1. você pode definir um TextFormatter programaticamente no seu código JavaFX para alcançar o mesmo efeito.

Aqui está um exemplo de como você poderia fazer isso programaticamente:

java

Copy code

import javafx.application.Application;

import javafx.scene.Scene;

import javafx.scene.control.TextField;

import javafx.scene.control.TextFormatter;

import javafx.stage.Stage;

import javafx.util.converter.IntegerStringConverter;

public class Main extends Application {

@Override

public void start(Stage primaryStage) throws Exception {

TextField textField = new TextField();

// Define o número máximo de caracteres

int maxLength = 10; // Por exemplo, definindo o limite como 10 caracteres

TextFormatter<String> formatter = new TextFormatter<>(change -> {

if (change.getText().length() <= maxLength) {

return change;

} else {

return null;

}

});

textField.setTextFormatter(formatter);

primaryStage.setScene(new Scene(textField, 200, 100));

primaryStage.show();

}

public static void main(String[] args) {

launch(args);

}

}

Este exemplo cria um TextField e define um TextFormatter nele que limita o número de caracteres inseridos. O limite é definido pelo valor de maxLength.

Lembre-se de que, se você quiser usar essa abordagem no Scene Builder, precisará definir o TextFormatter no seu código JavaFX após carregar o layout do Scene Builder.