

Task 1

Aim: Create Login/Registration /Sign Up Page as per application

Tools: Android Studio, Ms Word

Description about Application:

A signup page (also known as a registration page) enables users and organizations to independently register and gain access to your system. It is common to have multiple signup pages depending on the types of people and organizations you want to register.

Registration only happens the first time you access the system. It is a way to check your credentials. Every time after your initial registration, you will log on to the system using the username and password you created.

Our sign-in page has features like automatic Google sign-in, forgot password, skip and create new account option.

Due to registration or login, users will be able to authenticate themselves first before opening the notes app so that their information is safe from accidental viewing of other people.

Code:

```
package com.singhvikrant.mobilecomputingtasks

import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.provider.AlarmClock.EXTRA_MESSAGE
import android.text.method.HideReturnsTransformationMethod
import android.text.method.PasswordTransformationMethod
import android.view.View
import android.widget.EditText

import android.widget.ImageView
import android.widget.TextView

class LoginActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
```

```
setContentView(R.layout.activity_login)
```

```
val showPass: ImageView = findViewById(R.id.show_pass_btn) val hidePass: ImageView =  
findViewById(R.id.hide_pass_btn)
```

```
val password: EditText = findViewById(R.id.userPass)
```

```
val skip: TextView = findViewById(R.id.skip)
```

```
val createAccount: TextView = findViewById(R.id.signUp) showPass.setOnClickListener {
```

```
showPass.visibility = View.GONE
```

```
hidePass.visibility = View.VISIBLE
```

```
password.transformationMethod = HideReturnsTransformationMethod.getInstance() }
```

```
hidePass.setOnClickListener {
```

```
showPass.visibility = View.VISIBLE
```

```
hidePass.visibility = View.GONE
```

```
password.transformationMethod = PasswordTransformationMethod.getInstance() }
```

```
skip.setOnClickListener {
```

```
val message = ""
```

```
val intent = Intent(this, MainActivity::class.java).apply {
```

```
putExtra(EXTRA_MESSAGE, message)
```

```
} startActivity(intent)
```

```
finish()
```

```
} createAccount.setOnClickListener {
```

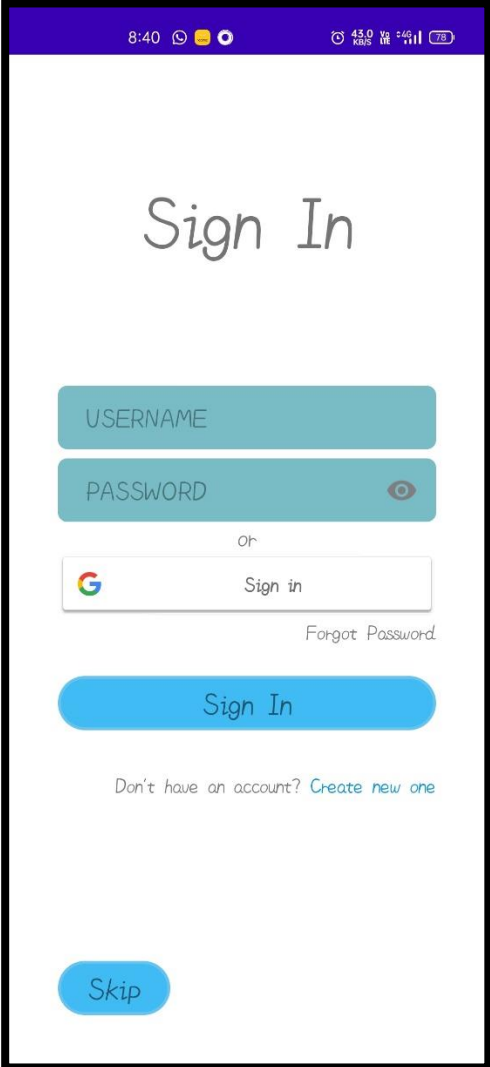
```
val message = "" val intent = Intent(this, SignupActivity::class.java).apply {
```

```
putExtra(EXTRA_MESSAGE, message)
```

```
} startActivity(intent)
```

```
finish() } }}
```

OUTPUT:




Sign In

USERNAME

PASSWORD

or


 Sign in

[Forgot Password](#)

Sign In

[Don't have an account? Create new one](#)

Skip



Sign Up

USERNAME

EMAIL

PASSWORD

Sign In

[Already have an account? Sign In](#)

Conclusion:

Hence, we have created an authentication page for our application and added features we believe would be user friendly.