anwenden und situationsspezifische Anwendungen für unterschiedliche Marktstrategien erstellen
- die betrieblichen Informationssysteme und die Werkzeuge der internen Unternehmenskommunikation verwenden, um kontextbezogene kommunikative Aktivitäten zu realisieren
- technische Berichte verfassen und Arbeitsprozesse dokumentieren

Fertigkeiten	Kenntnisse	Methodik-Didaktik für das Lehren	Inhalte	Fächerübergreifende Verbindungen Medien-Materialeinsatz
<p>Lösungsverfahren durch Algorithmen ausdrücken und diese mit verschiedenen Programmierstilen und geeigneten Softwarewerkzeugen implementieren</p> <p>Gleichzeitig mit der Projektentwicklung eine Dokumentation erstellen</p>	<p>Programmiersprachen und Softwareentwicklungsmethoden</p> <p>Entwicklungsphasen eines Softwareprojektes</p>	<p>Praktische Arbeiten:</p> <ul style="list-style-type: none"> • Erstellung von Programmen für die Lösung von Aufgabenstellungen aus der Betriebs-Praxis, aus der Mathematik und aus BWL. Damit soll einerseits der Sprachumfang von Ruby und andererseits die Fähigkeit, Lösungsschritte in einer algorithmisierten Form zu formulieren, eingeübt werden. • Einige mögliche Programme: MwSt.-Berechnungen, Zins-Berechnungen, Lohnberechnung, Warenkalkulation und weitere Aufgaben aus dem Fach BWL. Lösen von quadratischen Gleichungen und linearen Gleichungssystemen, Sortieralgorithmen, Umwandeln in Zahlensystemen, Einübung von Schleifen, Simulation einer Würfelstatistik, graphische Darstellung von einfachen Funktionen, Matrizenoperationen und weitere Aufgaben aus dem Fach Mathematik. <p><u>Methodik-Didaktik-Pool für das Lehren/Lernen:</u></p> <ul style="list-style-type: none"> • Präsentationen/Referate/Frontalunterricht (durch Lehrer und/oder Schüler) • Fragend-entwickelnde Methode und Diskussionen (Lehrer/Schüler, Schüler/Schüler) • Entdeckendes experimentelles Arbeiten • Eigenverantwortliches Handeln fördern • Dokumentation von Arbeitsprozessen • Einzelarbeit (selbständige Erarbeitung ausgewählter Themen, selbständige Einübung von gelernten Methoden und Techniken) • Partnerarbeit und Gruppenarbeit (Erarbeitung ausgewählter Themen im Team, gemeinsame Einübung von gelernten Methoden und Techniken) kooperatives Handeln – respektvoller Umgang miteinander – Lernen aus eigenen und fremden Fehlern • E-Learning mit moodle • Offener Unterricht • Projektarbeit – Werkstattarbeit, Ideenaustausch 	<p>EINFÜHRUNG</p> <p>Was ist Informatik</p> <ul style="list-style-type: none"> • Technische Informatik, Praktische Informatik, Angewandte Informatik, Theoretische Informatik • Geschichtlicher Hintergrund der Informatik • Grober Aufbau eines Computers <ul style="list-style-type: none"> ○ Hard- und Software ○ Die CPU ○ RAM- und ROM-Speicher ○ Externer Speicher: Festplatten, CD-ROM, DVD-Laufwerke, Speichersticks <p>Information und Daten</p> <ul style="list-style-type: none"> • Bits, Bitfolgen, Hexziffern, Bytes und Worte, Datei- und Speichergrößen <p>Informationsdarstellung</p> <ul style="list-style-type: none"> • Text, ASCII-CODE, ASCII-Erweiterungen, Unicode, Zeichenketten • Logische Werte, AND- OR- und XOR • Zahlendarstellungen, Binärdarstellung, das Oktalsystem und das Hexadezimalsystem, • Umwandlung in das Binär-, Oktal- und Hexadezimalsystem, Arithmetische Operationen • Darstellung ganzer Zahlen, die Zweierkomplementdarstellung, Standardformate für die Darstellung ganzer Zahlen • Dezimalzahlendarstellung • Gleitpunktzahlen: Reelle Zahlen, Real-Zahlenbereiche in Programmiersprachen <p>EINFÜHRUNG IN DIE PROGRAMMIERUNG MIT RUBY</p> <p>Installation von Ruby und erste Schritte</p> <p>Das erste Programm – Aufbau eines</p>	<p>Beispiele aus:</p> <ul style="list-style-type: none"> • Mathematik • BWL <p>Medieneinsatz:</p> <ul style="list-style-type: none"> • Häufiges Arbeiten am Computer im PC-Raum • Einsatz von Folien • Präsentation mit Hilfe von Powerpoint • Einsatz der Lernplattform moodle • Didaktische Lernprogramme im Internet • Schultafel (wenn vorhanden auch interaktive Tafel) • Laptop-Klasse, wenn vorgesehen • Skripten, die von Fachlehrern der Schule erstellt wurden • Berichte aus Fachzeitschriften

