



دانشگاه پیام نور

برنامه نویسی وب

دکتر عباسعلی رضائی دکتر مصطفی اخوان صفار

امروز کتابخوانی و علم‌آموزی نه تنها یک وظیفه‌ی ملی، که یک واجب دینی است.^۱

مقام معظم رهبری

در عصر حاضر یکی از شاخصه‌های ارزیابی رشد، توسعه و پیشرفت فرهنگی هر کشوری میزان تولید کتاب، مطالعه و کتابخوانی مردم آن مرز و بوم است. ایران اسلامی نیز از دیرباز تاکنون با داشتن تمدنی چندهزارساله و مراکز متعدد علمی، فرهنگی، کتابخانه‌های معتبر، علما و دانشمندان بزرگ با آثار ارزشمند تاریخی، سرآمد دولت‌ها و ملت‌های دیگر بوده و در عرصه فرهنگ و تمدن جهانی به‌سان خورشیدی تابناک همچنان می‌درخشد و با فرزندان نیک‌نهاد خویش هنرنمایی می‌کند. چه کسی است که در دنیا با دانشمندان فرزانه و نام‌آور ایرانی همچون ابوعلی سینا، ابوریحان بیرونی، فارابی، خوارزمی و ... همچنین شاعران برجسته‌ای نظیر فردوسی، سعدی، مولوی، حافظ و ... آشنا نباشد و در مقابل عظمت آنها سر تعظیم فرود نیاورد. تمامی این افتخارات ارزشمند، برگرفته از میزان عشق و علاقه فراوان ملت ما به فراگیری علم و دانش از طریق خواندن و مطالعه منابع و کتاب‌های گوناگون است. به شکرانه الهی، تاریخ و گذشته ما، همیشه درخشان و پر بار است. ولی اکنون در این زمینه در چه جایگاهی قرار داریم؟ آمار و ارقام ارائه‌شده از سوی مجامع و سازمان‌های فرهنگی در مورد سرانه مطالعه هر ایرانی، برایمان چندان امیدوارکننده نمی‌باشد.

کتاب، دروازه‌ای به سوی گستره دانش و معرفت است و کتاب خوب، یکی از بهترین ابزارهای کمال بشری است. همه دستاوردهای بشر در سراسر عمر جهان، تا آنجا که قابل کتابت بوده است، در میان دست‌نوشته‌هایی است که انسان‌ها پدید آورده و می‌آورند. در این مجموعه بی‌نظیر، تعالیم الهی، درس‌های پیامبران به بشر، و همچنین علوم مختلفی است که سعادت بشر بدون آگاهی از آنها امکان‌پذیر نیست. کسی که با دنیای زیبا و زندگی‌بخش کتاب ارتباط ندارد بی‌شک از مهم‌ترین دستاورد انسانی و نیز از بیشترین معارف الهی و بشری محروم است. با این دیدگاه، به‌روشنی می‌توان ارزش و مفهوم رمزی عمیق در این حقیقت تاریخی را دریافت که اولین خطاب خداوند متعال به پیامبر گرامی اسلام (ص) این است که «بخوان!» و در اولین سوره‌ای که بر آن فرستاده عظیم‌الشان خداوند، فرود آمده، نام «قلم» به تجلیل یاد

1. <https://farsi.khamenei.ir/message-content?id=2696>

شده است: «إِقْرَأْ وَ رَبُّكَ الْأَكْرَمُ. الَّذِي عَلَّمَ بِالْقَلَمِ» در اهمیت عنصر کتاب برای تکامل جامعه انسانی، همین بس که تمامی ادیان آسمانی و رجال بزرگ تاریخ بشری، از طریق کتاب جاودانه مانده اند.

دانشگاه پیام نور با گستره جغرافیایی ایران شمول خود با هدف آموزش برای همه، همه جا و همه وقت، به عنوان دانشگاهی کتاب محور در نظام آموزش عالی کشورمان، افتخار دارد جایگاه اندیشه سازی و خردورزی بخش عظیمی از جوانان جویای علم این مرز و بوم باشد. تلاش فراوانی در ایام طولانی فعالیت این دانشگاه انجام پذیرفته تا با بهره گیری از تجربه های گرانقدر استادان و صاحب نظران برجسته کشورمان، کتاب ها و منابع آموزشی درسی شاخص و خودآموز تولید شود. در آینده هم، این مهم با هدف ارتقای سطح علمی، روزآمدی و توجه بیشتر به نیازهای مخاطبان دانشگاه پیام نور با جدیت ادامه خواهد داشت. به طور قطع استفاده از نظرات استادان، صاحب نظران و دانشجویان محترم، ما را در انجام این وظیفه مهم و خطیر یاری رسان خواهد بود. پیشاپیش از تمامی عزیزانی که با نقد، تصحیح و پیشنهادهای خود ما را در انجام این وظیفه خطیر یاری می رسانند، سپاسگزاری می نمایم. لازم است از تمامی اندیشمندانی که تاکنون دانشگاه پیام نور را منزلگاه اندیشه سازی خود دانسته و ما را در تولید کتاب و محتوای آموزشی درسی یاری نموده اند، صمیمانه قدردانی گردد. موفقیت و بهروزی تمامی دانشجویان و دانش پژوهان عزیز آرزوی همیشگی ما است.

