// ignore\_for\_file: prefer\_const\_constructors

import 'dart:math';

import 'package:flutter/material.dart';

double \_currentSliderValue = 0;

bool \_maiuscula = true;

bool \_minusculas = true;

bool \_caracterespecial = true;

bool \_numeros = true;

double \_range = 6;

String \_pass = '';

Widget sizedBox() {

  return SizedBox(

    width: 200,

    height: 200,

    child: Image.network(

        "https://cdn.pixabay.com/photo/2013/04/01/09/02/read-only-98443\_1280.png"),

  );

}

Widget TextoMaior() {

  return Text(

    'Gerador automático de senha',

    style: TextStyle(fontSize: 32, fontWeight: FontWeight.w500),

    textAlign: TextAlign.center,

  );

}

Widget TextoMenor() {

  return Text(

    'Aqui você escolhe como deseja gerar sua senha',

    style: TextStyle(fontSize: 16, fontWeight: FontWeight.w400),

    textAlign: TextAlign.center,

  );

}

Widget FieldSenha() {

  return TextFormField(

      keyboardType: TextInputType.emailAddress,

      decoration: InputDecoration(

          border: OutlineInputBorder(), labelText: 'Senha Gerada'));

}

class geradorSenha extends StatefulWidget {

  const geradorSenha({Key? key}) : super(key: key);

  @override

  geradorSenhaState createState() {

    return geradorSenhaState();

  }

}

class geradorSenhaState extends State<geradorSenha> {

  @override

  Widget build(BuildContext context) {

    return MaterialApp(

      home: Scaffold(

        appBar: AppBar(

          title: const Text('Gerador de senha'),

          leading: IconButton(

            icon: Icon(Icons.arrow\_back),

            color: Colors.black38,

            onPressed: () => Navigator.pop(context, false),

          ),

        ),

        body: Container(

          padding: EdgeInsets.only(top: 60, left: 40, right: 40),

          child: ListView(

            children: [

              sizedBox(),

              TextoMaior(),

              TextoMenor(),

              SizedBox(

                height: 20,

              ),

              SizedBox(

                height: 20,

              ),

              Row(children: [

                Checkbox(

                    value: \_maiuscula,

                    onChanged: (bool? value) {

                      setState(() {

                        \_maiuscula = value!;

                      });

                    }),

                Text('[A-Z]'),

                Checkbox(

                    value: \_numeros,

                    onChanged: (bool? value) {

                      setState(() {

                        \_numeros = value!;

                      });

                    }),

                Text('[0-9]'),

                Checkbox(

                    value: \_caracterespecial,

                    onChanged: (bool? value) {

                      setState(() {

                        \_caracterespecial = value!;

                      });

                    }),

                Text('[@#!]')

              ]),

              SizedBox(

                height: 20,

              ),

              Slider(

                value: 8,

                max: 100,

                divisions: 1,

                label: \_currentSliderValue.round().toString(),

                onChanged: (newRange) {

                  setState(() {

                    \_range = newRange;

                  });

                },

              ),

              SizedBox(

                height: 20,

              ),

              Container(

                  height: 50,

                  padding: const EdgeInsets.fromLTRB(10, 0, 10, 0),

                  child: ElevatedButton(

                    child: const Text('Gerar senha'),

                    onPressed: () => {geradorPassword()},

                  )),

              SizedBox(

                height: 20,

              ),

              Text(\_pass),

            ],

          ),

        ),

      ),

    );

  }

}

void geradorPassword() {

  List<String> charList = <String>[

    \_maiuscula ? 'ABCDEFGHIJKLMNOPQRSTUVWXYZ' : '',

    \_minusculas ? 'abcdefghijklmnopqrstuvwxyz' : '',

    \_numeros ? '0123456789' : '',

    \_caracterespecial ? '!@#\$%&\*-=+,.<>;:/?' : ''

  ];

  final String chars = charList.join('');

  Random rnd = Random();

  \_pass = String.fromCharCodes(Iterable.generate(

      \_range.round() - 2, (\_) => chars.codeUnitAt(rnd.nextInt(chars.length))));

}