

```
public class FinallyClause {
    static int count = 0;
    public static void main(String[] args) {
        while (true) {
            try {
                if (++count == 2) throw new Exception();
                if (count == 3) break;
                System.out.println("(" + count + ") No exception");
            } catch (Exception e) {
                System.out.println("(" + count + ") Exception occurred");
            } finally {
                System.out.println("(" + count + ") in finally clause");
            }
        } // end while
        System.out.println("Main program ends");
    }
}
```

1. Why does the program print the two lines below?

(2) Exception occurred
countが2に達した際に、new Exception()へ飛ぶため、Exception eの出力が実行される。

(2) in finally clause
try,catch文のあらゆる処理についてfinallyメソッドは必ず実行されるため、countが3になるbreakされるまでfinallyの中身が実行される。

2. Why does the program print the line below (It looks like the program does not print the line because of the break statement)?

(3) in finally clause
finallyの中身はどんな処理にも関係なく実行されるから。

3. Why does the program print the line below?
Main program ends

countが3となりbreakされfinallyが実行された後、try-catchを抜けて出力の処理に到達するから。