دانشگاه پیام نور

فهرست مطالب

سیزده

پیشگفتار

۱	فصل اول: برنامه‌نویسی HTML
۱	هدف کلی
۱	هدف‌های یادگیری
۱	مقدمه
۲	۱-۱ تقسیم‌بندی زبان‌های برنامه‌نویسی
۷	۲-۱ تقسیم‌بندی زبان‌های برنامه‌نویسی تحت وب
۸	۳-۱ زبان HTML
۹	۴-۱ تگ‌های HTML
۱۰	۱-۴-۱ تگ‌های Head
۱۲	۲-۴-۱ تگ‌های Body
۱۴	۳-۴-۱ ایجاد لیست‌ها
۱۶	۴-۴-۱ ایجاد ابر پیوند
۱۷	۵-۴-۱ درج عکس
۱۷	۶-۴-۱ نحوه ایجاد جدول
۱۸	۷-۴-۱ ایجاد فرم
۲۰	۸-۴-۱ شیوه‌نامه (CSS)
۲۳	۹-۴-۱ فارسی‌نویسی
۲۴	۵-۱ جدول راهنما
۲۸	۶-۱ آشنایی با چهارچوب نرم‌افزاری بوت استرپ
۲۹	۱-۶-۱ شروع سریع کار با بوت استرپ
۳۰	۲-۶-۱ اضافه‌کردن به کد CSS
۳۰	۳-۶-۱ اضافه‌کردن به JS

۳۰	۱-۶-۴ قالب شروع کننده
۳۱	۱-۶-۵ تنظیمات مهم جهانی
۳۳	خلاصه فصل اول
۳۴	خودآزمایی چهارگزینه‌ای فصل اول
۳۵	خودآزمایی تشریحی فصل اول
۳۷	فصل دوم: زبان جاوا اسکریپت
۳۷	هدف کلی
۳۷	هدف‌های یادگیری
۳۷	مقدمه
۳۸	۲-۱ جاوا اسکریپت در برابر جاوا
۳۹	۲-۲ تگ جاوا اسکریپت
۴۰	۲-۳ خطاها
۴۱	۲-۴ مکان نوشتن کدها
۴۳	۲-۵ متغیرها
۴۴	۲-۵-۱ انواع متغیرها
۴۴	۲-۶ عملگرها
۴۷	۲-۷ اشیاء در جاوا اسکریپت
۵۶	۲-۸ اشیای مربوط به HTML DOM
۵۷	۲-۹ دستورهای شرطی
۵۸	۲-۱۰ حلقه‌ها
۶۰	۲-۱۱ آرایه‌ها
۶۲	۲-۱۲ توابع
۶۴	۲-۱۳ پنجره‌های ارتباط باکار بر
۶۵	۲-۱۴ چند نکته در اسکریپت‌نویسی
۶۶	۲-۱۵ دو مسئله حل شده
۶۷	۲-۱۶ رویدادها در جاوا اسکریپت
۸۲	۲-۱۷ تغییر محتوای متن در جاوا اسکریپت
۸۳	۲-۱۸ آشنایی با کتابخانه جی کوئری (jQuery)
۸۴	۲-۱۹ اضافه کردن جی کوئری به صفحات وب
۸۵	خلاصه فصل دوم
۸۶	خودآزمایی چهارگزینه‌ای فصل دوم
۸۸	خودآزمایی تشریحی فصل دوم
۸۹	فصل سوم: XML
۸۹	هدف کلی
۸۹	هدف‌های یادگیری

