

به نام خدا

آموزش چهارم

اهداف آموزشی این قسمت عبارتند از:

۱. آشنایی با علت به کارگیری پکیج ها در توسعه اندروید
۲. آشنایی با نحوه نامگذاری صحیح پکیج ها با ذکر مثال

پیش از ساخت اولین پروژه خود در محیط برنامه نویسی اندروید استودیو، نیاز است تا با نحوه نامگذاری صحیح پکیج ها در سیستم عامل اندروید آشنا شویم.

علت به کارگیری و همچنین نحوه نامگذاری پکیج ها

در زبان برنامه نویسی جاوا استفاده از Package راهی است که از آن طریق می توان کلاس های مرتبط با یکدیگر را سازماندهی کرد. در حقیقت کلاس هایی که در یک پکیج قرار می گیرند خواهند توانست به راحتی به یکدیگر دسترسی پیدا کنند. نحوه نامگذاری پکیج ها در جاوا به این شکل است که نام پکیج با نام دومین شرکت، سازمان و یا وب سایت شخصی آغاز می شود. به عبارت دیگر نام پکیج سلسله مراتبی است به این شکل که با کلی ترین نام که نام دومین است شروع می شود. سپس یک نقطه قرار می دهیم (در زبان جاوا فقط و فقط برای نامگذاری پکیج ها برای جدا سازی بخش های مختلف نام پکیج می توانیم از نقطه استفاده کنیم). حال نام اصلی وب سایت شرکت و یا سازمان و یا وب سایت شخصی را نوشته و مجدداً یک نقطه قرار می دهیم. پس از این نقطه نامی برای پکیج خود در نظر می گیریم. برای روشن تر شدن مطالب فوق پکیج پروژه خود را به این صورت زیر نامگذاری می کنیم:

اول نام دومین شرکت، سازمان و یا وب سایت خود را می نویسیم که در اینجا com است سپس یک نقطه قرار داده و بعد از آن نام اصلی وب سایت را نوشته که در این مثال behzadmoradi می باشد و مجدداً پس از قرار دادن نقطه ای دیگر نام دلخواه پکیج را می نویسیم که در این مثال نام package به معنی پکیج در نظر گرفته شده است.

چنانچه بخواهیم علت این طرز نامگذاری را بیان کنیم بایستی بگوییم که با نامگذاری پکیج ها به صورت برعکس از به هم ریختگی سورس های برنامه خود جلوگیری به عمل خواهد آمد. برای روشن شدن فایده این کار به مثال زیر توجه کنید:

دوره آموزش اندروید

کلیه حقوق متعلق به وب سایت نردبان است.

