

: Apache Commons

1. `java.math` را برای کلاسهای ریاضی کسب و کار گسترش می دهد (کار با اعداد بزرگ).
2. بسته بندی های قابل تغییر تایپ شده را به مقادیر اولیه و `Object` ارائه می دهد.
3. کلاسهای را برای کار با و دستکاری متن ارائه می دهد ، بخشی از آن به عنوان افزونه `java.text`.
4. کلاس ها و روش هایی را برای کار با تاریخ و مدت زمان ارائه می دهد
5. کمک می کند تا روشهای برابر (`Object`) ، `ToString ()` ، `hashCode ()` و مقایسه (`To Object`) سازگار باشد.
6. کلاسهای پشتیبانی را برای برنامه نویسی چند رشته فراهم می کند.

: Google Guava

1. `java.math` را برای کلاسهای ریاضی کسب و کار گسترش می دهد (کار با اعداد مختلط).
2. خطاهای کد نویسی را کاهش می دهد ، شیوه های کدگذاری استاندارد را تسهیل می کند و باعث می شود که با خواندن کد مختصر و آسان ، کد آن را افزایش دهید.
3. شامل انواع جدید مجموعه (مانند `multimap` و `multiset`) ، مجموعه های تغییرناپذیر ، یک کتابخانه گرافیکی و برنامه های کاربردی برای همزمانی ، `I / O` ، هش کردن ، ذخیره سازی ، بدوی ها ، رشته ها و موارد دیگر می شود.

: Concat

`String` دوم را به انتهای `string` اول متصل میکند.

```
public static void main(String args[]){
    String s1="Sara ";
    String s2="Tajernia";
    String s3=s1.concat(s2);
    System.out.println(s3);//Sara Tajernia
}
```

: toUpperCase

همه ی حروف را بزرگ میکند.

: toLowerCase

همه ی حروف را کوچک میکند.

: comperTo

اختلاف دو string را به صورت عدد میدهد.

```
public static void main(String args[]) {
    String str1 = "String method tutorial";
    String str2 = "compareTo method example";
    String str3 = "String method tutorial";

    int var1 = str1.compareTo( str2 );
    System.out.println("str1 & str2 comparison: "+var1);    //-16

    int var2 = str1.compareTo( str3 );
    System.out.println("str1 & str3 comparison: "+var2);    //0

    int var3 = str2.compareTo("compareTo method example");
    System.out.println("str2 & string argument comparison: "+var3);    //0
}
```

: replaceAll

به جای همه ی افضای یک حرف خاص میتواند یک حرف دیگر قرار دهد.

```
public static void main(String args[]){
String s1="java is a very good language";
String replaceString=s1.replaceAll("a","e");//replaces all occurrences of "a" to "e"
System.out.println(replaceString);    //jave is a very good language
}
```

: Split

به در صورت داده میگیرد یکی در حالی که فقط یک string گرفته و در string اصلی اگر آن وجود داشته باشد آن را حذف کرده و به چند قسمت مجازا در بیشترین حالت تقسیم میکند.
در حالت دوم علاوه بر string یک limit هم میگیرد که مثلاً در بیشترین حالت به تعداد limit مجزا میکند.

```
public static void main(String[] args) {
    String n =" hello how are you";
    String[] q = n.split("o", 3);

    for (String a : q){
        System.out.println(a);
    }
}
```

output:

hell

h

w are you

: substring

زیر مجموعه یک string را با شمارش حروف از اول آن string مشخص میکند و قبلی ها را حذف میکند.
و اگر دو عدد باشد قبلی های عدد اول و بعدی های عدد دوم را حذف میکند.

```
public static void main(String args[]) {  
    String str= new String("quick brown fox jumps over the lazy dog");  
    System.out.println("Substring starting from index 15:");  
    System.out.println(str.substring(15));  
    System.out.println("Substring starting from index 15 and ending at 20:");  
    System.out.println(str.substring(15, 20));  
}
```

output:

Substring starting from index 15:

jumps over the lazy dog

Substring starting from index 15 and ending at 20:

Jump

: Trim

متودی است که همه ی trailing space ها را پاک میکند.

```
public static void main(String[] args) {  
    String s1 ="  hello java string  ";  
    System.out.println(s1); //Without trim()  
    String tr = s1.trim();  
    System.out.println(tr); //With trim()  
}
```

output:

hello java string

hello java string