۸۹	مقدمه
۹۰	۳-۱ ابزارهای XML
۹۱	۳-۱-۱ قواعد نگ نویسی در XML
۹۱	۳-۲ سیستم تعریف نوع سند DTD
۹۳	۳-۳ روش XML Schema
۹۹	۳-۴ زبان تبدیل XML : XSLT
۱۰۲	۳-۵ تنظیم CSS
۱۰۴	۳-۶ XLink و XPointer
۱۰۵	۳-۷ مقدمه‌ای بر JSON
۱۰۵	۳-۷-۱ چرا از JSON استفاده کنیم
۱۰۷	۳-۷-۲ XML و JSON
۱۰۸	۳-۷-۳ HTML و JSON
۱۰۹	۳-۷-۴ PHP و JSON
۱۰۹	خلاصه فصل سوم
۱۱۰	خودآزمایی چهارگزینه‌ای فصل سوم
۱۱۱	خودآزمایی تشریحی فصل سوم
۱۱۳	فصل چهارم: Ajax
۱۱۳	هدف کلی
۱۱۳	هدف‌های یادگیری
۱۱۳	مقدمه
۱۱۴	۴-۱ تاریخچه
۱۱۷	۴-۲ مزایا و معایب Ajax
۱۲۲	۴-۳ شیء XMLHttpRequest
۱۲۴	۴-۴ افزودن جاوا اسکریپت
۱۲۵	۴-۵ نازک‌کاری با DOM
۱۳۴	خلاصه فصل چهارم
۱۳۴	خودآزمایی چهارگزینه‌ای فصل چهارم
۱۳۵	خودآزمایی تشریحی فصل چهارم
۱۳۷	فصل پنجم: زبان برنامه‌نویسی PHP 7
۱۳۷	هدف کلی
۱۳۷	هدف‌های یادگیری
۱۳۷	مقدمه
۱۴۰	۵-۱ برنامه‌نویسی بک‌اند (Backend) و فرانت‌اند (Frontend)
۱۴۱	۵-۲ نحوه نصب PHP 7
۱۵۰	۵-۳ ساختار کدها در PHP

۱۵۱	۴-۵ توضیحات در PHP
۱۵۳	۵-۵ متغیرها در php 7
۱۵۴	۶-۵ چاپ متغیرها در خروجی با دستور print و echo
۱۵۶	۷-۵ متغیرهای عمومی و محلی در زبان PHP
۱۵۸	۸-۵ انواع داده در php
۱۶۰	۹-۵ آرایه‌ها در PHP
۱۶۲	۱۰-۵ رشته‌های عددی در PHP
۱۶۳	۱۰-۵-۱ تبدیل رشته و عدد اعشاری به نوع صحیح در PHP
۱۶۳	۱۱-۵ توابع ریاضی در PHP
۱۶۵	۱۲-۵ ثابت‌ها در PHP
۱۶۶	۱۳-۵ آرایه‌های ثابت در PHP
۱۶۷	۱۴-۵ توابع رشته‌ها در PHP
۱۶۹	۱۵-۵ انواع عملگرها در PHP
۱۶۹	۱۵-۵-۱ عملگرهای محاسباتی
۱۶۹	۱۵-۵-۲ عملگرهای افزایشی / کاهششی
۱۷۰	۱۵-۵-۳ عملگرهای مقایسه‌ای
۱۷۰	۱۵-۵-۴ عملگرهای منطقی
۱۷۱	۱۵-۵-۵ عملگرهای رشته‌ای
۱۷۲	۱۵-۵-۶ عملگرهای آرایه‌ای
۱۷۳	۱۵-۵-۷ عملگرهای انتساب شرط
۱۷۳	۱۶-۵ دستورات شرطی در PHP
۱۷۶	۱۷-۵ حلقه‌ها در php
۱۸۰	۱۸-۵ توابع در PHP 7
۱۸۱	۱۸-۵-۱ آرگومان‌های تابع
۱۸۳	۱۸-۵-۲ تعیین مقدار پیش فرض برای آرگومان‌ها
۱۸۴	۱۸-۵-۳ برگرداندن مقادیر
۱۸۴	۱۸-۵-۴ تعیین نوع داده برگشتی
۱۸۵	۱۸-۵-۵ پاس دادن آرگومان به روش ارجاع
۱۸۶	۱۹-۵ انواع آرایه‌ها در زبان PHP
۱۸۹	۱۹-۵-۱ مرتب‌سازی آرایه‌ها در PHP
۱۹۰	۲۰-۵ متغیرهای فوق سراسری (سوپر گلوبال) در PHP
۱۹۴	۲۱-۵ برگه‌ها در PHP
۱۹۵	۲۱-۵-۱ فرق GET و POST
۱۹۶	۲۱-۵-۲ اعتبارسنجی فرم در PHP
۱۹۷	۲۱-۵-۳ دکمه‌های رادیویی
۱۹۹	۲۱-۵-۴ اعتبارسنجی داده‌های فرم با PHP