مدرس: بهزاد مرادی

فرض کنیم که قرار است برنامه ای طراحی کنیم که اخبار روزانه را در اختیار کاربر قرار دهد. در این برنامه می خواهیم دو پکیج تعریف کنیم که یکی برای اخبار روزانه باشد و پکیج دیگر برای آرشیو خبرها باشد. اگر نامگذاری پکیج ها به صورت برعکس نباشد، نامی که برای پکیج مرتبط با اخبار روزانه خواهیم ساخت به شکل `news.behzadmoradi.com` بوده و پکیج مرتبط با آرشیو به صورت `archive.behzadmoradi.com` خواهد شد. در حقیقت اشکال این نوع نامگذاری در اینجا است که این ذهنیت القاء می شود که یک پکیج اصلی به نام `news` وجود دارد که زیر مجموعه آن پکیج `behzadmroadi` است و زیر مجموعه پکیج `behzadmoradi` پکیج `com` است. حال ببینیم اگر نحوه نامگذاری را به صورت برعکس انجام دهیم چه معنایی القاء خواهد شد. نام پکیج اول را به `com.behzadmoradi.news` و نام پکیج دوم را به `com.behzadmoradi.archive` تغییر می دهیم. این نامگذاری مسلماً با معنی تر خواهد بود چرا که در این صورت ما یک پکیج کلی به نام `com` داریم که زیر مجموعه اش پکیج دیگری به نام `behzadmoradi` است و زیر مجموعه `behzadmoradi` دو پکیج به نام های `news` و `archive` وجود دارد. حال ببینیم که چرا در نامگذاری پکیج ها از نام دومین استفاده می کنیم. علت آن است که با این کار از تشابه اسمی به طور صد در صد جلوگیری به عمل خواهد آمد چرا که در کل دنیا فقط یک وب سایت به آدرس `behzadmoradi.com` وجود دارد. برای پی بردن به اهمیت این کار به مثال زیر توجه کنید. فرض کنیم که دو شرکت مختلف طراح `Library` های اندرویدی هستند به طوری که پکیج هایی می نویسند که دیگر برنامه نویسان صرفاً با `Import` کردن آن پکیج ها خواهند توانست سرعت طراحی اپلیکیشن خود را بالا ببرند و دیگر مجبور نباشند تا پکیجی را خود از اول بنویسند (در واقع `Library` به مجموعه کلاس هایی گفته می شود که با یکدیگر لینک بوده و کار خاصی را انجام می دهند). حال اگر هر دو شرکت در نحوه نامگذاری پکیج های خود از نام دومین خود استفاده نکنند، این احتمال وجود دارد که به صورت اتفاقی هر دو شرکت نام یکسانی را برای پکیج خود انتخاب کنند مثلاً هر دو نام `package.news` را برای پکیج مرتبط با آرشیو اخبار انتخاب کنند. اکنون ما در برنامه خود از این پکیج ها استفاده می کنیم اما نکته ای را که هرگز مد نظر قرار نداده ایم این است که در جایی از برنامه از پکیج `package.news` شرکت اول استفاده کرده و در جایی دیگر از پکیج `package.archive` شرکت دوم استفاده کرده ایم. از آنجا که ممکن است برخی کلاس های این پکیج ها با همدیگر ارتباط داشته باشند و ما به صورت ناخودآگاه این لینک را از بین برده ایم برنامه ما `Crash` خواهد کرد. اکنون ببینیم اگر نام این دو شرکت در نام پکیج ها گنجانده شود چه فایده ای خواهد داشت. فرض کنیم که نام شرکت اول ما `companyone` باشد و نام شرکت دوم ما `companytwo` باشد. حال شرکت `companyone` دو `Library` طراحی کرده است به نام های

دوره آموزش اندروید

کلیه حقوق متعلق به وب سایت نردبان است.

مدرس: بهزاد مرادی

com.companyone.package.archive و com.companyone.package.news و

از سوی دیگر شرکت companytwo هم دو library به نام های

com.companytwo.package.archive و com.companytwo.package.news

طراحی کرده است. به طور خلاصه این نحوه نامگذاری به ما کمک خواهد کرد که اگر خواستیم در برنامه خود از این Library ها استفاده کنیم، دو Library یی را مورد استفاده قرار دهیم که از یک شرکت بوده و به عبارت دیگر با یکدیگر مرتبط می باشند.

به عنوان نکته پایانی در مورد نحوه نامگذاری پکیج ها در اندروید بایستی گفت که چنانچه کسی دارای وب سایت شخصی نباشد می تواند به صورت فرضی نامی را برای خود در نظر بگیرد.

پس از مطالعه این آموزش انتظار می رود بتوانیم به سؤالات زیر پاسخ بدهیم:

۱. وظیفه پیکج ها چیست؟

۲. چرا در نام پیکج ها از آدرس دومین شرکت یا سازمان می بایست استفاده کرد؟

۳. منظور از یک Library چیست؟

در آموزش پنجم اقدام به آماده سازی محیط برنامه نویسی اندروید استودیو خواهیم کرد.

دوره آموزش اندروید

کلیه حقوق متعلق به وب سایت نردبان است.

مدرس: بهزاد مرادی