



برای آن‌که بتوانید از اطلاعات یا داده‌ها در برنامه کاربردی خود استفاده کنید؛ در اولین گام باید با نحوه نمایش و مقایسه نوع‌های داده‌ای آشنا شوید. در یک برنامه کاربردی شما با انواع مختلفی از نوع‌های داده‌ای سروکار دارید. در دنیای برنامه‌نویسی برای انجام محاسبات مختلف روی نوع‌های داده‌ای باید از عملگرهای درستی استفاده کنید. در غیر این صورت ممکن است نتیجه‌ای که به دنبال آن هستید را دریافت نکرده یا برنامه شما با نمایش پیغام خطایی متوقف شود. شما می‌توانید از متغیرها برای دسترسی به داده‌ها و ویرایش آن‌ها متناسب با نیاز کاری خود استفاده کرده و نتایج را ارزیابی کنید. اما چگونه می‌توانیم بر روند نمایش داده‌ها در برنامه کاربردی خود کنترل داشته باشیم؟

برای مطالعه بخش هشتم آموزش رایگان پایتون [اینجا](#) کلیک کنید

در زمان کدنویسی نکته مهمی که باید به آن توجه داشته باشید، قابل فهم بودن کدها است. در دنیای برنامه‌نویسی بارها و بارها شرایطی پیش می‌آید که مجبور هستید کدهای پیچیده و مشکلی را بنویسید و مهم‌تر از آن مجبور هستید در بخش‌های مختلف یک برنامه از کدهای خود استفاده کنید. اگر از راهکار موثری برای کم کردن تعداد کدهای برنامه خود استفاده نکنید، در عمل برنامه شما انباشته از کدهای یکسان و تکراری می‌شوند که برای انجام یک کار نوشته شده‌اند. برای حل این مشکل زبان‌های برنامه‌نویسی راهکاری موسوم به متدها/توابع را پیشنهاد دادند. توابع یک راهکار قدرتمند هستند که به شما اجازه می‌دهند کدهای خود را بسته‌بندی کنید. این کار نه تنها مانع از تکرار مجموعه دستورات می‌شود، بلکه کدهای شما را قابل فهم‌تر کرده و خوانایی کدها را افزایش می‌دهد.

چگونه می‌توانیم نحوه نمایش داده‌ها در پایتون را کنترل کنیم؟

در مقاله‌های قبل به شما گفتیم که اطلاعات در کامپیوتر شما به شکل صفرها و یک‌ها ذخیره می‌شوند. کامپیوترها نمی‌توانند مفهوم کاراکترها، مقادیر منطقی، تاریخ‌ها، زمان یا هر نوع اطلاعاتی که غیر عددی هستند را درک کنند. زمانی که با یک رشته در پایتون کار می‌کنید، باید ببینید که پایتون از چه راهکاری استفاده می‌کند تا به کامپیوتر شما مفهوم یک رشته را نشان دهد. متغیرها، جعبه‌های نگه‌دارنده‌ای هستند که برنامه شما از آن‌ها استفاده کرده و به پایتون می‌گوید که چگونه باید صفرها و یک‌ها را درون کامپیوتر شما ذخیره کند. بنابراین، مهم است که بدانید دیدگاه پایتون نسبت به داده‌ها متفاوت از دیدگاه شما و متفاوت از دیدگاه کامپیوترتان نسبت به رشته‌ها است. در حقیقت پایتون نقش یک واسط را داشته و به برنامه کاربردی شما اجازه می‌دهد با رشته‌ها به گونه‌ای کار کند که در عین حال کامپیوتر نیز مفهوم رشته‌ها را متوجه شود. برای مدیریت داده‌ها در یک برنامه کاربردی، باید نحوه نمایش داده‌ها

ا را کنترل کنید. عملگرها، توابع و حتا کنترل ورودی کاربر از جمله راهکارهایی هستند که اجازه می‌دهند بر نحوه ورود یا نمایش داده‌ها مدیریت اعمال کنید. شما از طریق مقایسه کردن داده‌ها قادر به کنترل نحوه نمایش داده‌ها هستید.