۲۰۰	۵-۲۱-۵ الزامی کردن فیلدهای فرم
۲۰۲	۵-۲۱-۶ نمایش پیغام‌های خطا
۲۰۶	۵-۲۲ اتصال به پایگاه داده در PHP
۲۰۹	۵-۲۳ ساخت پایگاه داده و جدول در PHP
۲۲۵	خلاصه فصل پنجم
۲۲۵	خودآزمایی چهارگزینه‌ای فصل پنجم
۲۲۸	خودآزمایی تشریحی فصل پنجم
۲۲۹	فصل ششم: برنامه‌نویسی وب با پایتون
۲۲۹	هدف کلی
۲۲۹	هدف‌های یادگیری
۲۲۹	مقدمه
۲۳۰	۶-۱ زبان پایتون و وب
۲۳۱	۶-۲ آشنایی با مفهوم پلتفرم توسعه تحت وب
۲۳۲	۶-۳ معرفی پلتفرم‌های برنامه‌نویسی وب با پایتون
۲۳۳	۶-۳-۱ فریم‌ورک‌های Full Stack زبان پایتون برای توسعه وب
۲۳۶	۶-۳-۲ فریم‌ورک‌های غیر فول استک زبان پایتون
۲۳۸	۶-۴ نصب پایتون در ویندوز
۲۴۲	۶-۵ تعریف متغیرها در پایتون
۲۴۷	۶-۶ انواع داده‌ها در پایتون
۲۴۷	۶-۶-۱ نوع داده Built-in
۲۴۸	۶-۶-۲ نوع‌های عددی در پایتون
۲۴۹	۶-۶-۳ تبدیل نوع در پایتون
۲۵۰	۶-۶-۴ اعداد تصادفی در پایتون
۲۵۲	۶-۷ رشته‌ها در پایتون
۲۵۴	۶-۷-۱ برخی توابع موجود در رشته‌ها
۲۵۷	۶-۸ عملگرها در پایتون
۲۶۰	۶-۹ لیست‌های در پایتون
۲۷۰	۶-۱۰ تاپل‌ها در پایتون
۲۷۶	۶-۱۱ مجموعه‌ها در پایتون
۲۸۱	۶-۱۲ دیکشنری در پایتون
۲۸۷	۶-۱۳ دیکشنری‌های تودرتو
۲۸۸	۶-۱۴ عبارات شرطی در پایتون
۲۹۰	۶-۱۴-۱ دستورات شرطی تودرتو
۲۹۱	۶-۱۵ حلقه‌ها در پایتون
۲۹۵	۶-۱۵-۱ حلقه تودرتو در پایتون
۲۹۶	۶-۱۶ توابع در پایتون

۲۹۸	۶-۱۶-۱ تابع بازگشتی در پایتون
۳۰۰	۶-۱۶-۲ تابع لامبدا در پایتون
۳۰۱	۶-۱۷ طراحی وب با Django
۳۰۱	۶-۱۷-۱ نصب پایگاه داده
۳۰۱	۶-۱۷-۲ نصب Django
۳۰۳	۶-۱۷-۳ ساخت یک پروژه با جنگو
۳۰۶	۶-۱۷-۴ پیاده سازی اپلیکیشن رأی گیری
۳۰۶	۶-۱۷-۵ تفاوت اپلیکیشن و پروژه
۳۰۹	۶-۱۷-۶ نصب پایگاه داده
۳۱۱	۶-۱۷-۷ ساخت مدل ها در جنگو
۳۱۲	۶-۱۷-۸ فعال سازی مدل ها
۳۱۹	۶-۱۷-۹ معرفی ادمین Django
۳۱۹	۶-۱۷-۱۰ ساخت یک کاربری ادمین
۳۲۰	۶-۱۷-۱۱ راه اندازی سرویس دهنده توسعه
۳۲۱	۶-۱۷-۱۲ ورود به سایت مدیریت (ادمین)
۳۲۳	۶-۱۷-۱۳ ساخت ویو در جنگو
۳۲۹	۶-۱۷-۱۴ استفاده از سیستم قالب
۳۳۰	۶-۱۷-۱۵ حذف URL های hardcoded در قالب ها
۳۳۱	خلاصه فصل ششم
۳۳۱	خودآزمایی چهارگزینه ای فصل ششم
۳۳۳	خودآزمایی تشریحی فصل ششم
۳۳۵	فصل هفتم: برنامه نویسی سوکت
۳۳۵	هدف کلی
۳۳۵	هدف های یادگیری
۳۳۵	مقدمه
۳۳۶	۷-۱ مدل سرویس دهنده / مشتری
۳۳۷	۷-۲ مفهوم سوکت
۳۳۷	۷-۳ انواع سوکت
۳۴۱	۷-۴ زبان برنامه نویسی جاوا
۳۴۱	۷-۴-۱ برنامه نویسی سوکت با جاوا
۳۴۴	۷-۴-۲ برنامه نویسی سوکت با پایتون
۳۵۲	خلاصه فصل هفتم
۳۵۲	خودآزمایی چهارگزینه ای فصل هفتم
۳۵۴	خودآزمایی تشریحی فصل هفتم