Fertigkeiten	Kenntnisse	Methodik-Didaktik für das Lehren	Inhalte	Fächerübergreifende Verbindungen Medien-Materialeinsatz
		<ul style="list-style-type: none"> • Zusammenhang mit realitätsbezogenen Anwendungen • Einsatz von geeigneter Software • Einsatz von anderen Medien, je nach Verfügbarkeit und Sinnhaftigkeit (Tageslichtprojektor, interaktive Tafel, Filme) 	<p>Programmes</p> <ul style="list-style-type: none"> • Variablenzuweisungen • Ein- und Ausgabeanweisungen: PUTS, GETS • Steuerzeichen entfernen • Arithmetische Operatoren: +, -, *, /, % • Logische Operatoren: NOT, AND, OR • Stringoperatoren: + • Vergleichsoperatoren: ==, !=, <, >, <=, >= • Wertzuweisung: = • Kommentare • Fehlermeldungen und ihre Bedeutung <p>Einfache Datentypen</p> <ul style="list-style-type: none"> • INTEGER, STRING, FLOAT, BOOLEAN • Datentypen umwandeln <p>Strukturierte Datentypen:</p> <ul style="list-style-type: none"> • Ein- und zweidimensionale Arrays <p>Bedingte Anweisungen:</p> <ul style="list-style-type: none"> • IF...END • IF...ELSE...END • IF...ELSIF...END <p>Wiederholte Anweisungen:</p> <ul style="list-style-type: none"> • WHILE...END, BEGIN...END WHILE • UNTIL...END, BEGIN...END UNTIL <p>Zählschleifen:</p> <ul style="list-style-type: none"> • TIMES...END • FOR...IN...END <p>Menü-Auswahl:</p> <ul style="list-style-type: none"> • CASE...END <p>Methoden</p> <p>Objektorientierte Programmierung</p> <ul style="list-style-type: none"> • Objekte und Klassen • Eigenschaften (Instanzvariablen, Membervariablen, Attribute) einer Klasse • Methoden einer Klasse • Konstruktor einer Klasse • Vererbung • Überschreiben von Methoden <p>Aufteilen eines Programmes auf mehrere Dateien</p> <ul style="list-style-type: none"> • Schlüsselwort require • Verwendung von Klassen <p>Umgang mit Dateien</p> <ul style="list-style-type: none"> • Lesen und Schreiben von Textdateien 	

Fertigkeiten	Kenntnisse	Methodik-Didaktik für das Lehren	Inhalte	Fächerübergreifende Verbindungen Medien-Materialeinsatz
			<ul style="list-style-type: none"> • Speichern von Objekten mit Hilfe von Textdateien • Speichern von Objekten mittels Serialisierung <p>Grafische Oberflächen mit Ruby</p> <ul style="list-style-type: none"> • Installation der Ruby/GTK-Bindings • Erstes Fenster programmieren • Erste grafische Komponenten einfügen: Button, Ereignis clicked • Ruby/GTK-Klassendokumentation • Mehrere Komponenten ins Fenster einfügen: VBox, HBox, Table, Layout • Fenster schließen und Schließen verhindern • Farbverwaltung • Weitere Komponenten: Label, Image, EventBox, RadioButton, Button mit Icon, Tooltip, ScrollBar, Dialog, MessageDialog, FileSelectionDialog, ColorSelectionDialog, FontSelectionDialog, CheckButton und Frame, DrawingArea, Combo, Calender, ScrolledWindow und TextView, Menu, Kontextmenü • Das Signal key_press_event • ProgressBar • Signale blockieren und entfernen • Mauszeiger ändern • Zeichnen: Linie, Ellipse <p>GRUNDLAGEN VON JAVA</p> <p>Was ist Java</p> <ul style="list-style-type: none"> • Applikationen (application) • Applets <p>Software und Online-Dokumentation</p> <ul style="list-style-type: none"> • Java Development Kit (JDK) • Java Compiler (javac) • Java Runtime System (java) • Online-Dokumentation (API) • Software-Tools • Beispiel: Einfache HelloWorld-Applikation bzw. Applet <p>Beispiele für einfache Programme</p> <ul style="list-style-type: none"> • Programm mit Eingaben • Wertzuweisungen • If-Anweisung • for-Schleife • while-Schleife • switch 	

