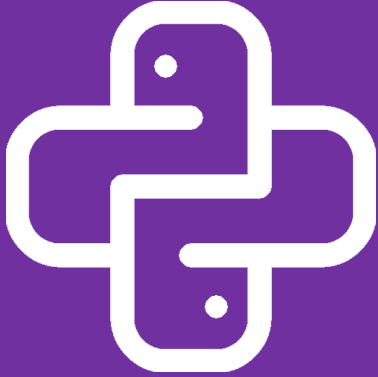


دوره آموزش پایتون - Jet Start



طول دوره: ۶ ساعت

مدرس: هادی تایانلو



عنوان دوره: دوره آموزش پایتون - Jet Start

موضوع: آموزش زبان برنامه نویسی پایتون

مخاطبین: علاقه‌مندان به شروع یادگیری

برنامه‌نویسی به زبان پایتون

طول دوره: ۶ ساعت

نحوه ارائه: به صورت غیرحضوری

مدرس: هادی تایانلو

دسترسی از طریق پلیر اختصاصی اسپات پلیر

روش مشاهده دوره‌های آموزشی محافظت شده



مدرس این دوره کیست؟

هادی تایانلو

CTO و مدیر فنی شرکت eentrans و همچنین مشاور ارشد امنیت شبکه و کارشناس ارشد زیر ساخت دیتاستر،
Enginer Cloud / Enginer Devops

از جمله فعالیت های او می توان به طراحی و پیاده سازی CD/CI مبتنی بر gitlab پیاده سازی زیر ساخت ذخیره سازی Ceph ، پیاده سازی زیر ساخت Siem مبتنی بر Splunk تدریس Splunk تدریس دوره Associate Certified Docker ، طراحی پیاده سازی DevOps پیاده سازی زیر ساخت مبتنی Openstack ، مدیریت و پیاده سازی پروژه های صنعت حمل نقل مدیریت، / پیاده سازی پروژه

های IOT برنامه نویسی به زبان پایتون ، توسعه زیر ساخت حمل و نقل و تجارت الکترونیک طراحی تولید نرم افزار مدیریت حمل و لجستیک، مشاوره و ارائه راهکار ارتقاء سطح امنیت تست زیر ساخت شبکه و تدوین پروسیجر گسترش امن شبکه تست زیر ساخت نرم افزاری و آنالیز تدوین طرح توسعه زیر ساخت شبکه و دیتاستر آموزش نیروهای مستقر به منظور راهبرد سیستم پیاده سازی Splunk پیاده سازی زیر ساخت level OS virtualization مبتنی بر Kubernetes اشاره نمود



بخش اول: مقدمه‌ای بر پایتون

پایتون یک زبان برنامه‌نویسی سطح بالا و شی‌گرا است. پایتون یک زبان ساده و خوانا بوده و تنوع و کاربردهای گسترده‌ای مانند توسعه وب و اپلیکیشن، علم داده و هوش مصنوعی دارد. در بخش اول این دوره، با مفاهیم اولیه و چگونگی نصب و راه‌اندازی پایتون آشنا می‌شویم.

- چرا زبان برنامه‌نویسی پایتون؟
- معرفی کمپانی‌هایی که از زبان پایتون استفاده می‌کنند.
- اهمیت نصب و راه‌اندازی درست
- نصب پایتون در سیستم عامل‌های مختلف مانند لینوکس، ویندوز و مک
- معرفی IDE ها (Visual Studio, Jupyter Notebook, PyCharm)
- نصب IDE محیط توسعه برای شروع کار
- نصب کتابخانه‌ها و مدیریت بسته‌ها
- استفاده از PIP برای نصب کتابخانه‌ها
- معرفی requirements.txt
- معرفی محیط‌های مجازی (Virtual Environments)



بخش دوم: بررسی ساختار پایه در پایتون

متغیر (variable) در پایتون را می‌توانید همانند یک محفظه در نظر بگیرید که برای ذخیره موقت اطلاعات استفاده می‌شود. در این بخش از دوره آموزش پایتون به تعریف متغیرها و اصول نام‌گذاری آن‌ها و همچنین، بررسی رشته‌ها در پایتون می‌پردازیم.

- بررسی اهمیت و کاربردهای مبانی ساختاری پایتون
- تعریف متغیرها
- نام‌گذاری متغیرها
- بررسی انواع داده‌ها در زبان پایتون
- بررسی رشته‌ها در پایتون
- تبدیل انواع داده در پایتون
 - (از int به float و برعکس)
 - از رشته به عدد و برعکس
- بررسی ورودی و خروجی (استفاده از print , input())

بخش سوم: بررسی عبارات شرطی در زبان پایتون

در این بخش، عبارات شرطی را مورد بررسی قرار می‌دهیم و به اهمیت و کاربرد این عبارات را در زبان بررسی پایتون می‌پردازیم. در پایان، با چند تمرین برنامه‌نویسی، آموخته‌های خود را می‌آزمایید.