۳۵۵	فصل هشتم: آشنایی با برخی ابزارهای طراحی صفات وب
۳۵۵	هدف کلی
۳۵۵	هدف‌های یادگیری
۳۵۵	مقدمه
۳۵۶	۱-۸ طراحی وب‌سایت با نرم‌افزار Dreamweaver
۳۵۷	۲-۸ طراحی وب‌سایت با نرم‌افزار Google Web Designer
۳۵۸	۳-۸ طراحی وب‌سایت با نرم‌افزار Rapidweaver
۳۵۹	۴-۸ طراحی وب‌سایت با نرم‌افزار KompoZer
۳۶۳	۵-۸ طراحی سایت با استفاده از cms آماده
۳۶۴	۱-۵-۸ طراحی سایت با نرم‌افزار DNN (DotNetNuke)
۳۶۵	۲-۵-۸ طراحی سایت با نرم‌افزار Joomla
۳۶۶	۳-۵-۸ طراحی سایت با نرم‌افزار word press
۳۶۷	۴-۵-۸ طراحی سایت با نرم‌افزار Drupal
۳۶۸	خلاصه فصل هشتم
۳۶۹	خودآزمایی چهارگزینه‌ای فصل هشتم
۳۶۹	خودآزمایی تشریحی فصل هشتم
۳۷۱	پاسخنامه
۳۷۳	واژه‌نامه انگلیسی - فارسی
۳۷۹	اختصارات
۳۸۵	منابع

پیشگفتار

فناوری ارتباطات و اطلاعات (فاوا) و انقلابی که از آن به عنوان انقلاب سوم یاد می شود، تغییرات گسترده ای را در چند سال اخیر در ابعاد مختلف زندگی جوامع اعم از فرهنگی، اجتماعی و اقتصادی پدید آورده است. این فناوری مهم ترین ابزار، روش و سرمایه برای توانمندسازی جوامع در قرن جدید محسوب شده و یک فرصت استثنایی برای جبران عقب ماندگی های کشورهای در حال توسعه در ورود به دوران فراصنعتی به شمار می رود.

کتاب حاضر بر اساس سرفصل درس برنامه نویسی وب تهیه و گردآوری شده است. در این کتاب زبان های برنامه نویسی HTML ، JavaScript ، XML ، فناوری Ajax ، پایتون و PHP معرفی شده است. در این روند سعی شده دانشجو به صورت گام به گام با زبان های اینترنتی و اصول طراحی سایت آشنا شده و در نهایت بتواند یک سایت پویا شامل پایگاه داده را طراحی کند.

از خواننده های عزیزی که از این کتاب استفاده می کنند صمیمانه تقاضا داریم نظرات خود را جهت بهبود در ویرایش بعدی عنوان نمایند.

فصل اول

برنامه‌نویسی HTML^۱

هدف کلی

آشنایی با زبان برنامه‌نویسی HTML و توابع موجود در آن

هدف‌های یادگیری

- با مطالعه این فصل انتظار می‌رود دانشجو بتواند:
۱. آشنایی با انواع زبان‌های برنامه‌نویسی تحت وب
 ۲. خصوصیات تگ‌های HTML را بشناسد.
 ۳. پرستفاده‌ترین تگ‌های HTML را بشناسد.
 ۴. با نحوه ایجاد فرم و جدول در HTML آشنا شود.
 ۵. آشنایی مقدماتی با چهارچوب نرم‌افزاری بوت استرپ.

مقدمه

همان‌طوری‌که می‌دانید در محیط اینترنت سرویس‌های متعددی به‌منظور ارائه خدمات فعال می‌باشند. بدون شک سرویس وب یکی از مهم‌ترین سرویس‌های موجود روی اینترنت است که بیشترین تأثیر را در عمومیت یافتن اینترنت در سطح جوامع بشری برعهده داشته است. وب در نگاه اول مانند یک سیستم اطلاعاتی توزیع شده است. در

سیستم فوق، اطلاعات (با اشکال متفاوت) براساس استانداردهای مربوطه و با یک ساختار مشخص قرار گرفته و با استفاده از ارتباطات تعریف شده بین اطلاعات می‌توان به سرعت و به سادگی از اطلاعات موجود استفاده نمود. ساختار وب براساس مدل مشتری و سرویس دهنده است و اگر بخواهید در این محیط برنامه کاربردی بنویسید باید با زبان‌های تحت وب و ساختار برنامه‌نویسی آن‌ها آشنا شوید. در ادامه ابتدا زبان‌های برنامه‌نویسی را از جنبه‌های مختلف تقسیم‌بندی نموده و پس از شناخت جایگاه زبان‌های تحت وب یکی از زبان‌های اولیه در اینترنت یعنی HTML توضیح داده می‌شوند.

۱-۱ تقسیم‌بندی زبان‌های برنامه‌نویسی

زبان‌های برنامه‌نویسی را از چهار دیدگاه مختلف می‌توان تقسیم‌بندی نمود:

الف) روش‌های برنامه‌نویسی

۱. زیرروالی: اولین قدم در پیشرفت برنامه‌نویسی است. در این راستا زبان‌های فترن و کوپول عرضه شدند. در این روش، برنامه‌نویس مسائل را به چندین عملکرد تقسیم می‌کند و با هر عملکرد یا تابع به صورت جداگانه رفتار می‌کند اما پیچیدگی تابع هنوز پابرجاست.