Fertigkeiten	Kenntnisse	Methodik-Didaktik für das Lehren	Inhalte	Fächerübergreifende Verbindungen Medien-Materialeinsatz
Die Entwicklungstendenzen und allgemeinen Merkmale aktueller Betriebssysteme beschreiben	Charakteristika der Betriebssysteme	<p>AUFBAU DES COMPUTERS</p> <ul style="list-style-type: none"> • Zusammenbau eines Computers aus den Hardwarekomponenten • Installation eines Betriebssystems • Konfiguration der Hardwarekomponenten (Treiber) • Installation von Anwendungssoftware • Umgang mit technischen Handbüchern und Fachzeitschriften <p>Computerwerkstatt</p> <ul style="list-style-type: none"> • Komponenten • Zusammenbau eines Rechners • periphere Geräte • Installation von Software (Betriebssysteme, Anwendungssoftware, usw.) 	<p>Allgemeines</p> <ul style="list-style-type: none"> • Aufbau und Funktionsweise eines Betriebssystems • Das BIOS • Booten eines Betriebssystems • Betriebssysteme im Vergleich: DOS, MS-Windows, Linux • Grundlagen zur parallelen Installation mehrerer Betriebssysteme auf einem Computer • Festplattenarchitektur • Partitionen • Dateisysteme • Partitionstabelle • Bootmanager • Installation des Linux-Grundsystems • Installation von Standardsoftware wie Office-Pakete, Internet-Browser mit Plugins, e-Mail-Clients • Entwicklungsumgebungen und Programmierwerkzeuge für Ruby und Java • Grundlegende Konsolenbefehle 	
Hypermedia zur Unterstützung der betrieblichen Kommunikation entwerfen	Strukturelemente von Hypermedia	<p>PRAKTISCHES ARBEITEN MIT DEM INTERNET</p> <ul style="list-style-type: none"> • Umgang mit Suchmaschinen • Nutzung des Internets zum Aktualisieren der Software, Treiber, usw. • Suchen im Internet nach bereitgestellten Quellcodes für verschiedene Probleme. • Aufbau einer E-Mail-Adresse • Zuweisung einer IP-Adresse im Netz 	<p>INTERNET UND WWW</p> <ul style="list-style-type: none"> • Entstehung des Internet Das ARPA-Net, wissenschaftliche Einrichtungen, das Netz der Netze • Basis-Standards im Internet Das TCP/IP-Protokoll, IP-Adressierung, Client-Server-Technologie, DNS – Domain Name Service, Routing und Gateways • Dienste im Internet E-Mail, Telnet und SSH, File Transfer (FTP), Chat (IRC), Newsgroups (News), World Wide Web (WWW) 	

Fach INFORMATIK **2. Biennium (4. Klasse) Wirtschaftsfachoberschule Bruneck – Schwerpunkt: Wirtschaftsinformatik**

Zielsetzungen, Kompetenzen am Ende der 5. Klasse

(Kompetent sein heißt, Wissen, Fertigkeiten und Kenntnisse zur Lösung von fremd- bzw. selbstgestellten Aufgaben nutzen zu können)

Der Schüler, die Schülerin kann:

- mit mathematischen und informatischen Werkzeugen ökonomische und soziale Ereignisse analysieren
- betriebliche Informationssysteme unter Berücksichtigung verschiedener Unternehmensformen interpretieren
- betriebliche Organisationsmodelle analysieren und situationsgerechte effiziente informationstechnische Lösungen ermitteln
- die Daten im Betrieb mit Informationssystemen verwalten
- die Vorgehensweisen und Werkzeuge der Planung und Betriebsführung anwenden und die Ergebnisse analysieren
- die geeigneten Vorgehensmodelle des Projektmanagements anwenden und situationsspezifische Anwendungen für unterschiedliche Marktstrategien erstellen
- die betrieblichen Informationssysteme und die Werkzeuge der internen Unternehmenskommunikation verwenden, um kontextbezogene kommunikative Aktivitäten zu realisieren
- technische Berichte verfassen und Arbeitsprozesse dokumentieren

