Infix to Postfix Conversion

```
package stack1;
import java.util.Stack;
public class InfixToPostfix {
                    // Function to check if a character is an operator
    private static boolean isOperator(char c) {
    return c == '+' || c == '-' || c == '*' || c == '/' || c=='^';
    }

    // Function to check the precedence of an operator
    private static int precedence(char operator) {
        switch (operator) {
            case '+':
            case '-':
                return 1;
            case '*':
            case '/':
                return 2;
            case '^': return 3;
        }
        return -1;
    }

    // Function to convert infix expression to postfix expression
    public static String infixToPostfix(String infixExpression) {
        StringBuilder postfix = new StringBuilder();
        Stack<Character> stack = new Stack<>();

        for (char c : infixExpression.toCharArray()) {
            if (Character.isLetterOrDigit(c)) {
                postfix.append(c);
            } else if (c == '(') {
                stack.push(c);
            } else if (c == ')') {
                while (!stack.isEmpty() && stack.peek() != '(') {
                    postfix.append(stack.pop());
                }
                stack.pop(); // Pop the '('
            } else if (isOperator(c)) {
                while (!stack.isEmpty() && precedence(c) <=
precedence(stack.peek())) {
                    postfix.append(stack.pop());
                }
                stack.push(c);
            }
        }

        while (!stack.isEmpty()) {
            if (stack.peek() == '(') {
                return "Invalid infix expression"; // Unmatched '('
            }
            postfix.append(stack.pop());
        }

        return postfix.toString();
    }

    public static void main(String[] args) {
        String infixExpression = "A + B * C ^ D / E";
        String postfixExpression = infixToPostfix(infixExpression);
        System.out.println("Infix Expression: " + infixExpression);
        System.out.println("Postfix Expression: " + postfixExpression);
    }
}
```