۲. ساخت یافته: در دهه ۱۹۶۰ میلادی توسعه نرم‌افزار دچار مشکلات عدیده‌ای شد. در آن زمان سبک خاصی برای برنامه‌نویسی وجود نداشت و برنامه‌ها بدون هیچ‌گونه ساختار خاصی نوشته می‌شدند. وجود دستور پرش (goto) نیز مشکلات بسیاری را برای فهم و درک برنامه توسط افراد دیگر ایجاد می‌نمود، چراکه جریان اجرای برنامه به طور مرتب تغییر جهت داده و دنبال نمودن آن دشوار می‌شد. لذا نوشتن برنامه‌های عملی بسیار زمان‌بر و پرهزینه بود و معمولاً اشکال زدایی، اعمال تغییرات و گسترش برنامه‌ها بسیار مشکل بود. فعالیت‌های پژوهشی در این دهه باعث به وجود آمدن برنامه‌نویسی ساخت یافته شد؛ روش منظمی که باعث ایجاد برنامه‌هایی کاملاً واضح و خوانا شد که اشکال زدایی و خطایابی آن‌ها نیز بسیار ساده‌تر بود.

اصلی‌ترین نکته در این روش عدم استفاده از دستور پرش (goto) است. تحقیقات بوهم و ژاکوپینی نشان داد که می‌توان هر برنامه‌ای را بدون دستور پرش و فقط با استفاده از ۳ ساختار کنترلی یعنی ترتیب، انتخاب و تکرار نوشت. ساختار ترتیب، همان اجرای دستورها به صورت متوالی (یکی پس از دیگری) است که کلیه زبان‌های برنامه‌نویسی در حالت عادی به همان صورت عمل می‌کنند. ساختار انتخاب به برنامه‌نویس اجازه می‌دهد که براساس درستی یا نادرستی یک شرط، تصمیم بگیرد و کدام مجموعه از دستورها اجرا شود. ساختار تکرار نیز به برنامه‌نویسان اجازه می‌دهد مجموعه خاصی از دستورها را تا زمانی که شرط خاصی برقرار باشد، تکرار نمایند. هر برنامه ساخت یافته از تعدادی بلوک تشکیل می‌شود که این بلوک‌ها به ترتیب اجرا می‌شوند تا برنامه خاتمه یابد (ساختار ترتیب). هر بلوک می‌تواند یک دستور ساده مانند خواندن، نوشتن یا تخصیص مقدار به یک متغیر باشد و یا اینکه شامل دستورهایی باشد که یکی از سه ساختار فوق را پیاده‌سازی کنند. نکته مهم اینجاست که در مورد دستورهای داخل هر بلوک نیز همین قوانین برقرار است و این دستورها می‌توانند از تعدادی بلوک به شرح فوق ایجاد شوند و تشکیل ساختارهایی مانند حلقه‌های تودرتو را دهند. نکته مهم اینجاست که طبق قوانین فوق یک حلقه تکرار یا داخل حلقه دیگر است و یا خارج آن قرار می‌گیرد و هیچ‌گاه حلقه‌های روی هم افتاده نخواهید داشت. زبان پاسکال در سال ۱۹۷۱ توسط پروفیسور نیکلاس ویرث و کمی بعد زبان C توسط دنیس ریچی، از این نوع هستند. با ورود این زبان‌ها نیاز به نسخه اول BASIC کم شده و نسخه‌های جدیدی از آن مانند QBASIC برای رفع کمبودهای نسخه اول به بازار عرضه شد که ضمن حفظ سادگی بیسیک، قابلیت برنامه‌نویسی ساخت یافته را نیز به آن افزودند.

۳. ماژولار: این نوع برنامه‌نویسی که به نام قطعه قطعه نیز معروف است، نوع توسعه یافته برنامه ساخت یافته است که در آن قطعات کد به قسمت‌های کوچک‌تری به نام ماژول تقسیم می‌شوند. از دیگر ویژگی‌های برنامه‌نویسی ماژولار کپسوله سازی اطلاعات است. یعنی هر یک از متغیرها و قطعه کدها در محدوده کاری خود فعال هستند و این فعالیت از دید سایر قسمت‌های برنامه پنهان می‌باشد.

۴. شیء‌گرا: این روش از بهترین مفاهیم برنامه‌نویسی به روش ساخت یافته به وجود آمده و با چندین مفهوم قوی ترکیب شده تا امکان سازمان‌دهی برنامه‌ها را به‌طور

کارآمد فراهم کند. از طریق این روش پیچیدگی کاهش می‌یابد و می‌توان برنامه‌های بزرگ‌تر را هدایت نمود. در این روش واحد اصلی برنامه شیء نامیده می‌شود. هر شیء شامل ویژگی‌هایی است که حالت فعلی شیء را مشخص می‌کند و در اثر رویدادهای مختلف رفتار یا عکس‌العمل نشان می‌دهد. می‌توان به زبان‌هایی مانند ویژوال بیسیک و سی شارپ (از شرکت مایکروسافت)، دلفی (از شرکت بورلند) و جاوا (از شرکت سان) اشاره نمود.

