Programm	= Automatisierung von Arbeitsabläufen funktioniert nach dem EVA-Prinzip: Eingabe -> Verarbeitung -> Ausgabe, d.h. Gewinnung von Ausgabedaten aus Eingabedaten
OOP	Programmcode ist in Objekten angeordnet, welche Mitglieder von Klassen sind. Eine Klasse beschreibt eine Kategorie von Objekten mit bestimmten Eigenschaften. Klassen können Eigenschaften weitergeben und verwandte Klassen erzeugen durch ein Verfahren namens Vererbung. Vorteil der OOP: Flexiblere und leistungsfähigere Programmentwicklung Gegenteil: prozedurale Sprachen -> Programme mit sequentiell abzuarbeitenden Anweisungen
Java allg.	Kennt keine Mehrfachvererbung! Java-Programm -> Java-Compiler -> Plattformunabhängiger Bytecode -> Interpreter VM Solaris, Interpreter VM Windows, Interpreter VM Linux etc. Jedes Objekt kümmert sich um seine Daten und gerät damit nicht in Konflikt mit den Methoden anderer Objekte. -> Kapselung
Klasse	Beschreibung von Daten und der Vorgehensweise, wie mit diesen Daten zu arbeiten ist.
Objekt	Ist ein bestimmter Satz von Daten, der nach der Beschreibung einer Klasse aufgebaut ist.
Eigenschaften	=Attribute (z.B. Name, Alter)
Methode	=Fähigkeiten (Jede Methode stellt eine Aufgabe dar, die das Objekt erfüllen kann, z.B. lesen, schreiben)
Swing	JButton JLabel JTextField JTextArea JTable
Swing-Container	JPanel JScrollPane JMenuBar JFileChooser
DataExpress	Database TableDataSet TextDataFile QueryDataSet
dbSwing	JdbLabel JdbTextField JdbList JdbComboBox
Interpreter Java	Dient dazu, kompilierte Java-Programme auszuführen, die als Bytecode in den .class-Dateien vorliegen.
Packages	gruppieren Klassen
Vererbung	ermöglicht Erzeugung einer Klassenhierarchie, deren oberstes Element die Klasse Object ist. Über Vererbung wird eine neue Klasse (Subklasse) auf Basis einer vorhandenen Klasse (Superklasse) erstellt. Die abgeleitete Klasse übernimmt dabei sämtliche Eigenschaften und Methoden der Superklasse und kann weitere hinzufügen. Übergeordnete Klassen = Superklassen; untergeordnete Klassen = Subklassen Eine Klasse hat höchstens eine Superklasse, kann aber beliebig viele Subklassen haben. Konstruktoren werden nicht vererbt, es ist jedoch möglich, den Konstruktor der Superklasse aufzurufen (super). Syntax: *class Rechenkalender extends Kalender*
Wrapper-Klassen	-> um Konvertierungsfunktionen nutzen zu können; Wrapper erlaubt es, Instanzen von Zahlen-Objekten zu bilden Void Byte Short Integer Long Float Double Boolean Charakter
Public	Eigenschaften/Methode ist in der Klasse selbst, in allen abgeleiteten Klassen und über die Instanz der Klasse sichtbar. -> Sichtbar für jeden In jeder Datei kann nur eine Klasse öffentlich sein, das ist genau die Klasse, nach der die Datei benannt ist.
Protected	Eigenschaften/Methode ist in der Klasse selbst, in allen abgeleiteten Klassen sichtbar. -> Sichtbar nur zum Zwecke der Vererbung in der gleichen Datei oder im gleichen Package
Private	Eigenschaften/Methode ist nur in der Klasse selbst sichtbar. -> Sichtbar für niemanden
Package Scoped	Standardsichtbarkeit
Überladen von Methoden	Gleiche Hierarchieebene; unterschiedliche Anzahl von Parametern
Überschreiben von Methoden	Unterschiedliche Hierarchieebenen; gleiche Parameterliste
Exception-Handling	Mit try-catch-Anweisung

Java-Klassendefinition	- Definition des Klassennamens - Deklaration von Klassenvaribalen und Eigenschaften - Konstruktoren: Klasse hat wenigstens einen Konstruktor - Methoden -> Inhalt der Klassendefinition wird durch geschweifte Klammern umschlossen.
Objekt-Definition	- Klasse definieren - Dem Objekt einen Namen zuordnen (Deklaration) - Speicherplatz für dieses Objekt im Computer reservieren (Instantiieren eines Objekts). Der Java-Code, der dafür benutzt wird heißt Konstruktor. Label (Klasse) helloLabel (Objekt) = new Label (Instantiierung des Objekts) ("Du schon wieder") (Parameter);
Methoden-Definition	- Festlegung der Geheimhaltungsstufe (falls abweichend vom Standard) - Angabe des Rückgabewertes einer Methode - Namen der Methode - Angabe der Parameter, die übergeben werden müssen, um die Aufgabe zu erfüllen
JVM (Java Virtual Machine)	Virtueller Prozessor, der den Java-Binärcode in der .class-Datei verwenden kann. Die beim Aufruf des Interpreters angegebene .class-Datei muss eine Methode public static void main (String [] args) besitzen, damit sie der Interpreter starten kann.