Fertigkeiten	Kenntnisse	Methodik-Didaktik für das Lehren	Inhalte	Fächerübergreifende Verbindungen Medien-Materialeinsatz
<p>Lösungsverfahren durch Algorithmen ausdrücken und diese mit verschiedenen Programmierstilen und geeigneten Softwarewerkzeugen implementieren</p> <p>Gleichzeitig mit der Projektentwicklung eine Dokumentation erstellen</p>	<p>Programmiersprachen und Softwareentwicklungsmethoden</p> <p>Entwicklungsphasen eines Softwareprojektes</p>	<p>Praktische Arbeiten:</p> <ul style="list-style-type: none"> • Erstellung von Programmen für die Lösung von Aufgabenstellungen aus der Betriebs-Praxis, aus der Mathematik und aus BWL. Damit soll einerseits der Sprachumfang von Java und andererseits die Fähigkeit, Lösungsschritte in einer algorithmisierten Form zu formulieren, eingeübt werden. • Einige mögliche Programme: MwSt.-Berechnungen, Zins-Berechnungen, Lohnberechnung, Mitarbeiterverwaltung, Warenkalkulation und weitere Aufgaben aus dem Fach BWL. Lösen von quadratischen Gleichungen und linearen Gleichungssystemen, Sortieralgorithmen, Umwandeln in Zahlensystemen, Einübung von Schleifen, Simulation einer Würfelstatistik, graphische Darstellung von einfachen Funktionen, Matrizenoperationen und weitere Aufgaben aus dem Fach Mathematik. <p><u>Methodik-Didaktik-Pool für das Lehren/Lernen:</u></p> <ul style="list-style-type: none"> • Präsentationen/Referate/Frontalunterricht (durch Lehrer und/oder Schüler) • Fragend-entwickelnde Methode und Diskussionen (Lehrer/Schüler, Schüler/Schüler) • Entdeckendes experimentelles Arbeiten • Eigenverantwortliches Handeln fördern • Dokumentation von Arbeitsprozessen • Einzelarbeit (selbständige Erarbeitung ausgewählter Themen, selbständige Einübung von gelernten Methoden und Techniken) • Partnerarbeit und Gruppenarbeit (Erarbeitung ausgewählter Themen im Team, gemeinsame Einübung von gelernten Methoden und Techniken) kooperatives Handeln – respektvoller Umgang miteinander – Lernen aus eigenen und fremden Fehlern • E-Learning mit moodle • Offener Unterricht 	<p>PROGRAMMIERUNG IN JAVA</p> <p>Grundlagen der Java-Programmierung Variablen, primitive Datentypen, Referenz-Typen, Konstanten, Arrays, arithmetische Operatoren, Vergleichsoperatoren, logische Operatoren, Zuweisungsoperatoren, Einlesen von der Tastatur, Kontrollstrukturen (if – else, switch, for, while, do while), Methoden.</p> <p>Programmieren mit Objekten Klassen und Objekte, Konstruktoren, Overloading, Eigenschaften und Methoden, Klassen in Pakete auslagern, Kapselung, Vererbung, abstrakte Klassen, Interface.</p> <p>Grafikprogrammierung in AWT und SWING AWT und SWING, Grundlagen der Grafikausgabe, Container, Komponenten, Layout-Manager, Canvas, Event-Listener, Grafikmethoden, AWT-Komponenten, SWING-Komponenten, Ereignisbehandlung, Tabellen, Menüs, Aufbau einer GUI-Applikation.</p> <p>Ein- und Ausgabe von Daten Dateien und Ordner, Datenströme, Byteweise Datenströme, Zeichenweise Datenströme, andere Datenströme.</p>	<p>Beispiele aus:</p> <ul style="list-style-type: none"> • Mathematik • BWL • aus der Praxis <p>Medieneinsatz:</p> <ul style="list-style-type: none"> • Häufiges Arbeiten am Computer im PC-Raum • Einsatz von Folien • Präsentation mit Hilfe von Powerpoint • Einsatz der Lernplattform moodle • Didaktische Lernprogramme im Internet • Schultafel (wenn vorhanden auch interaktive Tafel) • Laptop-Klasse, wenn vorgesehen • Skripten, die von Fachlehrern der Schule erstellt wurden • Berichte aus Fachzeitschriften