ب) نزدیکی به زبان ماشین

۱. سطح بالا: زبان‌هایی هستند که به زبان انسان یعنی زبان محاوره‌ای انگلیسی نزدیک و فهم آن‌ها برای انسان بسیار آسان است ولی برای کامپیوتر فقط به کمک مترجم زبان برنامه‌نویسی یعنی کامپایلرها و مفسرها قابل فهم هستند. امروزه بیشتر برنامه‌نویسان این زبان را برای برنامه‌نویسی انتخاب می‌کنند. مثال‌هایی از این زبان‌ها عبارت‌اند از بیسیک، فرترن، پاسکال، C.

۲. سطح میانی: این زبان شکل ساده‌تر زبان ماشین است، به این صورت که برای هر دستورالعمل زبان ماشین، یک اسم نمادین انتخاب شده است (مانند دستور ADD به جای کد دودویی دستورالعمل جمع یا مانند LDA برای قراردادن اطلاعات در انباره) که به خاطر سپردن و برنامه‌نویسی با آن‌ها برای انسان‌ها ساده‌تر است. اما این برنامه‌ها برای ماشین قابل فهم نیست و باید قبل از اجرا شدن توسط برنامه مترجمی به نام اسمبلر به زبان ماشین تبدیل شود.

۳. سطح پایین: این زبان به زبان کامپیوتر (زبان صفر و یک) بسیار نزدیک و فهم آن برای کامپیوتر بسیار آسان است. اما چون به صورت صفر و یک است برای انسان قابل فهم نیست و اگر هم باشد انسان به سختی می‌تواند آن را بفهمد. طراحان سخت‌افزار هر کامپیوتر، زبان ماشین خاص خود را برای آن ماشین طراحی می‌نمایند. به همین دلیل هر برنامه‌ای که به زبان ماشین نوشته شود، فقط روی همان ماشین خاص کار می‌کند، به همین دلیل برنامه‌های نوشته شده به زبان ماشین را غیرقابل حمل می‌نامند.

ج) نوع ترجمه

۱. کامپایلری: در این نوع زبان‌ها ابتدا کل برنامه از لحاظ خطای املائی (نه خطای منطقی) بررسی شده، در صورتی که خطایی نداشته باشد، برنامه به زبان ماشین ترجمه و خروجی Object رمز تولید می‌شود. خروجی حاصل می‌تواند روی کامپیوتری که عمل کامپایلر روی آن صورت گرفته است، اجرا شود (پسوندهای com و exe). به‌طور کلی مراحل کامپایلر را می‌توان به چهار قسمت تشخیص توکن (دستور)، پارس، تولید کد زبان ماشین و بهینه‌سازی کدهای تولیدشده تقسیم نمود. از انواع زبان کامپایلری می‌توان زبان برنامه‌نویسی پاسکال و C را نام برد. از مزایای برنامه‌های کامپایلر شده می‌توان به سرعت بالای اجرای برنامه‌ها و عدم نیاز به کد منبع و کامپایلر در زمان اجرا اشاره نمود.

۲. مفسری: مفسر نیز کار کامپایلر را انجام می‌دهد، با این تفاوت که مفسر، دستورهای برنامه را خط به خط می‌خواند (تشخیص توکن در یک سطر) سپس مطابق با کتابخانه خود آن‌ها را پارس نموده و بعد اجرا می‌کند، سپس نوبت دستور بعدی است. به عبارت دیگر هر مفسر دارای حلقه‌ای است که هر بار و برای هر خط از برنامه مراحل تشخیص توکن، پارس و اجرا انجام می‌شود. علی‌رغم سادگی، زبان‌های تفسیری در پیاده‌سازی زبان چند اشکال مهم دارند:

- نمی‌توانند کدها را بهینه نمایند (مرحله چهارم از عمل کامپایلر). این مشکل زمانی که فقط یک خط از دستور را در نظر بگیریم، زیاد بد به نظر نمی‌رسد، ولی زمانی که بحث در مورد یک برنامه حجیم با محاسبات پیچیده باشد، مشکل کاملاً نمایان می‌شود.
- کندی اجرای برنامه‌ها در مقایسه با زبان‌های کامپایلری.
- نیاز به تفسیر هر دستور حتی اگر در یک حلقه آن دستور را هزاران بار داشته باشید.
- در ترجمه می‌توان یک‌بار ترجمه نمود و چندبار از آن استفاده نمود ولی در تفسیر یک‌بار ترجمه و یک‌بار اجرا می‌شود.
- در روش تفسیر به دلیل یک مرحله‌ای بودن ترجمه و اجرا ممکن است کلیه خطاها کشف نشود ولی در روش ترجمه چون در دو فاز مختلف (فاز اول کامپایل و فاز دوم اجرا) انجام می‌شود کلید خطاها قابل کشف هستند.