- اهمیت عبارات شرطی در برنامه‌نویسی و بررسی کاربردهای آن‌ها
- تعریف عبارات شرطی if , elif , else
- توضیح و بررسی اصول اندازه‌گیری شروط
- نحوه ترکیب عبارات شرطی if , elif , else در یک بلوک کد
- بررسی range () در حلقه for
- ارائه تمرین و بررسی مثال‌های گوناگون در این حوزه

```

177         defaults = {}
178     }
179 }
180
181 global scale_setting = FloatProperty(
182     name="Scale",
183     min=0.0, max=1000.0,
184     default=1.0,
185 )
186
187 def execute(self, context):
188     # get the folder
189     folder_path = (os.path.dirname(self.filepath))
190
191     # get objects selected in the viewport
192     viewport_selection = bpy.context.selected_objects
193
194     # get export objects
195     obj_export_list = viewport_selection
196     if self.use_selection_setting == False:
197         obj_export_list = [i for i in bpy.context.scene.objects]
198
199     # deselect all objects
200     bpy.ops.object.select_all(action='DESELECT')
201
202     for item in obj_export_list:
203         item.select = True
204         if item.type == 'MESH':
205             file_path = os.path.join(folder_path, "{}.obj".format(item.name))
206             bpy.ops.export_scene.obj(filepath=file_path, use_selection=True,
207                 axis_forward=self.axis_forward_setting,
208                 axis_up=self.axis_up_setting,
209                 use_animation=self.use_animation_setting,
210                 use_mesh_modifiers=self.use_mesh_modifiers_setting,
211                 use_edges=self.use_edges_setting,
212                 use_smooth_groups=self.use_smooth_groups_setting,
213                 use_smooth_groups_bitflags=self.use_smooth_groups_bitflags_setting,
214                 use_normals=self.use_normals_setting,
215                 use_uv=self.use_uv_setting,
216                 use_armature=self.use_armature_setting,

```



بخش چهارم: بررسی حلقه‌ها در زبان پایتون

حلقه تکرار در پایتون قابلیت‌هایی است که به شما امکان فراخوانی چندین عنصر را می‌دهد و با دستورات آن می‌توانید عناصر مختلف لیست‌ها، تاپل‌ها و دیکشنری‌ها را فراخوانی کنید و عملیات موردنظر را روی آن انجام دهید. در این بخش، به بررسی حلقه‌ها در پایتون می‌پردازیم.

- چرا در زبان برنامه‌نویسی به حلقه‌ها نیاز داریم و چه زمانی از آن استفاده کنیم؟
- تعریف و نحوه نوشتن حلقه `for`
- تعریف و نحوه نوشتن حلقه `while`
- بررسی حلقه‌های تو در تو (Nested Loops)
- ارائه تمرین و بررسی مثال‌های گوناگون در این حوزه

بخش پنجم: بررسی توابع در زبان پایتون

تابع در پایتون گروهی از عبارتهای مرتبط است که یک کار مشخص را انجام می‌دهند. توابع کمک می‌کنند تا برنامه به بخش‌های کوچک‌تر شکسته شود.

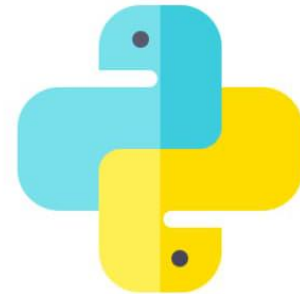
- تابع چیست؟
- چرا نیاز به توابع داریم؟
- استفاده از `def` برای تعریف تابع
- نام‌گذاری توابع
- پارامترها و ورودی‌های تابع:
- a. تعریف پارامترها و تفاوت با آرگومان‌ها
- b. پارامترهای موقعیتی و کلیدی
- ارائه تمرین و بررسی مثال‌های پیرامون توابع در پایتون



بخش پنجم: پیاده‌سازی پروژه‌های کاربردی در حوزه پایتون

- مدیریت کتابخانه شخصی:

در اولین پروژه این دوره، به ایجاد یک برنامه برای مدیریت کتابخانه شخصی می‌پردازیم که امکان ثبت و حذف کتاب‌ها و جستجو براساس نام و یا نویسنده را دارا باشد.



- سیستم رزرواسیون سینما:

در دومین پروژه این دوره، به ایجاد یک برنامه ساده برای مدیریت رزرواسیون صندلی‌ها در یک سالن سینما می‌پردازیم.



نحوه مشاهده دوره چگونه است؟

این دوره آموزشی را می‌توانید در یک پلیمر اختصاصی مشاهده فرمایید. به راحتی می‌توانید این نرم‌افزار را مناسب با سیستم عامل خود (ویندوز، مک، اندروید، لینوکس و یا وب) دانلود نصب کرده و پس از کپی کلید لایسنس داخل نرم‌افزار، محصول خریداری شده را تماشا کنید.



صدور فاکتور رسمی چگونه است؟

در صورت تمایل به دریافت فاکتور رسمی، پیش از خرید خود با واحد فروش مجموعه (۰۲۱ - ۹۱ ۰۷ ۰۰ ۱۷) تماس حاصل نمایید. شایان ذکر است، امکان صدور فاکتور رسمی پس از خرید آنلاین از سایت مجموعه به هیچ عنوان وجود نخواهد داشت.

آدرس: تهران، یوسف آباد، میدان فرهنگ، خیابان ۳۳، پلاک ۲۹، زنگ ۲، دفتر نیک آموز
شماره تماس: ۰۲۱ - ۹۱ ۰۷ ۰۰ ۱۷ | موبایل فروش: ۰۹۱۰ ۴۰۰۶ ۲۰۶