String	String ist kein elementarer Java-Datentyp sondern eine Klasse! Wird in String-Objekt gespeichert.
MediaTracker	Die Klasse MediaTracker überwacht das Laden eines oder mehrerer Bilder.
paint	Ausgabe in ein Grafikfenster erfolgt mit dieser Methode.
repaint	Zeichnet Fenster neu.
Arrays	Eindiemensionale + Mehrdimensionale Arrays z.B. char-Array, int-Array
Serialisierung	Umformung von Objekten, die sich im Arbeitspeicher befinden, in ein dateifähiges Format.
Destruktor	Hat geringe Bedeutung in Java, da Java über ein automatisches Speichermanagement (Garbage Collector) verfügt. Syntax: *Protected void finalize()*
final-Attribute	Eigenschaft, die nicht verändert werden darf (Konstante). Z.B. final double MWSt = 0.16;
final-Methode	Darf nicht verändert werden, das heißt ein Überschreiben ist nicht möglich.
final-Klasse	Haben keine Subklassen! z.B. java.lang.math, java.lang.string
IDE	Integrierte Entwicklungsumgebung (JBuilder, Visual Café, Eclipse, Netbeans)
java.txt.*	Stellt Formatklassen zur Darstellung diverser Datentypen zur Verfügung.
Logische Operatoren **Vergleichsoperatoren** **Verknüpfungsoperator** **Inkrementoperator** **Dekrementoperator** **Zuweisungsoperator** **Operatoren**	&&; ||; ^ (ausschließendes oder) >; >=; <; <=; ==; != + (zur Konkatenation zweier Zeichenketten) ++ -- x=x+1 +; -; /; %; *
Escape-Code-Literale (Steuerzeichen)	Sind nicht direkt druckbar oder durch die Tastatur darstellbar. Z.B. \n für neue Zeile, \t für Tabulator
double-Werte	Verfügen nicht über eigene Methoden. Hilfskonstrukt: Wrapper-Klasse Double
Postfix-Inkrementierung	y = x++;
Präfix-Inkrementierung	z = ++x;
Clipping Region	Damit kann im Grafikkontext die Ausgabe auf einen bestimmten Bereich eingeschränkt werden
Listener	Registriert konkretes Ereignis (z.B. Mausklick) und ruft Ereignisprozedur auf.
AWT	Abstract Window Toolkit -> Originäre Grafikbibliothek von Java. Heute gibt es bereits die Swing-Grafikbibliothek. Syntax: *import java.awt.*;*
Bitmap-Behandlung	Klasse Toolkit; Methode getImage -> Bitmap wird geladen Klasse Graphics; Methode drawImage -> Bitmap wird angezeigt
Grundschema der Variablendeklaration	Datentyp Variablenname;
This-Zeiger	Referenz auf Fenster-Objekt
break	Komplette Kontrollstruktur wird abgebrochen
continue	Sprung zum Schleifenkopf, d.h. laufende Iteration wird abgebrochen, die Kontrollstruktur aber mit der nächsten Iteration fortgesetzt
Java-Anwendung	Stand-Alone-Anwendung, braucht JVM, wird durch Aufruf ihrer main()-Methode gestartet. Java-Applikation braucht eine Hauptklasse, die mit der Hardware und dem Java-Bytecode-Interpreter kommuniziert.
Java-Applet	braucht HTML-Browser oder Applet-Viewer, Webbrowser erzeugt Instanz der Applet-Klasse und ruft die Methoden init() und start() auf Alles was ein Applet bei seiner Initialisierung - beim Laden von einem Web-Server - ausführen soll, wird über init definiert. Applets haben keinen Zugriff auf lokale Dateien des Browser-Rechners; Ausführung externer Programme nicht gestattet (es sei denn zertifiziertes Applet). Applets funktionieren AWT-basiert, d.h. ereignisorientiert unter Einbeziehung einer GUI im Gegensatz zu Java-Applikationen, die auch als textorientierte Konsolenprogramme laufen können

Threads	Threads sind eigenständige Programmteile, die paralell zu anderen Threads laufen können. In Java durch Klasse Thread und Interface Runnable zur Verfügung gestellt.
Interface Runnable	Ermöglicht die Implementierung von Threads in einer Klasse. Erzeugung des Konstruktors Methode start: Thread anstoßen Methode run: Darstellung des Thread-Inhalts
Rechnerplattform	Kombination aus Prozessor und Betriebssystem
***.jar**	Steht für JavaArchive. Dieses Archiv fasst beliebige Dateien in einer Datei zusammen.
JRE	Java Runtime Environment, enthält die Programme und Bibliotheken des JDK, die zum Ausführen von Java-Programmen notwendig sind.