نحوه ارزیابی و مقایسه داده‌ها

پایتون راهکار مستقیمی در اختیارتان قرار می‌دهد تا بدون مشکل بتوانید داده‌ها را با یکدیگر مقایسه کنید. این راهکار در ماهیتی به نام عملگرها مستتر شده است. اما چرا به مقایسه داده‌ها نیاز داریم؟ فرض کنید برنامه کاربردی شما به یک بانک اطلاعاتی که شامل رکوردهای اطلاعاتی و سوابق بیمه شدگان است متصل است. برای آن‌که به کاربری که قصد دارد اطلاعات مربوط به سوابق بیمه خود را مشاهده کند اجازه دهید که به اطلاعاتش دسترسی پیدا کند، مجبور هستید یک فرآیند ارزیابی را انجام دهید. زمانی که کاربر در فیلدهای ورودی شماره ملی و شماره تامین اجتماعی خود را وارد می‌کند، شما باید این اطلاعات را با اطلاعات موجود در بانک اطلاعاتی مقایسه کنید تا مطمئن شوید کاربری که قصد دسترسی به اطلاعات را دارد همان کاربری است که اطلاعاتش درون بانک اطلاعاتی ثبت شده است.

عملگرها ضمن آن‌که به شما اجازه می‌دهند مقایسه‌ای میان داده‌ها داشته باشید، همچنین اجازه می‌دهند اطلاعات نیز دستکاری کنید. البته دقت کنید که عملگرها و همچنین فرآیند مقایسه کردن تنها محدود به داده‌ها نمی‌شوند. در برخی موارد شما نیاز دارید خروجی دو تابع را بررسی کنید. **پایتون** به شما اجازه می‌دهد در سطوح مختلفی فرآیند ارزیابی را انجام داده و بدون آن‌که به مشکلی برخورد کنید، در برنامه خود داده‌ها را مدیریت کنید.

کامپیوترها چگونه فرآیند مقایسه را انجام داده و آن را مدیریت می‌کنند؟

کامپیوترها در ارتباط با متدهای بسته‌بندی همچون توابع یا هرگونه ساختاری که در **پایتون** ایجاد می‌کنید هیچ‌گونه شناختی ندارند. همه این راهکارها با هدف ساده کردن برنامه‌نویسی ابداع شده‌اند. با این حال، کامپیوتر در ارتباط با بیشتر عملگرها قادر هستند یک محاسبه مستقیم را انجام دهند. بیشتر عملگرهای تعریف شده در **پایتون** یک خروجی مستقیم دارند. به عبارت دقیق‌تر، کامپیوترها معنای واقعی بیشتر عملگرهای **پایتون** را درک می‌کنند. به طور مثال، زمانی که از کامپیوتر سوال می‌کنید کدامیک از دو مقدار بزرگ‌تر از دیگری است، کامپیوتر یک محاسبه مستقیم را با عملگر مربوطه انجام می‌دهد. با این حال، همه ارزیابی‌ها به شکل مستقیم انجام نمی‌شود. کامپیوترها فقط با اعداد کار می‌کنند. در نتیجه زمانی که از **پایتون** درباره انجام مقایسه‌ای میان دو رشته سوال می‌کنید، کاری که پایتون انجام می‌دهد این است که مقدار عددی هر کاراکتری که درون یک رشته قرار دارد را مقایسه می‌کند. به طور مثال کاراکتر A دارای مقدار عددی 65 است. اما کاراکتر a با حرف کوچک دارای مقدار عددی 97 است. پس $a > A$ است. با این توصیف شما در دنیای واقعی ABC را مساوی با abc می‌دانید، در حالی که برای کامپیوترها این‌گونه نیست، زیرا آن‌ها فرآیند مقایسه را بر مبنای مقدار عددی کاراکترها انجام می‌دهند.