Fertigkeiten	Kenntnisse	Methodik-Didaktik für das Lehren	Inhalte	Fächerübergreifende Verbindungen Medien-Materialeinsatz
		<ul style="list-style-type: none"> • Projektarbeit – Werkstattarbeit, Ideenaustausch • Zusammenhang mit realitätsbezogenen Anwendungen • Einsatz von geeigneter Software • Einsatz von anderen Medien, je nach Verfügbarkeit und Sinnhaftigkeit (Tageslichtprojektor, interaktive Tafel, Filme) 		
<p>Statische und dynamische Web-Seiten entwerfen und erstellen</p> <p>Grafiken und Videos erstellen, anpassen und optimieren.</p> <p>Web-Seiten im Internet publizieren.</p>	<p>Sprachen und Werkzeuge zur Implementierung von Web-Seiten</p> <p>Utility-Software für die Erstellung und Verwaltung von multimedialen Objekten.</p> <p>Struktur, Usability und Zugänglichkeit einer Web-Site.</p>	<p>Praktische Arbeiten:</p> <ul style="list-style-type: none"> • Erstellung einer eigenen Homepage, die dann auf dem Schulserver publiziert wird. • Diese Homepage wird als Grundlage für viele Übungen im Bereich Web-Entwicklung verwendet. • Bilder und Videos bearbeiten mit entsprechender Software für den Einsatz auf Webseiten. Videos sollen hauptsächlich verlinkt werden (z.B. Youtube). • Mit Hilfe von CSS-Styles eine Vorlage für eine Homepage erstellen • Mit Hilfe von JavaScript Formulare validieren und Fehlermeldungen clientseitig ausgeben. Beispiele zur Anzeige von Datum und Kalender. <p>• Eine Applikation in ein Applet umwandeln und auf verschiedenen Browsern testen.</p> <p>• Mit Hilfe von FTP die Homepage auf dem Webserver unter dem bereitgestellten Account publizieren.</p>	<p>HTML Grundlagen, Textformatierung, Hinter- und Vordergrundfarben, Verweise intern und extern, Bilder einfügen, Listen, Formulare, Tabellen, Frames</p> <p>Cascading Style-Sheets (CSS) - Formatvorlagen Aufbau, Anwendung, Formate für einzelne HTML-Tags, Formate zentral in der HTML-Datei, Formate extern in eigener Datei</p> <p>JavaScript und HTML Grundlagen, clientseitige Programmierung von Webseiten, Zugriffe auf das DOM-Objekt, einfache Beispiele, Möglichkeiten durch den Einsatz von JavaScript</p> <p>Java und HTML Erstellung eines Java-Applets, Integration eines Applets in eine Webseite, Applets und JAR-Dateien, Umwandlung einer Applikation in ein Applet</p> <p>FTP – File Transfer Protokoll Mit Hilfe von FTP die Webseiten auf einen Webserver kopieren</p> <p>Html5 Neuerungen, Verbesserungen, Ausblick</p> <p>PROGRAMMIERUNG VON DYNAMISCHEN WEBSEITEN MIT PHP</p> <p>Grundlagen Webserver, Apache, IIS, XAMPP, Installation</p> <p>Sprachreferenz Grundbefehle: echo, print, printf, Zuweisungen, Operatoren, Kommentare Variablen: Integer, Float, Double, Boolean, String, Array, Arrayfunktionen, Typkonvertierung, automatische Umwandlung, Konstanten Entscheidungsanweisungen: IF, ELSE, ELSEIF Schleifen: DO...WHILE, FOR, SWITCH-Anweisung Dateien einbinden: include, require, Funktionen: Variablenparameter, variable Argumentenliste</p>	

