```java
public class Exceptions {

    public String checkException(double num1, double num2, char op) {

        String result = "No Exception found";

        try {
            if(op == '*' && (num1 == 0 || num2 == 0) ) {
                throw new MutliplyByZeroException("Multiplication with zero results in zero");
            }

            if(op == '/' && (num2 == 0)) {
                throw new DivideByZeroException("Division by zero results in infinity");
            }

            if(op != '+' && op != '-' && op != '*' && op != '/') {
                throw new ArithmeticException(op + " is not a valid operator");
            }
        } catch(MutliplyByZeroException mulByZeroEx) {
            result = mulByZeroEx.getMessage();
        } catch(DivideByZeroException divByZeroEx) {
            result = divByZeroEx.getMessage();
        } catch(Exception ex) {
            result = ex.getMessage();
        }

        return result;
    }
}

```


1 public class DivideByZeroException extends Exception {
2
3 public DivideByZeroException(String s) {
4 super(s);
5 }
6
7 }
8


public class MutliplyByZeroException extends Exception {
public MutliplyByZeroException(String s) {
super(s);
}
}

```
        } catch(DivideByZeroException divByZeroEx) {
            result = divByZeroEx.getMessage();
        } catch(Exception ex) {
            result = ex.getMessage();
        }

        return result;
    }

    public double calculate(double num1, double num2, char op) {
        double result = 0.0;

        if(checkException(num1, num2, op).equalsIgnoreCase("No Exception found")) {
            switch(op) {
                case '+':
                    result = num1 + num2;
                    break;
                case '-':
                    result = num1-num2;
                    break;
                case '*':
                    result = num1*num2;
                    break;
                case '/':
                    result = num1/num2;
                    break;
            }
        }

        return result;
    }
}
```