Explain lambda expression and stream with practical examples"

lambda expression: A lambda expression is **a short block of code which takes in parameters and returns a value**. Lambda expressions are similar to methods, but they do not need a name and they can be implemented right in the body of a method.

Example:

**public** **static** **void** **main**(args[]){

List languages = Arrays.asList("Java", "Scala", "C++", "Haskell", "Lisp");

System.out.println("Languages which starts with J :");

filter(languages, (str)->str.startsWith("J"));

System.out.println("Languages which ends with a ");

filter(languages, (str)->str.endsWith("a"));

}

**public** **static** **void** **filter**(List names, Predicate condition) {

**for**(String **name:** names) {

**if**(condition.test(name)) {

System.out.println(name + " ");

} }

**Stream:** Stream API is **used to process collections of objects**. A stream is a sequence of objects that supports various methods which can be pipelined to produce the desired result. A stream is not a data structure instead it takes input from the Collections, Arrays or I/O channels.

Example:

List<String> strList = Arrays.asList("abc", "", "bcd", "", "defg", "jk");

long count = strList.stream()

.filter(x -> x.isEmpty())

.count();