Fertigkeiten	Kenntnisse	Methodik-Didaktik für das Lehren	Inhalte	Fächerübergreifende Verbindungen Medien-Materialeinsatz
			Bearbeitung von Formularen Formulare, Werte übergeben, Formulare und JavaScript Datenbanken MySQL-Datenbank: mit Datenbank verbinden, einlesen, löschen, ändern von Daten, MySQL-Datenbankfunktionen.	
<p>Die Aufgaben eines Datenbankmanagementsystems erklären und dieses als Schnittstelle zwischen Benutzer und Daten einordnen.</p> <p>Datenbanken mit Bezug auf betriebliche Anforderungen entwerfen und erstellen</p> <p>Grundlegende Abfragen erstellen</p>	<p>Funktion eines Datenbankmanagementsystems</p> <p>Modellierung von Datenbanken</p> <p>Die Sprache SQL zur Datenbankabfrage</p>	<p>Praktische Arbeiten:</p> <ul style="list-style-type: none"> • Aufbau einer praktischen Datenbank Faktura mit ca. 2300 Kunden, ca. 10.000 Rechnungen, ca. 21.000 Rechnungszeilen und 77 Artikel. • Mit Hilfe dieser Datenbank werden umfangreiche SQL-Abfragen erstellt. • Andere Beispiele: Datenbank für eine Bibliothek und Beispiele bekannter Südtiroler Unternehmen wie z.B. Gasser, Senfter, Siebenföcher, Innerhofer und dergleichen. 	<p>DATENBANKSYSTEME</p> <p>Datenbanken und Datenbanksysteme</p> <p>Datenmodelle Datenbankprogrammierung, was ist eigentlich eine Datenbank, Beziehungen zwischen Tabellen – das relationale Modell, die Rolle der Schlüssel, Datenintegrität, lokale Datenbanken und Remote-Datenbanken, das Client/Server-Prinzip, allgemeine Überlegungen zum Datenbankdesign, Normalisierung von Tabellen, Verknüpfen der Tabellen, Sekundärindex.</p> <p>Die Abfragesprache SQL Überblick über SQL, Allgemeine Regeln zum Umgang mit SQL, Einfache Abfragen mit SELECT, Joins, Verknüpfung verschiedener Relationen, Einfügen, Ändern und Löschen von Datensätzen (INSERT, DELETE und UPDATE)</p>	

Fach INFORMATIK **5. Klasse** **Wirtschaftsfachoberschule Bruneck – Schwerpunkt: Wirtschaftsinformatik**

Zielsetzungen, Kompetenzen am Ende der 5. Klasse

(Kompetent sein heißt, Wissen, Fertigkeiten und Kenntnisse zur Lösung von fremd- bzw. selbstgestellten Aufgaben nutzen zu können)

Der Schüler, die Schülerin kann:

- mit mathematischen und informatischen Werkzeugen ökonomische und soziale Ereignisse analysieren
- betriebliche Informationssysteme unter Berücksichtigung verschiedener Unternehmensformen interpretieren
- betriebliche Organisationsmodelle analysieren und situationsgerechte effiziente informationstechnische Lösungen ermitteln
- die Daten im Betrieb mit Informationssystemen verwalten
- die Vorgehensweisen und Werkzeuge der Planung und Betriebsführung anwenden und die Ergebnisse analysieren
- die geeigneten Vorgehensmodelle des Projektmanagements anwenden und situationsspezifische Anwendungen für unterschiedliche Marktstrategien erstellen
- die betrieblichen Informationssysteme und die Werkzeuge der internen Unternehmenskommunikation verwenden, um kontextbezogene kommunikative Aktivitäten zu realisieren
- technische Berichte verfassen und Arbeitsprozesse dokumentieren