(د) رابط برنامه‌نویسی

۱. مبتنی بر متن: برنامه‌هایی که همه‌چیز در آن با نوشتن کدها تولید می‌شود و در آن خروجی در اکثر مواقع به صورت متن است. زبان‌های مبتنی بر متن به کاربر امکان دسترسی مستقیم به گرافیک را نمی‌دهند. یک بسته نرم‌افزاری ویژه اغلب به عناصر گرافیکی نیاز دارد که به آن افزوده شود. این بسته نرم‌افزاری اغلب به نوشتن خطوط زیادی از کدها نیاز دارد که به کد برنامه اصلی اضافه می‌شوند. مثل زبان C و پاسکال.
۲. مبتنی بر گرافیک (ویژوال): که در آن برای اشیاء، برنامه نوشته می‌شود و خروجی به صورت اشیاء گرافیکی است. زبان‌های مبتنی بر گرافیک امکان دسترسی مستقیم کاربر به گرافیک را فراهم می‌کنند. از زبان‌های مبتنی بر گرافیک می‌توان برای ایجاد سریع برنامه‌های تحت ویندوز نیز استفاده نمود، بدون اینکه نیاز به یادگیری و استفاده از بسته‌های نرم‌افزاری اضافه داشته باشد. مانند Visual basic, visual c++ از محصولات شرکت مایکروسافت یا دلفی از محصولات شرکت بولند. این تقسیم‌بندی‌ها در شکل ۱-۱ آورده شده است.



شکل ۱-۱. تقسیم‌بندی کلی زبان‌های برنامه‌نویسی

۱-۲ تقسیم‌بندی زبان‌های برنامه‌نویسی تحت وب

زبان‌های برنامه‌نویسی که در محیط اینترنت استفاده می‌شوند را می‌توان از جنبه مکان اجرا به دودسته کلی تقسیم‌بندی نمود: زبان‌های سمت مشتری یا سرویس‌گیرنده (Client Side) و زبان‌های سمت سرویس‌دهنده (ServerSide).

- سمت مشتری: برنامه‌هایی که با این زبان‌ها نوشته شوند هرگز توسط سرویس‌دهنده پردازش نمی‌شوند، بلکه تنها توسط مرورگر سرویس‌گیرنده پردازش می‌شود. به این معنی است که وقتی شما یک صفحه سمت مشتری را در مرورگر درخواست می‌کنید، سرویس‌دهنده کاری انجام نداده و مرورگر مشتری کدها را خط به خط خوانده و نمایش می‌دهد. زبان‌هایی مانند HTML، JavaScript یا VBScript از این نوع هستند. زبان HTML یک زبان مفسری ساده برای نمایش داده اطلاعات است که نمی‌شد در آن الگوریتم برنامه‌نویسی داشت برای رفع این مشکل زبان‌های اسکریپتی جهت ایجاد تحول و پویایی در صفحات وب ایجاد شدند. این زبان‌ها از روی زبان‌های برنامه‌نویسی ساخته شدند و به همین دلیل دارای تشابه بسیاری باهم هستند. این زبان‌ها در اصل نمونه کوچک‌شده زبان‌های مادر خود هستند. برای بالابردن امنیت در محیط شبکه و اینترنت تعدادی از دستورها و امکانات زبان‌های بزرگ در این زبان‌ها حذف‌شده‌اند مثل امکان نوشتن فایل یا پاک نمودن فایل‌ها روی سیستم کاربر، مانند زبان‌های برنامه‌نویسی وجود ندارد.

- سمت سرویس‌دهنده: این نوع اسکریپت‌ها تماماً روی سرویس‌دهنده وب پردازش می‌شوند. مرورگر سرویس‌گیرنده این نوع اسکریپت‌ها را دریافت ننموده و نمی‌شناسد، بلکه تنها خروجی آن‌ها را دریافت می‌کند. وقتی یک صفحه به زبان‌های سمت سرویس‌دهنده نوشته می‌شود، این صفحه وب تنها شامل کدهای HTML نیست بلکه شامل کدهای برنامه‌نویسی به زبان‌های دیگر نیز مانند پایتون، php، asp یا asp.net نیز می‌باشد که مرورگر مشتری قادر به درک آن‌ها نیست. این کدها توسط یک سرویس‌دهنده وب خوانده و به زبان HTML ترجمه می‌شوند و سپس برای مرورگر مشتری ارسال می‌شوند. در ادامه برنامه‌نویسی به زبان HTML که ازجمله زبان‌های اولیه مفسری در اینترنت است توضیح داده می‌شود.