

پاسخ سوالات دستور کار جلسه چهارم کارگاه برنامه نویسی پیشرفته

یکی از راه های افزودن فایل jar، توسط IDE ها است. افزودن این نوع فایل توسط IntelliJ در شکل های ۱ تا ۳ آورده شده است. از جمله مجموعه کتابخانه های معروف و پرکاربرد موجود می توان به Google Guava و Apache Commons اشاره کرد.

انجام دهید: کتابخانه های موجود در این دو مجموعه را جستجو و بررسی کنید.

• Google Gueva

پروژه Guava شامل مجموعه ای از کتابخانه های شرکت گوگل است که به زبان Java تولید شده اند. این کتابخانه ها برای تسهیل برنامه نویسی در موارد مختلفی نظیر collection، caching، concurrency، string processing و غیره تهیه شده اند. طراحی خوب و استفاده مناسب از Design Pattern ها از دیگر مزایای Guava است.

• Apache Commons

یک کتابخانه جاوا است که توسط بنیاد آپاچی توسعه داده می شود و با استفاده از آن کار های رایج در برنامه نویسی جاوا ساده تر می شوند. این کتابخانه شامل ۶ قسمت مهم است:

Utility classes: یکسری متد های استاتیک برای کار های رایج مثل کار با فایل ها، متد هایی برای مقایسه سریع String ها و ...

Input: پیاده سازی های کاربردی از Stream و Reader، برای مثال TeeInputStream.

Output: پیاده سازی های کاربردی از Stream و Writer برای مثال TeeOutputStream.

Filters: پیاده سازی های مختلفی از فیلتر های فایل.

Comparators: پیاده سازی های مختلفی از java.util.Comparator برای فایل ها (مقایسه کردن و sort کردن فایل ها و فولدر ها)

File Monitor: ابزاری برای مونیتور کردن رویداد های فایل سیستم (اطلاع رسانی تغییرات یک فایل و ...)

https://en.wikipedia.org/wiki/Google_Guava

<https://barnamenevis.org/showthread.php?351616-%D9%88%D8%A8%DB%8C%D9%86%D8%A7%D8%B1-%D9%87%D8%A7%DB%8C-%D8%A2%D9%85%D9%88%D8%B2%D8%B4%DB%8C-%D8%AC%D8%A7%D9%88%D8%A7>

https://en.wikipedia.org/wiki/Apache_Commons

<https://javabyab.com/15806/%DA%A9%D8%AA%D8%A7%D8%A8%D8%AE%D8%A7%D9%86%D9%87-apache-commons-io-%D8%AF%D8%B1-%D8%AC%D8%A7%D9%88%D8%A7-%DA%86%D9%87-%DA%A9%D8%A7%D8%B1%D8%A8%D8%B1%D8%AF%DB%8C-%D8%AF%D8%A7%D8%B1%D8%AF%D8%9F>

<https://javabyab.com/15806/%DA%A9%D8%AA%D8%A7%D8%A8%D8%AE%D8%A7%D9%86%D9%87-apache-commons-io-%D8%AF%D8%B1-%D8%AC%D8%A7%D9%88%D8%A7-%DA%86%D9%87-%DA%A9%D8%A7%D8%B1%D8%A8%D8%B1%D8%AF%DB%8C-%D8%AF%D8%A7%D8%B1%D8%AF%D8%9F>

انجام دهید: مستندات متدهای `concat`، `toUpperCase`، `toLowerCase`، `compareTo`، `replaceAll`، `split`، `substring` و `trim` از کلاس `String` را مطالعه کنید.

• **concat**: `String concat(String str)`

رشته ای را به رشته ی قبلی می چسباند و رشته ای را بر میگرداند که با آن الحاق شده است. این رشته به انتهای آن الحاق می شود.

• **toLowerCase و toUpperCase**: `String toLowerCase()` و `String toUpperCase()`

هر یک از این متدها به ترتیب جهت تبدیل کاراکترهای استرینگ مد نظر به حروف بزرگ و کوچک به کار گرفته می شوند. به طور مثال، فرض کنیم استرینگی به صورت «MoHammAd» را به عنوان ورودی از کاربر دریافت کرده و نیاز داریم تا تمامی کاراکترهای آن را به صورت یکسان به حروف کوچک یا بزرگ تبدیل کرده سپس در حافظه ذخیره سازیم که برای این منظور استفاده از متدهای فوق الذکر راه کار مناسبی می باشد.

• **compareTo**: `int compareTo(String anotherString)`

رشته ای را با رشته ی ورودی مقایسه می کند. اگر این دو رشته با هم برابر باشند این متد مقدار صفر بر میگرداند، اگر رشته ی ورودی از آن رشته بزرگ تر باشد مقدار منفی و اگر رشته ی ورودی از آن کوچک تر باشد، مقداری مثبت بر میگرداند.

• **replaceAll**: `String replaceAll(String regex, String replacement)`

تمام زیر رشته هایی را که با قالب دریافتی مطابقت دارند را جایگزین می کند.

• **split**: `String[] split(String regex, int limit)`

این متد یک پارامتر ورودی دارد که بر اساس آن یک رشته را به چندین زیر رشته تقسیم می کند و نتیجه، در یک آرایه ی رشته ای برگردانده می شود در واقع خروجی این متد یک آرایه است.

• **substring**: `String substring(int beginIndex, int endIndex)`

با استفاده از این متد می توانیم قسمتی از رشته اصلی را استخراج کنیم. نوع داده ای خروجی این متد از نوع رشته است. این متد دو پارامتر ورودی دارد که پارامتر اول نقطه شروع را مشخص می کند و پارامتر دوم طول رشته ای که می خواهیم استخراج کنیم را مشخص می کند.

• **trim**: `String trim()`

با استفاده از این متد می توانیم فضاهای خالی (اسپیس) یک رشته را حذف کنیم.