Fertigkeiten	Kenntnisse	Methodik-Didaktik für das Lehren	Inhalte	Fächerübergreifende Verbindungen Medien-Materialeinsatz
<p>Lösungsverfahren durch Algorithmen ausdrücken und diese mit verschiedenen Programmierstilen und geeigneten Softwarewerkzeugen implementieren</p> <p>Gleichzeitig mit der Projektentwicklung eine Dokumentation erstellen</p> <p>Innovative technologische Aspekte zur Verbesserung der betrieblichen Organisation ermitteln</p>	<p>Programmiersprachen und Softwareentwicklungsmethoden</p> <p>Entwicklungsphasen eines Softwareprojektes</p> <p>Das betriebliche Informationssystem</p>	<p>Praktische Arbeiten:</p> <ul style="list-style-type: none"> • Erstellung von Programmen für die Lösung von Aufgabenstellungen aus der Betriebs-Praxis, aus der Mathematik und aus BWL. Damit soll einerseits der Sprachumfang von Java und andererseits die Fähigkeit, Lösungsschritte in einer algorithmisierten Form zu formulieren, eingeübt werden. • Einige mögliche Programme: Berechnung der Quadratwurzel mit dem Verfahren von Newton, Fakultät, Fibonacci-Folge. Erstellung von Anwendungen für die Betriebspraxis: <ul style="list-style-type: none"> - Liste der Kunden (mit Blättern-Funktion) - Liste der Rechnungen - Umsätze der Kunden <p><u>Methodik-Didaktik-Pool für das Lehren/Lernen:</u></p> <ul style="list-style-type: none"> • Präsentationen/Referate/Frontalunterricht (durch Lehrer und/oder Schüler) • Fragend-entwickelnde Methode und Diskussionen (Lehrer/Schüler, Schüler/Schüler) • Entdeckendes experimentelles Arbeiten • Eigenverantwortliches Handeln fördern • Dokumentation von Arbeitsprozessen • Einzelarbeit (selbständige Erarbeitung ausgewählter Themen, selbständige Einübung von gelernten Methoden und Techniken) • Partnerarbeit und Gruppenarbeit (Erarbeitung ausgewählter Themen im Team, gemeinsame Einübung von gelernten Methoden und Techniken) kooperatives Handeln – respektvoller Umgang miteinander – Lernen aus eigenen und fremden Fehlern • E-Learning mit moodle • Offener Unterricht • Projektarbeit – Werkstattarbeit, Ideenaustausch • Zusammenhang mit realitätsbezogenen Anwendungen 	<p>PROGRAMMIERUNG IN JAVA</p> <p>Entwicklungsumgebung (Eclipse, NetBeans, oder ähnliche)</p> <ul style="list-style-type: none"> • Download und Installation der IDE • Aufbau und Erweiterung durch Plugins • Vorteile einer IDE in der SW-Entwicklung • Importieren und Ausführen von Programmen in der Entwicklungsumgebung • Dokumentation von Programmen mit Javadoc • Rekursive Programmieretechniken • Einsatz der Klasse BigInteger für beliebig große ganze Zahlen <p>Konzepte der Objektorientierten Programmierung (OOP)</p> <ul style="list-style-type: none"> • Wiederholung der Grundbegriffe Klasse, Objekt, Eigenschaft, getter- und setter-Methoden, Konstruktor • Erweiterungen: Vererbung, Kapselung, Overloading, Polymorphismus, Interface, Mehrfachvererbung • Einsatz eines UML-Plugins zum Erstellen von Class-Diagrammen • Vorteile der OOP bei der Programmierung <p>Grafikprogrammierung in AWT und SWING</p> <ul style="list-style-type: none"> • Unterschiede AWT und SWING • Komponenten von GUI-Oberflächen: Container, Button, Label, Textfelder, Check-Gruppen, Drop-Down, Radio-Buttons,... • Ereignisorientierte Programmierung • Anordnung der Komponenten durch Layout-Manager (Border, Grid, Flow, Null-Layout). • Kennenlernen eines GUI-Plugins zum Erstellen von grafischen Oberflächen <p>Datenbanken und Java</p> <ul style="list-style-type: none"> • Download und Installation des JDBC-Divers 	<p>Beispiele aus:</p> <ul style="list-style-type: none"> • Mathematik • BWL • aus der Praxis <p>Medieneinsatz:</p> <ul style="list-style-type: none"> • Häufiges Arbeiten am Computer im PC-Raum • Einsatz von Folien • Präsentation mit Hilfe von Powerpoint • Einsatz der Lernplattform moodle • Didaktische Lernprogramme im Internet • Schultafel (wenn vorhanden auch interaktive Tafel) • Laptop-Klasse, wenn vorgesehen • Skripten, die von Fachlehrern der Schule erstellt wurden • Berichte aus Fachzeitschriften

