

Zadanie 3. Generator hasła

Napisz aplikację, która będzie generować losowe hasła składające się z małych i dużych liter alfabetu angielskiego.

Layout zawiera nagłówek, przycisk do generowania hasła oraz obszar do wyświetlania wygenerowanego hasła.

Opis Layoutu XML

- Użyj kontenera typu relatywnego.
- Określa szerokość layoutu, która ma zajmować całą dostępną przestrzeń na ekranie.
- Określa wysokość layoutu, która ma zajmować całą dostępną przestrzeń na ekranie.
- Ustawia tło layoutu na kolor o kodzie szesnastkowym **#CCFF90** kolor tła umieść w zasobach aplikacji (np. **res/values/colors.xml**)
- **TextView (textView):** Wyświetla nagłówek "Generator Hasła".
 - Szerokość TextView ma być dopasowana do szerokości rodzica.
 - Wysokość TextView ma być dopasowana do zawartości tekstu.
 - Tekst wyświetlany w TextView **"Generator Hasła"**:
 - Rozmiar tekstu (40 punktów).
 - Wyrównanie tekstu do środka.
 - Wewnętrzny margines (20 pikseli) wokół tekstu.
 - Kolor tekstu, pobrany z zasobów aplikacji (**#1A237E**)
 - Kolor tła TextView ustawione na kolor o kodzie szesnastkowym **#EADC61** (odcień żółtego) , pobrany z zasobów aplikacji.
- **Button (button):** Przycisk do generowania hasła.
 - Szerokość Button ma być dopasowana do zawartości tekstu.



- Wysokość Button ma być dopasowana do zawartości tekstu.
 - Tekst wyświetlany na przycisku "**Generuj hasło**":
 - Kolor tła przycisku ustawiony na **#EADC61** (odcień żółtego) , pobrany z zasobów aplikacji.
 - Pozycja przycisku poniżej TextView
 - Wewnętrzny margines (20 pikseli) wokół tekstu przycisku.
 - Rozmiar tekstu na przycisku (20 punktów).
 - Kolor tekstu na przycisku, pobrany z zasobów aplikacji (**#1A237E**).
- **passwordTextView** : Wyświetla wygenerowane hasło.
 - Szerokość TextView ma być dopasowana do zawartości tekstu.
 - Wysokość TextView ma być dopasowana do zawartości tekstu.
 - Wyśrodkowanie TextView wewnątrz rodzica (RelativeLayout).
 - Rozmiar tekstu (40 punktów).
 - Styl tekstu pogrubiony.
 - Kolor tekstu ustawiony na biały, pobrany z zasobów aplikacji.

Napisz klasę PasswordGenerator, która będzie generować losowe hasła składające się z małych i dużych liter alfabetu angielskiego.

Specyfikacja klasy:

- Nazwa klasy: PasswordGenerator
- Pakiet: com.example.PasswordGenerator

Metoda:

- `public static String generatePassword(int length)`: Metoda statyczna generująca losowe hasło.
 - Parametr `length`: Długość generowanego hasła.
 - Zwraca: Losowo wygenerowane hasło jako String.

W pliku MainActivity.java zdefiniuj nasłuchiwanie buttona, który wyświetla na TextView wygenerowane hasło.

```

package com.example.PasswordGenerator;
import java.util.Random;
2 usages
public class PasswordGenerator {

    2 usages
    static final String letters = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz";

    1 usage
    public static String generatePassword(int length) {

        Random random = new Random();
        StringBuilder password = new StringBuilder();

        for (int i = 0; i < length; i++) {
            int randomIndex = random.nextInt(letters.length());
            char randomChar = letters.charAt(randomIndex);
            password.append(randomChar);
        }

        return password.toString();
    }
}

```

Dodatkowo do samodzielnego wykonania :

Zaimplementuj EditText z właściwością

android: hint = "ilość znaków", którym użytkownik określa jak długie ma być hasło oraz

CheckBox „znaki specjalne”, który po uruchomieniu jest niezaznaczony. Po zaznaczeniu tej opcji mają się losowo generować dodatkowo znaki specjalne.

Ustawienia wspólne dla obu elementów (**EditText** i **CheckBox**) zamieść w pliku style.xml i podłącz do elementów **EditText** i **CheckBox**:

1. szerokość elementu na dopasowującą się do zawartości.
2. wysokość elementu na dopasowującą się do zawartości.
3. wewnętrzny odstęp o wartości 20dp do wszystkich czterech stron elementu.
4. kolor tekstu na biały
5. rozmiar tekstu na 30sp
6. zewnętrzny margines z lewej strony elementu o wartości 20dp.