کار با عملگرها

عملگرها پایه و اساس کنترل و مدیریت داده‌ها در یک برنامه کاربردی را شکل می‌دهند. شما از عملگرها برای تعریف این‌که چگونه یک بخش از داده‌ها باید با بخش دیگری مقایسه شده و چگونه اطلاعاتی که درون یک متغیر قرار دارند باید ویرایش شوند استفاده می‌کنید. در حقیقت، عملگرها برای انجام هر فرآیندی که مرتبط با دنیای ریاضیات بوده و زمانی که قرار است انتصابی انجام شده و داده‌ای به متغیرها تخصیص داده شود استفاده می‌شوند. زمانی که از یک عملگر استفاده می‌کنید، باید از یک متغیر یا یک عبارت استفاده کنید. در شماره‌های گذشته با متغیرها و مفهوم آن‌ها آشنا شدید. اما یک عبارت چیست؟ یک عبارت یک معادله یا فرمولی است که یک مفهوم ریاضی را تشریح می‌کند. در اغلب موارد، نتیجه ارزیابی یک عبارت برابر با یک مقدار منطقی درست یا غلط است.

تعریف عملگرها

یک عملگر می‌تواند یک یا چند ورودی را در قالب متغیرها یا عبارات دریافت کرده، عملی را انجام داده (مقایسه یا ویرایش) و سپس خروجی را مطابق با نتیجه مدنظر برنامه‌نویس ارائه کند. عملگرها با توجه به کاری که انجام می‌شوند و همچنین میزان اثرگذاری که دارند طبقه‌بندی می‌شوند. به طور مثال، یک عملگر یگانه تنها با یک متغیر یا یک عبارت کار می‌کند، در مقابل یک عملگر باینری به دو متغیر یا عبارت نیاز دارد. عناصری که به عنوان ورودی به یک عملگر تخصیص داده می‌شوند عملوند نام دارند. عملوندهایی که در سمت چپ یک عملگر قرار می‌گیرند به نام

عملوند سمت چپ نامیده می‌شوند، درحالی که عملوندهایی که در سمت راست یک عملگر قرار می‌گیرند به نام عملوند سمت راست نامیده می‌شوند. عملگرها در پایتون به شرح یگانه (Unary)، ریاضی (Arithmetic)، رابطه‌ای (Relational)، منطقی (Logical)، بیتی (bits)، انتصابی (Assignment)، عضویت (Membership) و شناسه‌ای (Identity) طبقه‌بندی می‌شوند.

عملگرهایی که در گروه‌های مختلف قرار می‌گیرند برای انجام کار خاصی استفاده می‌شوند. به‌طور مثال، عملگرهای ریاضی برای انجام عملیاتی که ریشه ریاضی دارند استفاده می‌شوند، در حالی که عملگرهای رابطه‌ای برای انجام مقایسه‌ها استفاده می‌شوند.

در شماره آینده آموزش پایتون به معرفی عملگرهای متعلق به هر گروه خواهیم پرداخت.

تاریخ انتشار:

16 بهمن 1397

نشانی منبع:

<https://www.shabakeh-mag.com/workshop/programming/14553/%D8%A2%D9%85%D9%88%D8%B2%D8%B4-%D8%B1%D8%A7%DB%8C%DA%AF%D8%A7%D9%86-%D9%BE%D8%A7%DB%8C%D8%AA%D9%88%D9%86-python-%E2%80%93-%D8%A2%D8%B4%D9%86%D8%A7%DB%8C%DB%8C-%D8%A8%D8%A7-%D8%B9%D9%85%D9%84%DA%AF%D8%B1%D9%87%D8%A7-%D9%88-%D8%B9%D9%85%D9%84%D9%88%D9%86%D8%AF%D9%87%D8%A7>