Fertigkeiten	Kenntnisse	Methodik-Didaktik für das Lehren	Inhalte	Fächerübergreifende Verbindungen Medien-Materialeinsatz
<p>Datenbanken mit Bezug auf betriebliche Anforderungen entwerfen und erstellen</p> <p>Grundlegende Abfragen erstellen</p>	<p>Modellierung von Datenbanken</p> <p>Die Sprache SQL zur Datenbankabfrage</p>	<ul style="list-style-type: none"> Einsatz von geeigneter Software Einsatz von anderen Medien, je nach Verfügbarkeit und Sinnhaftigkeit (Tageslichtprojektor, interaktive Tafel, Filme) <p>Praktische Arbeiten: Anhand von komplexen betrieblichen Aufgabenstellungen werden Datenmodelle graphisch entworfen (ER-Diagramme) und in eine redundanzfreie Datenstruktur umgesetzt. Geübt werden vor allem Maturaaufgaben der letzten Jahre.</p> <p>Laut den betrieblichen Aufgabenstellungen (siehe oben) werden komplexe SQL-Abfragen entworfen und umgesetzt. Dabei werden vor allem Abfragen über mehrere Tabellen geübt.</p>	<ul style="list-style-type: none"> Einsatz der Abfragesprache SQL Datenbankzugriffe in Java: Klassen Connection, Statement Methoden executeQuery() bzw. executeUpdate(). Datenabfragen mit SELECT Datenmanipulationen: INSERT, UPDATE, DELETE <p>Native App für Android</p> <ul style="list-style-type: none"> Als Projektarbeit konzipiert Einblick in die Java-Programmierung einer APP Konkrete Anwendung realisieren mit Zugriff auf eine Web-Datenbank <p>Relationale Datenbanken</p> <ul style="list-style-type: none"> Definitionen: Datenbank, DBMS, Redundanz, Datenintegrität, Sicherheit, Primär- und Fremdschlüssel Datenbanken modellieren mit dem ER-Modell Normalisierung: 1., 2. und 3. Normalform Abfragesprache SQL: Abfragen und Datenbankmanipulationen (DQL und DML) Transaktionen, Commit und Rollback Verwendung der Tools phpMyAdmin oder den MySQL-Query-Browser 	
<p>Die technischen Möglichkeiten des Internets nutzen und seine Entwicklungen abschätzen</p> <p>Im Internet Web-Seiten publizieren</p> <p>Die Kommunikation im Netz organisieren, um den Informationsfluss zu verbessern</p> <p>Remote-Datenbanken mit grafischer Web-Benutzeroberfläche zu betrieblichen Anforderungen implementieren</p>	<p>Netze für den Betrieb und für die öffentliche Verwaltung</p> <p>Datenbanken im Netz</p>	<p>Einsatz von Dreamweaver</p>	<p>HTML-Grundlagen</p> <ul style="list-style-type: none"> HTML5-und CSS3 Verwendung der neuen Möglichkeiten zur Erstellung moderner responsiver Webseiten Formulare, Werte übergeben, Formulare und JavaScript, neue Möglichkeiten mit HTML5 WebApps erstellen mit HTML5 und CSS3 für iPhone, iPad und Android <p>PHP und Datenbanken</p> <ul style="list-style-type: none"> MySQL-Datenbank: mit Datenbank verbinden, Einlesen, Löschen, Ändern von Daten, MySQL-Datenbankfunktionen. Einsatz des CMS-Systems Joomla <p>Web 2.0</p> <ul style="list-style-type: none"> Webseiten erweitern mit der AJAX-Technologie Dadurch können Teile einer Webseite mit Daten aus der Datenbank oder einem neuen StyleSheet verändert werden. Es muss nicht mehr dauernd die gesamte Webseite neu geladen werden. Einsatz des Frameworks jQuery 	

Fertigkeiten	Kenntnisse	Methodik-Didaktik für das Lehren	Inhalte	Fächerübergreifende Verbindungen Medien-Materialeinsatz
<p>Software zur Unterstützung betrieblicher Prozesse auswählen und verwenden.</p> <p>An Projekten zur optimalen Steuerung der Geschäftsprozesse mitarbeiten</p>	<p>Entwicklungstechniken für Projekte zur Integration der Geschäftsprozesse</p>		<p>Einführung in ERP-Systeme</p> <ul style="list-style-type: none"> • Definition • Beispiele • Verwendung im Betrieb 	
<p>Die rechtlichen Aspekte der Nutzung von Netzwerken erläutern, mit besonderer Aufmerksamkeit auf die Datensicherheit</p>	<p>Sicherheit in der Informationstechnik</p> <p>Schutz der Privatsphäre und des Urheberrechts</p> <p>Strafrechtliche Bestimmungen</p>		<p>Datenschutz und Datensicherheit</p> <ul style="list-style-type: none"> • Einteilung der Daten. • Was sind sensible Daten? • Wie können Daten geschützt werden? • Vorkehrungen im Betrieb, um dem Datenschutzgesetz Genüge zu leisten. 	<p>Fächerübergreifend mit Rechtskunde: Das Datenschutzgesetz</p>