JavaBeans	Software-Komponenten, um beispielsweise grafische Oberflächen zu entwickeln.
Exceptions (Ausnahmen)	Auslöser: Laufzeitfehler im Programm oder innerhalb der JVM, aber auch Programmierer kann Exceptions auslösen. Compiler erwartet an bestimmter Stelle im Programm, dass ein Exceptionhandling (Ausnahmebehandlung) durchgeführt wird. Das Auslösen einer Exception wird throw (werfen) genannt. Exceptions werden immer innerhalb einer Methode ausgelöst. Behandeln einer Exception wird catch (auffangen) genannt.
CLASSPATH	Java sucht Klassendateien im Klassenpfad
GET	statisch: Get-Parameter werden an einen Link angehängt dynamisch: Auswertung der Input-Felder des HTML-Formulars
JSP	Vorteile: Programmiersprache ist Java, Zugriffe auf alle Java-Klassen und JavaBeans
Konstruktor	Konkrete Objekte werden instanziiert (von der Klasse abgeleitet) durch spezielle Methoden: Konstruktoren
Klassenvariablen	Deklaration mit Modifier static. Auf diese Variablen kann direkt über den jeweiligen Klassennamen zugegriffen werden. Variable existiert für alle Objeke einer Klasse genau ein Mal.
Instanzvariablen	Werden für jedes Objekt einer Klasse angelegt und von Java automatisch initialisiert.
lokale Variablen	Befinden sich innerhalb einer Methode und sind nur in dieser sichtbar. Lokale Variablen werden von Java nicht automatisch initialisiert und müssen deswegen vor ihrer Verwendung initialisiert werden.
Kontrollstrukturen	1. Fallunterscheidung / Auswahl if...then...else 2. Mehrfachauswahl switch 3. Wiederholung / Schleife Zählschleife (Kopf- oder Fußgesteuert) for-Schleife (wenn bestimmte Zahl von Wiederholungen benötigt wird) whilte-Schleife (wenn etwas solange wiederholt werden soll, bis Bedingung erfüllt ist) 4. Sequenz -> einfache Aufeinanderfolge von Befehlen
Java-Servlet	Java-Programm, dass auf Server ausgeführt wird.
JSP	Erweiterung der Servlet-Technologie, die entwickelt wurde, um das Erstellen von HTML- und XML-Seiten zu vereinfachen. Ermöglicht Dynamik. In JSP keine Geschäftslogik, da alles in Bean verlagert -> bessere Trennung als im Servlet! Datenbankabfragen mit try-catch sind in JSP nicht nötig
getProperty	damit wird auf Beans zugegriffen
Schnittstellen	CGI, Java-Servlets, SSI (Server Side Includes, z.B. ASP, JSP) Alternative zu CGI: APIs (z.B. ISAPI von Microsoft)
Webanfrage	Browser -> Request -> Server -> CGI -> Response an Server get (Request hängt an URL) post (Request im Body-Teil) put (Dateien an Server senden) delete (Dateien vom Server löschen) trace (Kopie des Requests liefern) options (Infos zu Serverobjekten)
Typ1-Verbindung JDBC/ODBC-Bridge	import java.sql.* 3 Objekte notwendig: 1. Connection (Verbindung zur Datenquelle) 2. Statement (zur Ausführung von SQL-Befehlen) 3. ResultSet (Ergebnistabelle) ODBC-Treiber einrichten
Verteilte Webanwendung (Schichtenarchitektur/ Three-Tier-Application)	1. Präsentationsschicht (Client / Webbrowser) 2. Steuerungsschicht (Webserver, Servlet, Java-Anwendung) 3. Datenschicht (Datenbankserver, relationale DB)
MVC-Modell (Model-View-Controller-Modell)	Trennung eines Programms in die drei Einheiten Geschäftslogik (M), Präsentation (V) und Interaktion (C) mit dem Ziel Flexibilität und Wiederverwendbarkeit zu erhöhen. Spätere Änderungen und Erweiterungen sind einfach zu halten. Das Modell sorgt für Übersicht und Ordnung und ermöglicht sogar eine Rollenverteilung. z.B. erstellt Web-Designer das Erscheinungsbild, Programmierer die Geschäftslogik und Datenbankexperte kümmert sich um optimale Datenverwaltung.
Session-Management	getCookie, setCookie

Kommentare	Informationen, die für den Leser des Java-Codes und nicht für den Computer bestimmt sind, stehen zwischen den Zeichen /* (oder/**) und */.
Panel	Fasst andere Objekte wie in einem Container zusammen. Kann Buttons oder auch andere Steuerelemente der GUI enthalten.
Elementare Datentypen	*Ganze Zahlen* - byte (8 Bit) - short (16 Bit) - int (32 Bit) - long (64 Bit) *Dezimalzahlen* - float (32 Bit) - double (64 Bit)